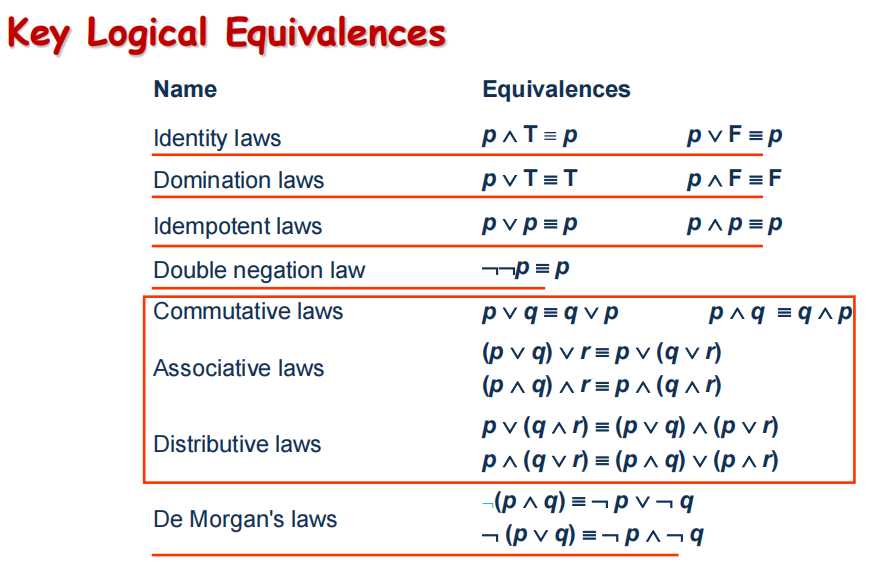
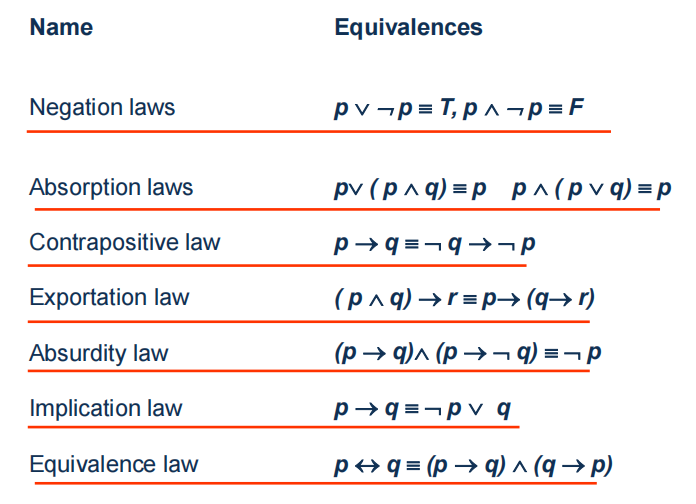
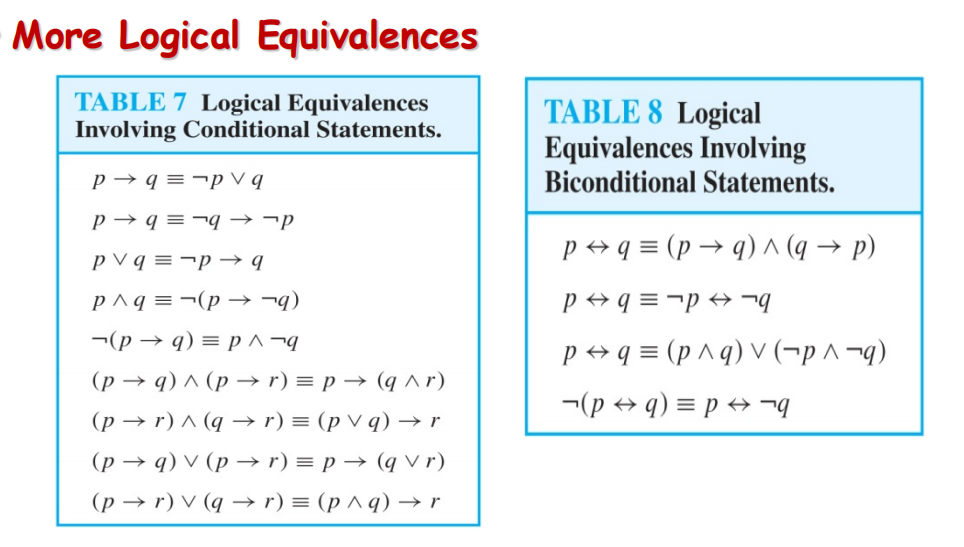
**离散数学第一章（1.3开始）**

**（1.3）**







**The Dual of a Compound Proposition**

**（对偶式）**

**The dual of compound proposition that contains only the logical operators∨ , ∧ and ¬ is the proposition obtained by replacing each ∨ by ∧,each ∧ by ∨,each T by F and each F by T. The dual of *S* is denoted by *S*\*. （对偶式前提，只包含∨、∧、¬这三个运算符*。*）**

**For example,**

1. ***S*=(*p* ∨ *¬* *q*) ∧*r* ∨ *T* *S*\*= (*p* ∧ *¬* *q*) ∨ *r* ∧ *F***
2. ***S= (p ∧ q) → (p ∨ q) ≡ ¬(p ∧ q) ∨ (p ∨ q)***

***S\*= ¬(p ∨ q) ∧ (p ∧ q)***

***【Theorem】 let s and t are two compound propositions, s ≡ t if and***

***only if s\* ≡ t\* .***

**Functionally Complete Collection of Logical Operators（逻辑运算符的功能完整集合）**

**A collection of logical operators is called functionally complete if every**

**compound proposition is logically equivalent to a compound proposition involving only these logical operators.**

**For example, {¬,∧,∨,→,↔},{¬, ∧ , ∨}, {¬,∧},{¬,∨},{|},{↓}**

**are all functionally complete operators.**

**Propositional Normal Forms （命题范式）**

**There are two types of normal forms in propositional calculus:**

**disjunctive normal form(DNF，析取范式) and conjunctive normal form(CNF，合取范式)**

**Disjunctive normal form（析取范式）**

**A *literal* is a variable or its negation.**

**Conjunctions（合取式） with literals as conjuncts are called *conjunctive clauses (clauses) 基本积*. ————相应的，合取范式中相应的那些析取式就称为基本和。**

**For example,** *p* ∧ *q*, *p* ∧ ¬*q*, ¬*p* ∧ *q*, ¬*p* ∧ ¬*q*

***A formula is said to be in disjunctive normal form if it is written as a***

***disjunction, in which all the terms are conjunctions of literals.***

( *p* ∧ *q*) ∨ ( *p* ∧ ¬*q*) ****  *p ∧*  ( *p* ∨ *q*) ****

More DNF or CNF：

¬p ∨ (q ∧ ¬r) DNF

¬p ∧ (q ∨ ¬r) ∧ (¬q ∨ r) CNF

p DNF & CNF

¬p ∨ q （基本和） DNF & CNF

¬p ∧ q ∧ ¬r （基本积） DNF & CNF

（**析取范式是一些基本积做析取，合取范式是一些基本和做合取，而单个命题变量既可以看成DNF,也可以看成CNF，由此可得一个基本和和基本积也可以看做DNF或CNF**）

【 Theorem 1 】Any formula A is tautologically equivalent to some formula in DNF (CNF).

Proof:先列出真值表，然后找命题为真的那几组变量赋值，再把他们通过（a1^b1^c1^...^¬m1^ ¬n1^...）∨（a2^b2^c2^...^¬m1^ ¬n2^...）∨

类似的的形式构成一个DNF；或者找命题为假的那几组变量赋值，再把他们通过（具体操作在后面……）

**How to obtain normal form**

**(1) Use of the following logical equivalences to eliminate →,↔.**

***p* → *q* ≡** ¬***p* ∨ *q***

***p* ↔ *q* ≡ (*p* → *q*) ∧(*q* → *p*)**

**(2) Use of the following logical equivalences to eliminate** ¬ **,∨, ∧ from the scope of** ¬ **such that any** ¬ **has only an atom as its scope.**

¬**(*p*1 ∨ *p*2∨…∨ *pn*) ≡**¬ ***p*1 ∧** ¬ ***p*2 ∧…∧** ¬ ***pn***

¬¬***p* ≡ *p***

1. **Use of the commutative laws, the distributive laws and the associative laws to obtain normal form.**

**Example : Convert the following formula into conjunctive and disjunctive normal forms. ¬( p ∨ q)↔( p ∧ q)**

***solution:***

***¬( p ∨ q)↔( p ∧ q)***

***≡ (¬( p ∨ q) → ( p ∧ q)) ∧ (( p ∧ q) → ¬( p ∨ q))***

***≡ (( p ∨ q) ∨ ( p ∧ q)) ∧ (¬( p ∧ q) ∨ ¬( p ∨ q))***

***≡ (( p ∨ q ∨ p) ∧ ( p ∨ q ∨ q)) ∧ ((¬p ∨ ¬q) ∨ (¬p ∧ ¬q))***

***≡ ( p ∨ q) ∧ (¬p ∨ ¬q ∨ ¬p) ∧ (¬p ∨ ¬q ∨ ¬q)***

***≡ ( p ∨ q) ∧ (¬p ∨ ¬q)\****

***≡ (( p ∨ q) ∧ ¬p) ∨ (( p ∨ q) ∧ ¬q)***

***≡ ( p ∧ ¬p) ∨ (q ∧ ¬p) ∨ ( p ∧ ¬q) ∨ (q ∧ ¬q)\*\****

***≡ (q ∧ ¬p) ∨ ( p ∧ ¬q)\*\*\****

**Example : Find the assignments of *p* and *q* for which the following formula is true. ( p → q) → p**

***solution: The assignments of p and***

***( p → q) → p q for which the formula is***

***≡ ¬(¬p ∨ q) ∨ p true:***

***≡ ( p ∧ ¬q) ∨ p p q***

***≡ p T F***

***T F***

**Conjunctive normal form（合取范式）**

 **A compound proposition is in Conjunctive Normal Form (CNF) if it is a conjunction（合取式） of disjunctions（基本和）.**

 **Every proposition can be put in an equivalent CNF.**

 **Conjunctive Normal Form (CNF) can be obtained by eliminating implications（用等价公式消去蕴含式）, moving negation inwards（用德摩根定律把否定词移到里面去） and using the distributive and associative laws（用分配律和结合律，有必要也可以用交换律）.**

 **Important in resolution theorem proving used in artificial Intelligence (AI).**

 **A compound proposition can be put in conjunctive normal form through repeated application of the logical equivalences covered earlier.**

**Full disjunctive normal form（主析取范式）**

1. **Minterm（极小项） & Maxterm（极大项）**

**A *minterm* is a conjunctive of literals in which each variable is represented exactly once. (每个命题变量恰好出现一次）**

**For example,**

**If a formula has the variables *p*, *q*, *r*, then *p*∧** ¬***q*∧ *r* is a minterm, but *p*∧** ¬***q* and *p*∧** ¬***p*∧ *r* are not.**

1. **Full disjunctive normal form**

**If a formula is expressed as a disjunction of minterms（极小项的析取）, it is said to be in *full disjunctive normal form.***

**For example,**

( *p* ∧ *q* ∧ *r*) ∨ ( *p* ∧ *q* ∧ ¬*r*) ∨ (¬*p* ∧ *q* ∧ *r*) ∨ (¬*p* ∧ ¬*q* ∧ ¬*r*)

1. **How to obtain full disjunctive normal form**

**Any formula *A* is tautologically equivalent to a formula in full disjunctive normal form.**

**First, obtain disjunctive normal form, then use of negation law and**

**distributive laws to obtain full disjunctive forms. （首先，先找到它的析取范式，再用一些等价公示把它转换为主析取范式）**

*A* ≡ *A*∧ (*q* ∨ ¬*q*) ≡ (*A*∧ *q*) ∨ (*A*∧ ¬*q*) (缺哪个变量就补哪个）

**Example : Convert the following formula into full disjunctive normal form.** ( *p* ∧ *q*) ∨ (¬*p* ∧ *r*) ∨ (*q* ∧ *r*)

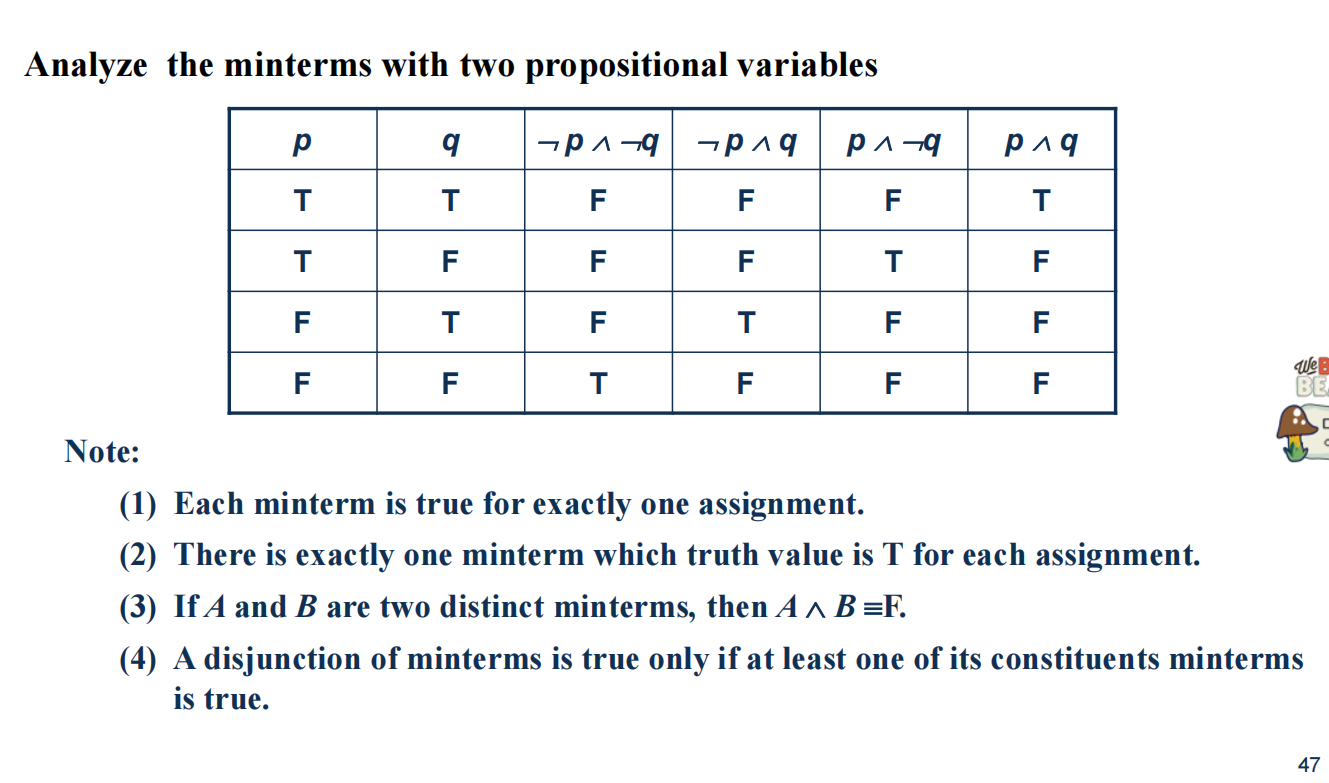
***solution:***

***( p ∧ q) ∨ (¬p ∧ r) ∨ (q ∧ r)***

***≡ ( p ∧ q ∧ (r ∨ ¬r))∨ (¬p ∧ (q ∨ ¬q) ∧ r) ∨ (( p ∨ ¬p) ∧ q ∧ r)***

***≡ ( p ∧ q ∧ r)∨ ( p ∧ q ∧ ¬r)∨ (¬p ∧ q ∧ r)∨ (¬p ∧ ¬q ∧ r)∨ ( p ∧ q ∧ r)∨ (¬p ∧ q ∧ r)***

***≡ ( p ∧ q ∧ r) ∨ ( p ∧ q ∧ ¬r) ∨ (¬p ∧ q ∧ r) ∨ (¬p ∧ ¬q ∧ r)***



**掌握第三条！（tips：minterms（极小项））**

**翻译：**

1. **对每一组赋值只有一个极小项为真**

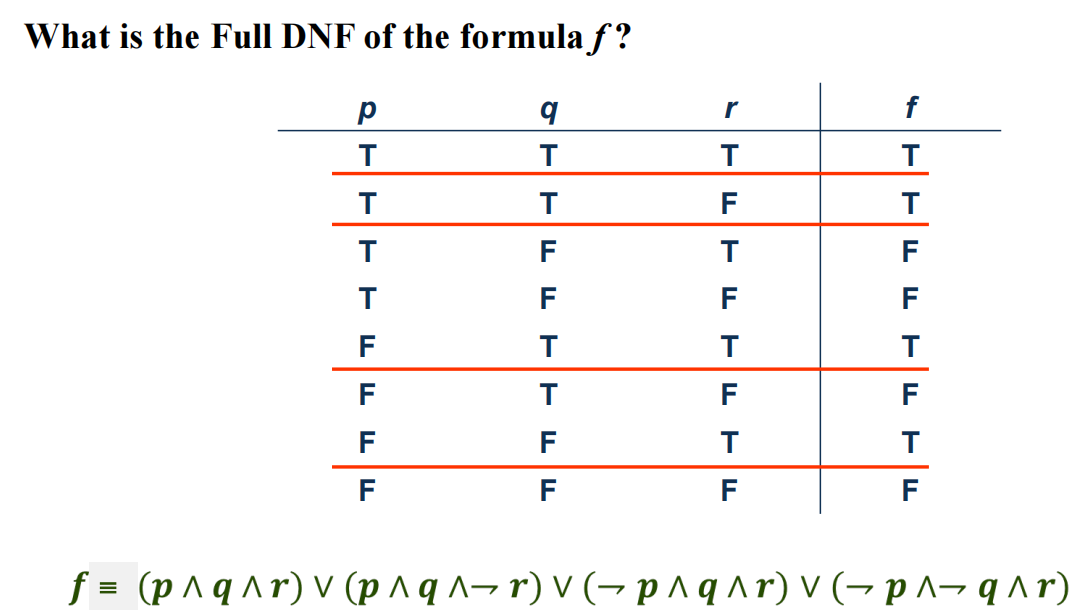
**（3）不同的极小项做合取一定为假**

**（4）极小项的析取式为真当且仅当其中有一组极小项为真**

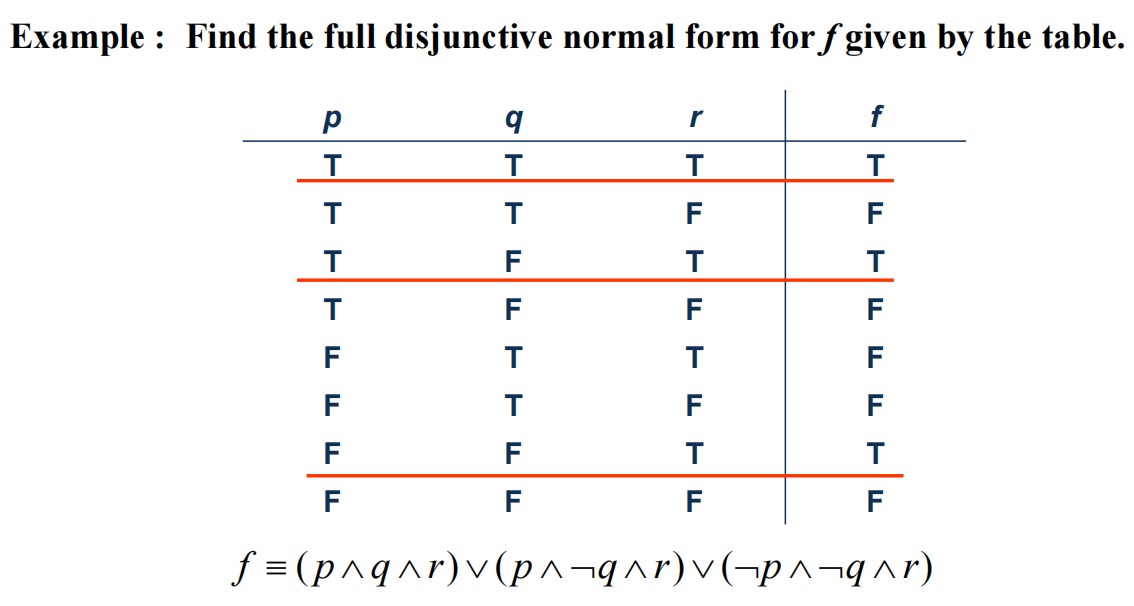
找一个公式f的DNF和CNF方法:

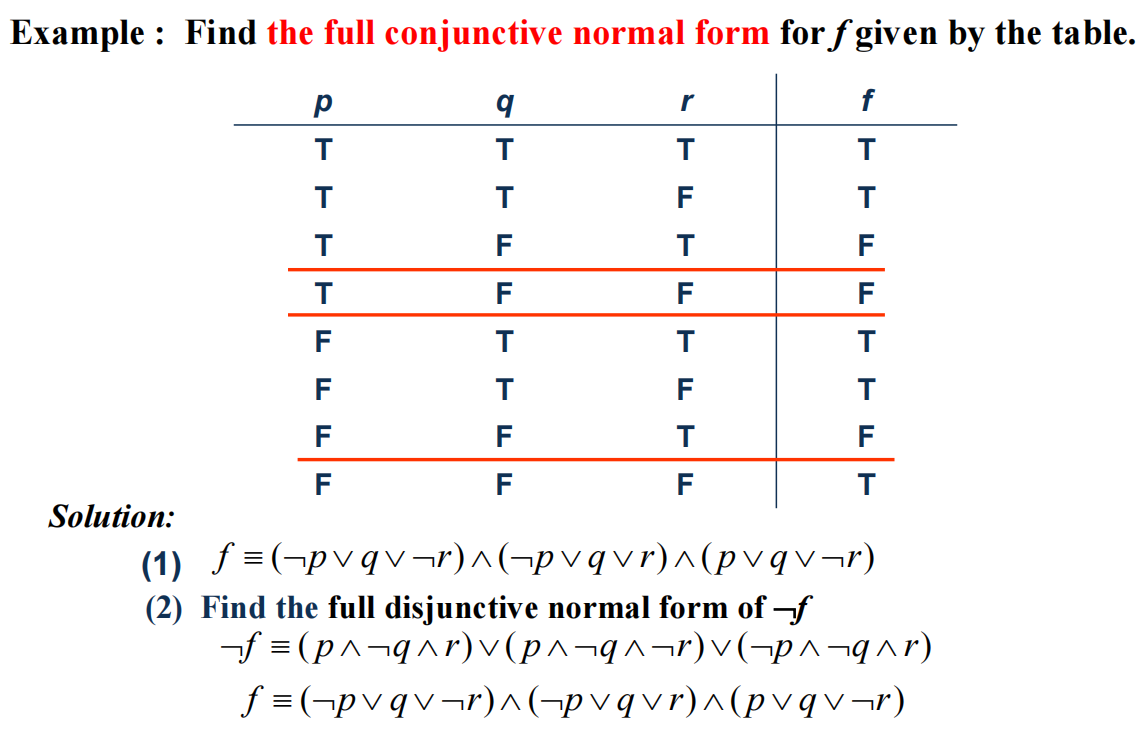
先列出真值表，然后找f为真的那几组变量赋值，再把他们通过（a1^b1^c1^...^¬m1^ ¬n1^...）∨（a2^b2^c2^...^¬m1^ ¬n2^...）∨...

类似的的形式构成一个DNF；或者找到¬f的DNF（具体方法是先找f为假的那几组命题变量，再把结果取反¬f，然后找到它的DNF，再由取反律得到CNF，下面有例子）



**（一些通过真值表确定主析取范式的例子）**





**（上面是找主合取范式的例子，可以借鉴主析取范式的做法）**

**（1.4）**

**Universal Quantifier**

**∀*x P*(*x*)is read as *“*For all *x*, *P*(*x*)” or “For every *x*, *P*(*x*)”**

**Existential Quantifier**

 **∃*x P*(*x*) is read as *“*For some *x*, P(*x*)”, or as “There is an**

***x* such that P(*x*),” or “For at least one *x*, P(*x*).”**

**Thinking about Quantifiers**

 **When the domain of discourse is finite, we can think of**

**quantification as looping through（遍历） the elements of the**

**domain.**

• To evaluate ∀*x P*(*x*) loop through all *x* in the domain.

 If at every step P(*x*) is true, then ∀*x P*(*x*) is true.

 If at a step P(*x*) is false, then ∀*x P*(*x*) is false and the loop terminates（终止）.

• To evaluate ∃*x P*(*x*) loop through all *x* in the domain.

 If at some step, P(*x*) is true, then ∃*x P*(*x*) is true and the loop terminates.

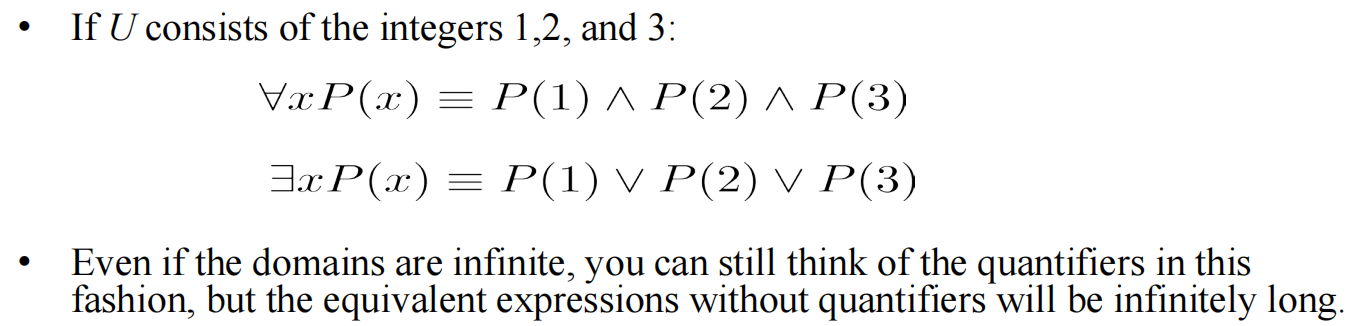
 If the loop ends without finding an *x* for which P(*x*) is true, then ∃*x P*(*x*) is false.

• Even if the domains are infinite, we can still think of the quantifiers this

fashion, but the loops will not terminate in some cases.

 If the domain is finite（有限的）, a universally quantified proposition is equivalent to a conjunction of propositions without quantifiers and an existentially quantified proposition is equivalent to a disjunction of

propositions without quantifiers. （见下图）



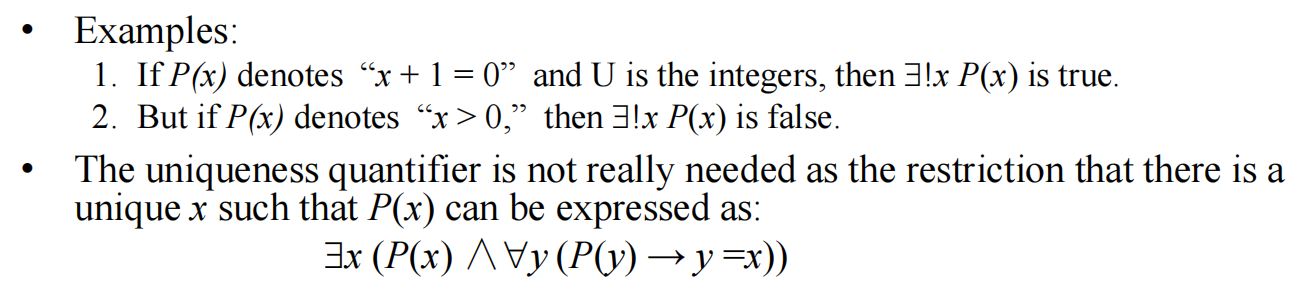
**Uniqueness Quantifier（唯一性量词）**

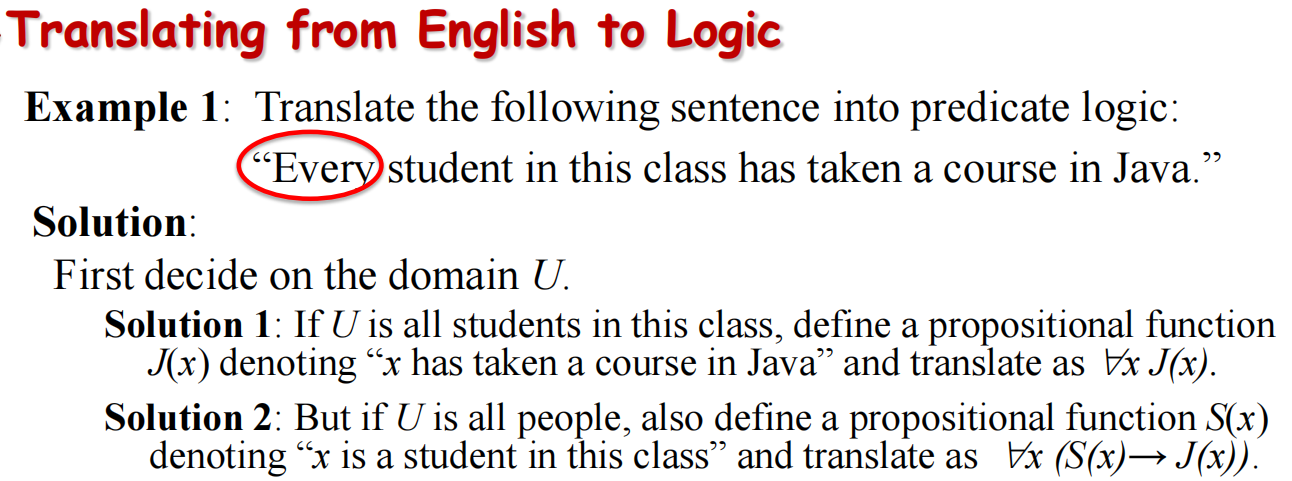
 **∃!*x P*(*x*) means that *P*(*x*) is true for one and only one *x* in the universe of discourse.**

• This is commonly expressed in English in the following equivalent ways:

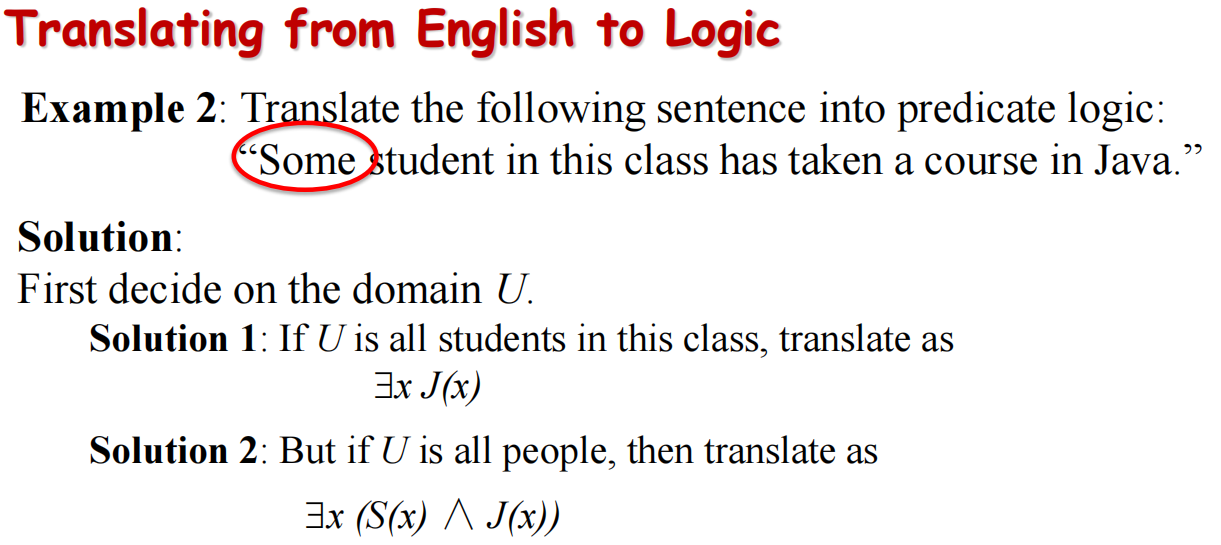
– “There is a unique *x* such that *P*(*x*).”

– “There is one and only one *x* such that *P*(*x*)”

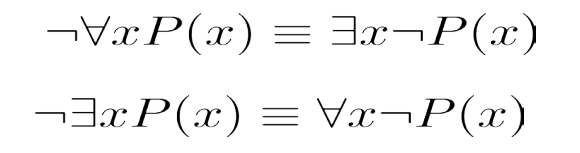


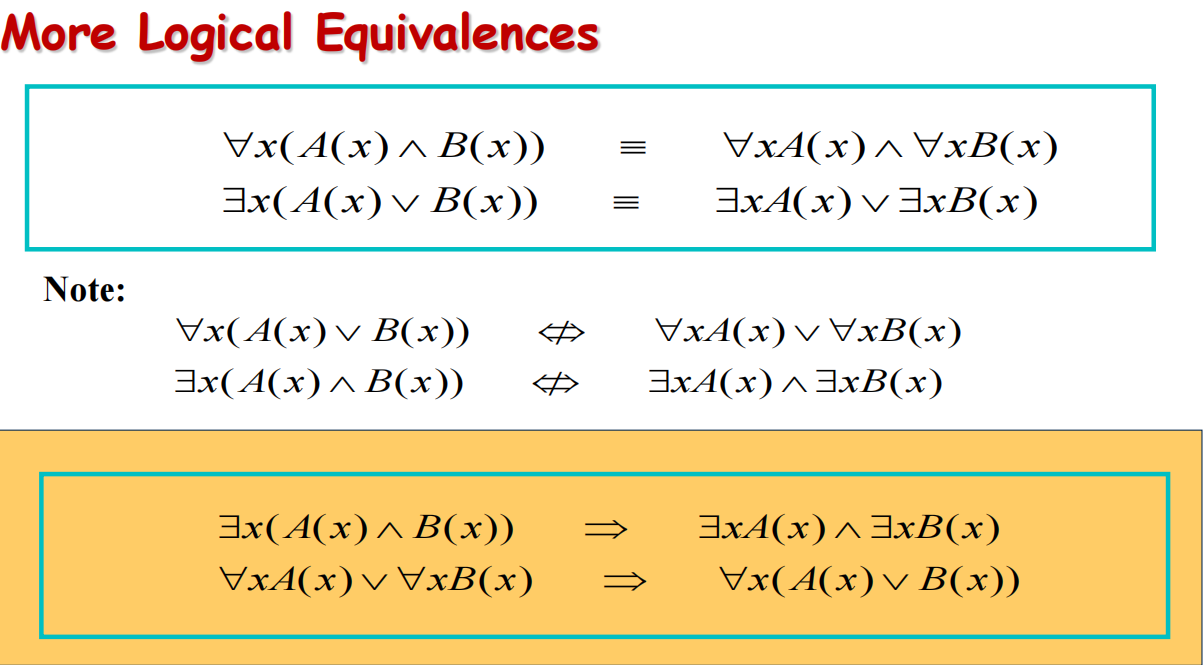


**（*∀用的是→ ，*∃用的是∧！！！）**



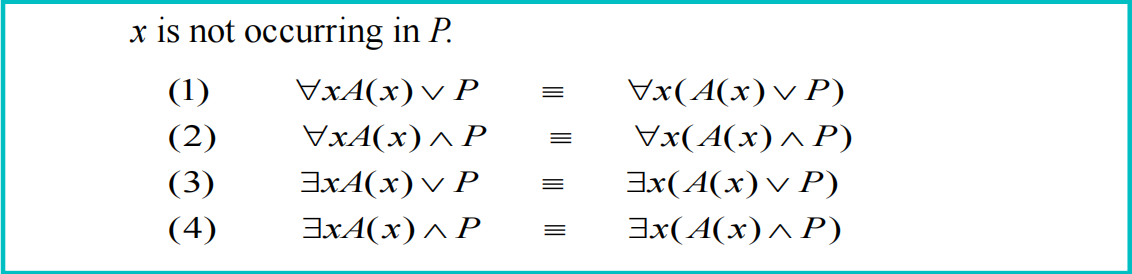
**De Morgan’s Laws for Quantifiers**



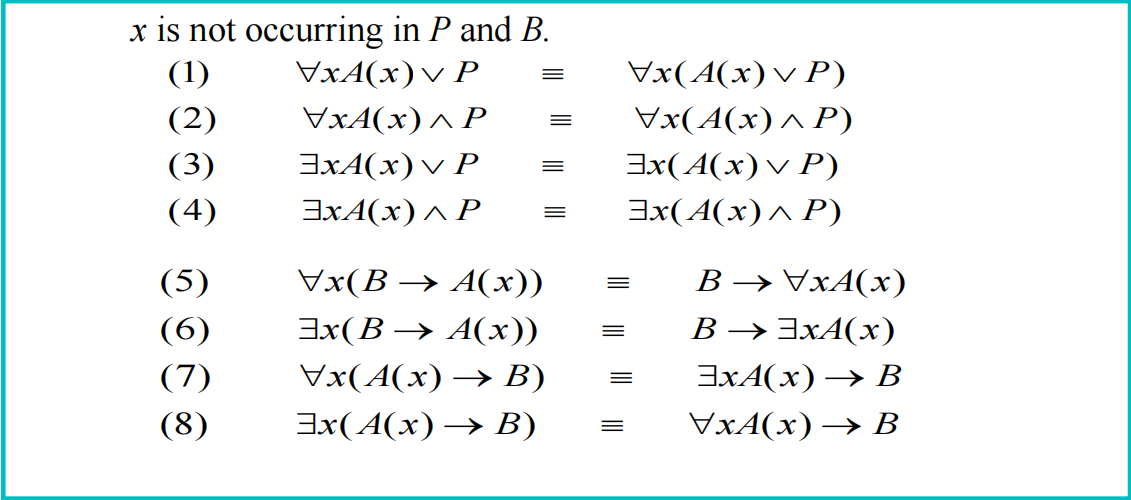


**More Logical Equivalences**

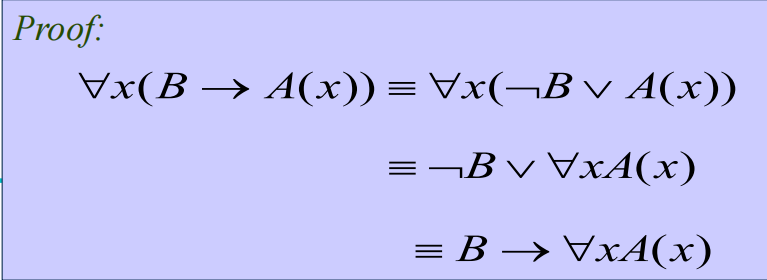
**（P不在x的辖域内）**

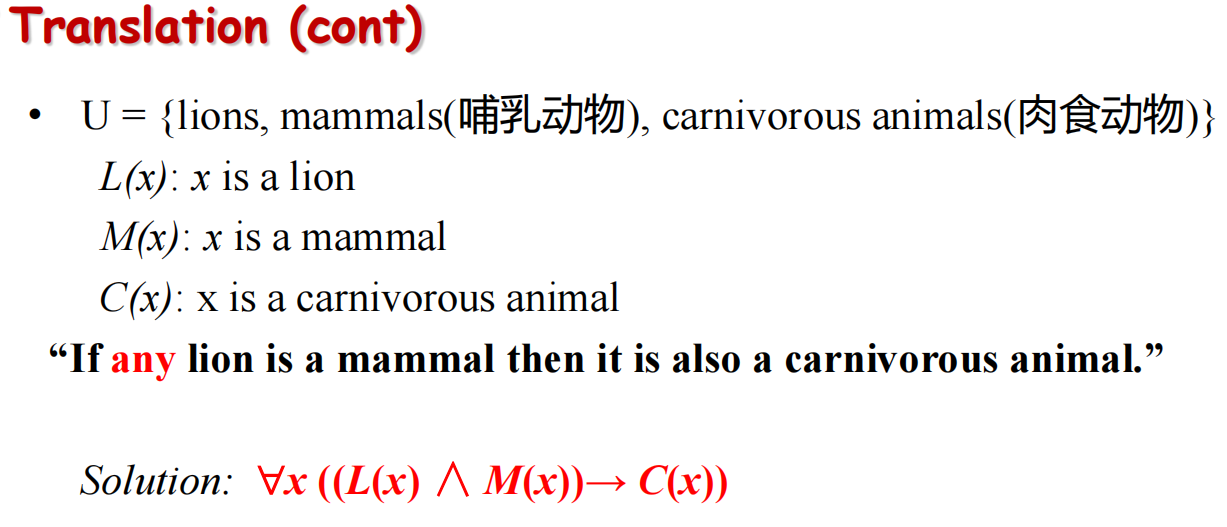


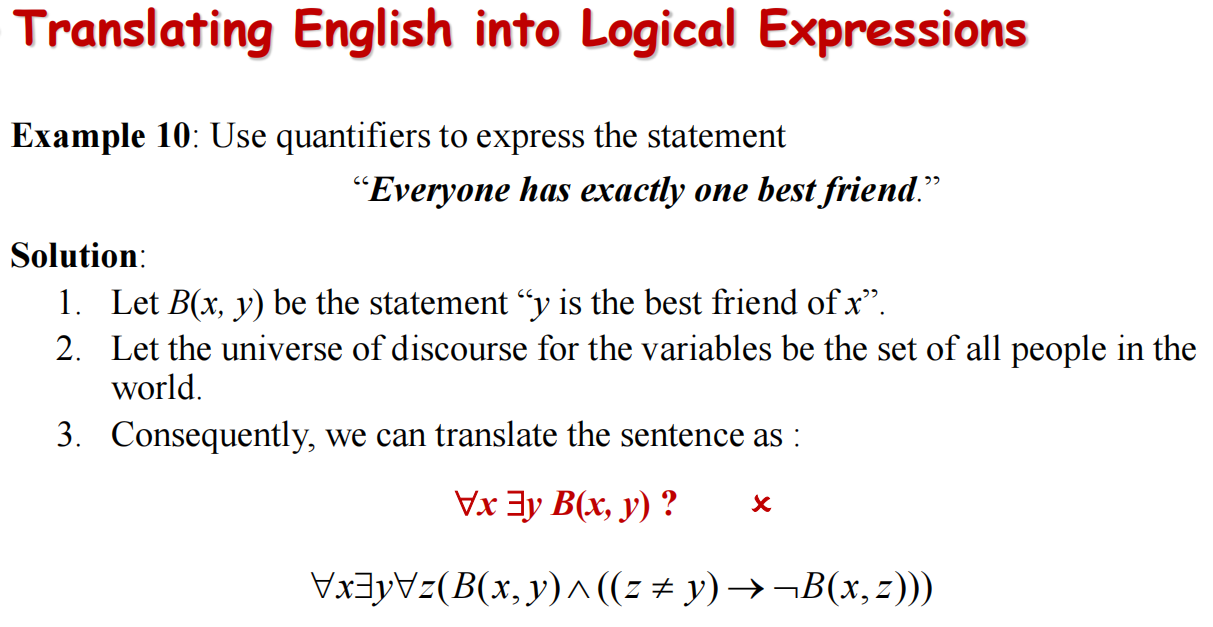
**（P和B不在X的辖域中）**



**（举例，第5条的证明）**



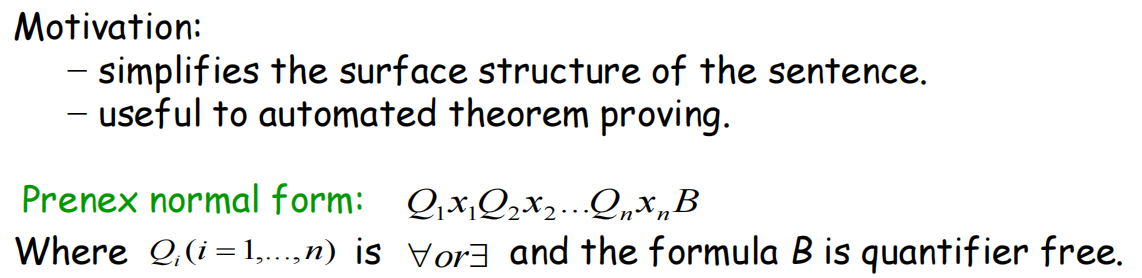




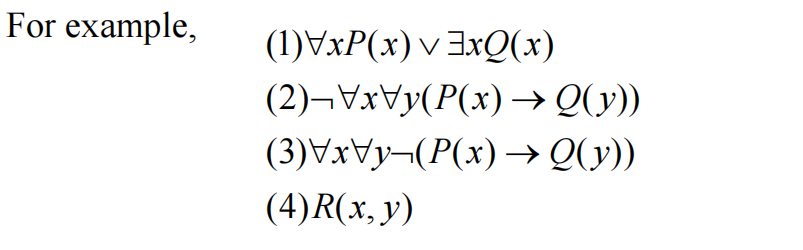
**（注意是exactly one，有且仅有一个）**

**（1.5）**

**Prenex Normal Forms （前束范式）**



**（注意，Qi(i =1,...,n)不能是否定式！！）**



**（其中只有3和4是前束范式！！）**

**Algorithm for prenex normal form**

**Any expression can be converted into prenex normal form.**

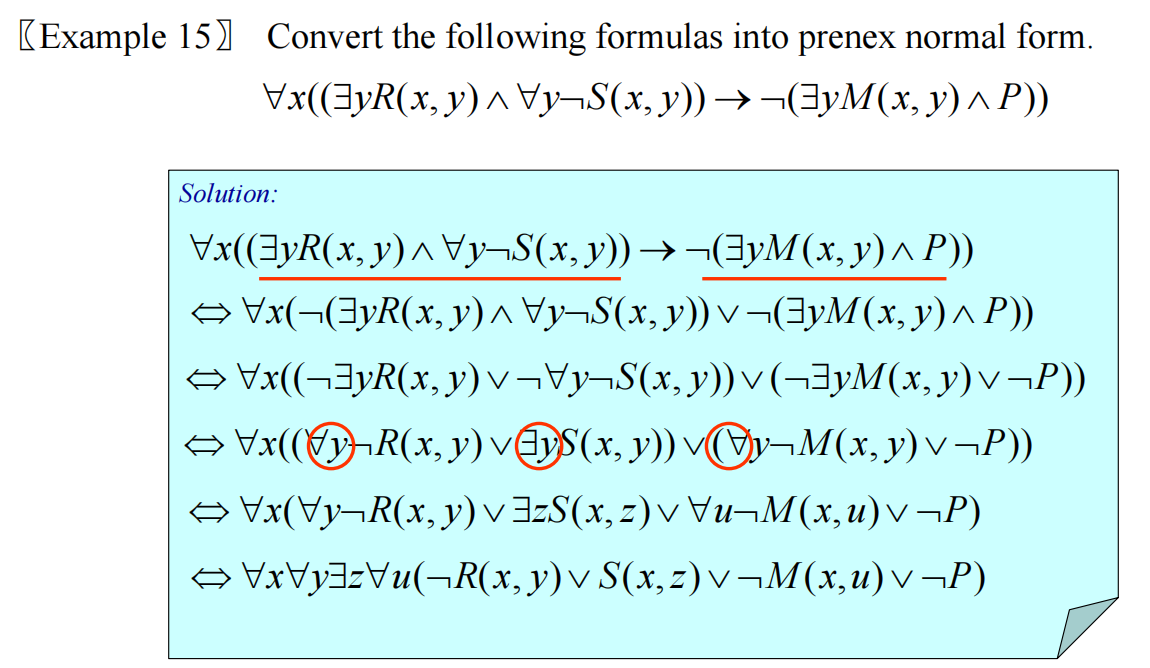
How to obtain prenex normal form?

1. Eliminate all occurrences of → and ↔ from the formula in question.（前束范式中不能包含箭头）

2. Move all negations inward such that（使得）, in the end, negation only appear as part of literals. （把所有否定都用德摩根定律移到最后的literal中）

3. Standardize the variables a part(when necessary).（给变量重命名）

4. The prenex normal form can now be obtained by moving all quantifiers to the front of the formula



（倒数第三步-->倒数第二步采用了变量重命名，使得所有变量可以移到前面去）

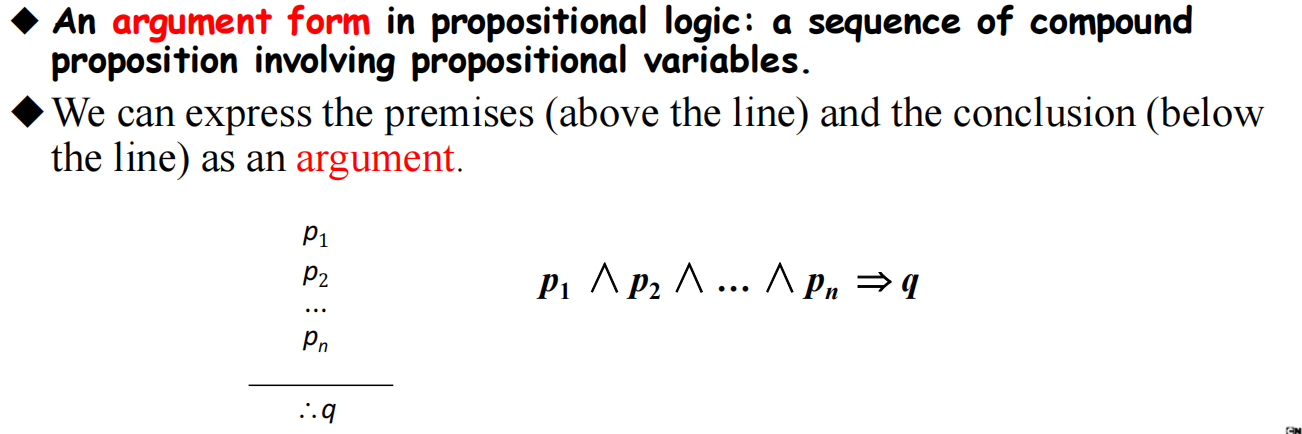
**（1.6）**

**Premise、hypotheses（前提，后者是复数形式）**

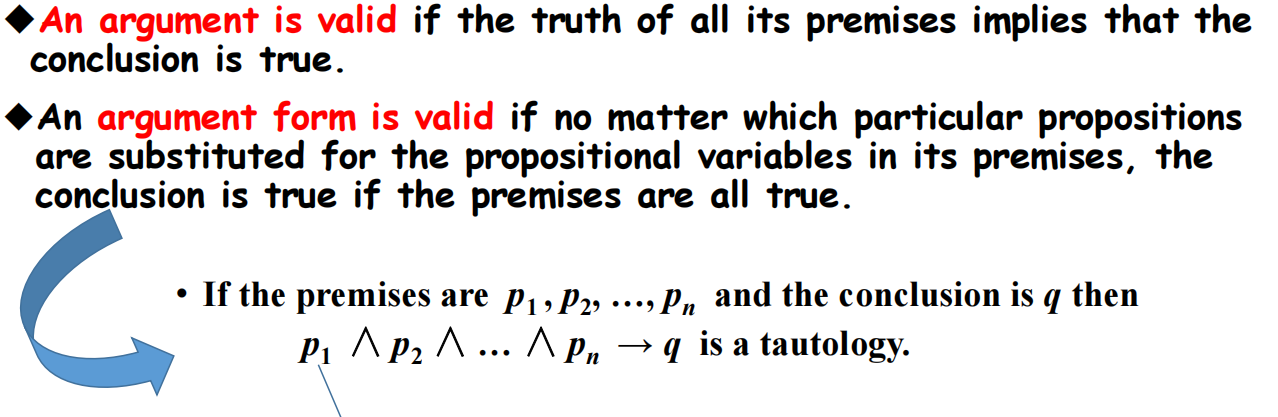
**Arguments**

**An argument in propositional logic is a sequence of propositions.**

**All but（除了……都是） the final proposition are called premises. The last statement is the conclusion.**



**Valid（有效的） Arguments & Argument Form**



**Rules of Inference**

• **Propositional Logic**

Inference Rules

 **Modus Ponens （假言推理）**

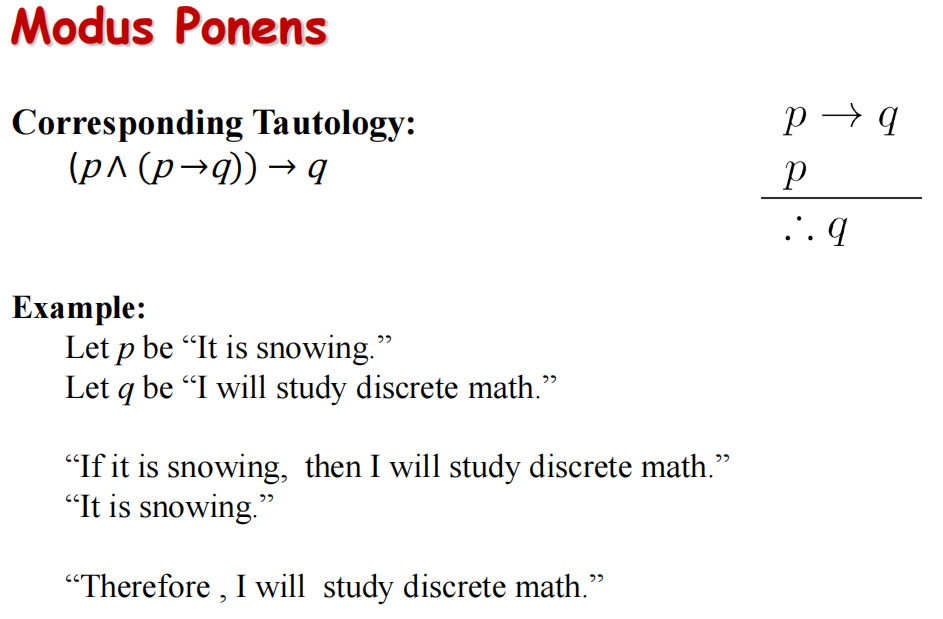
 **Modus Tollens （拒取式）**

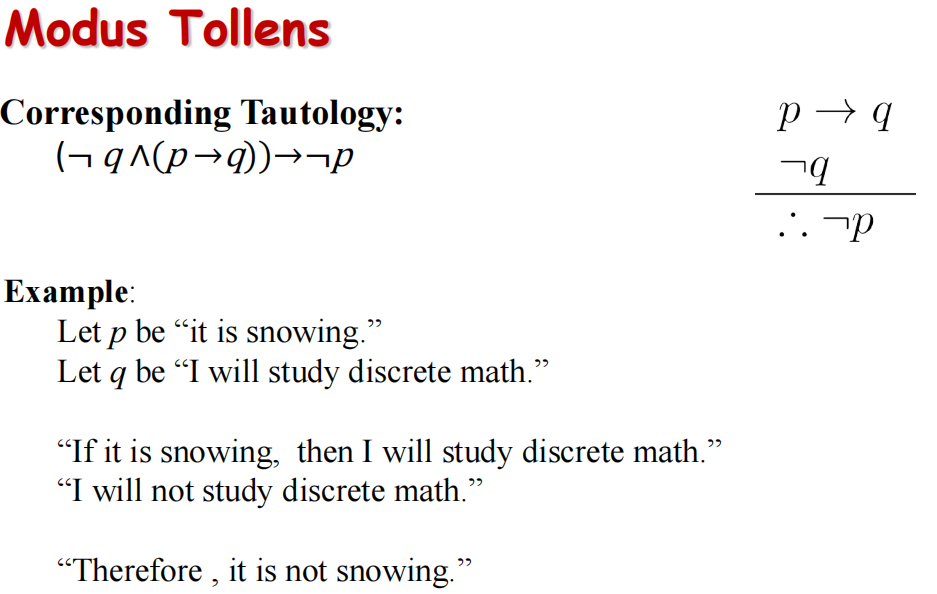
 **Hypothetical Syllogism （假言三段论）**

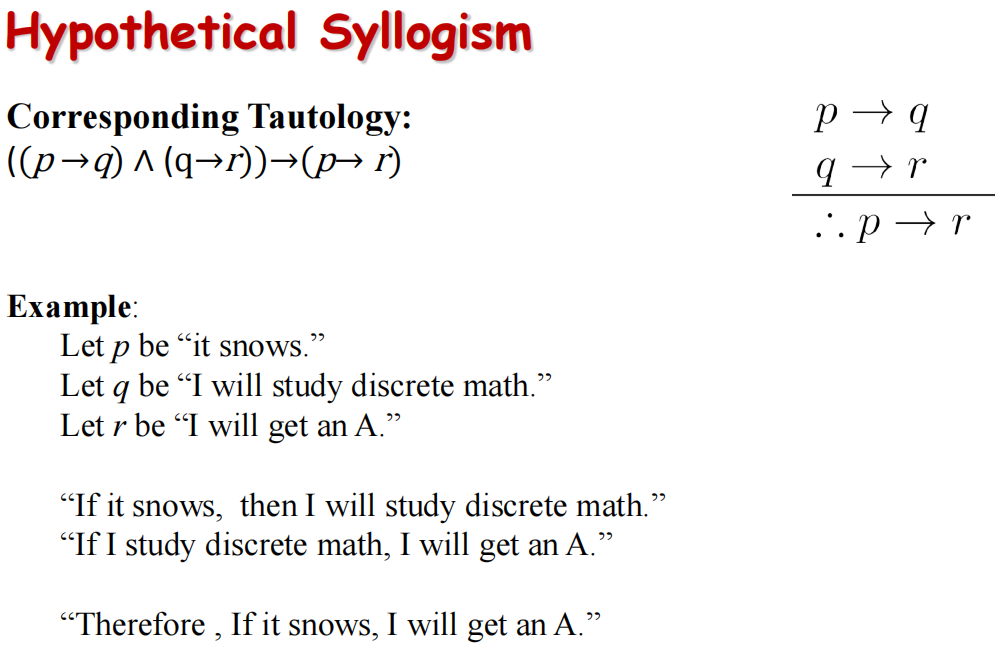
 **Disjunctive Syllogism （析取三段论）**

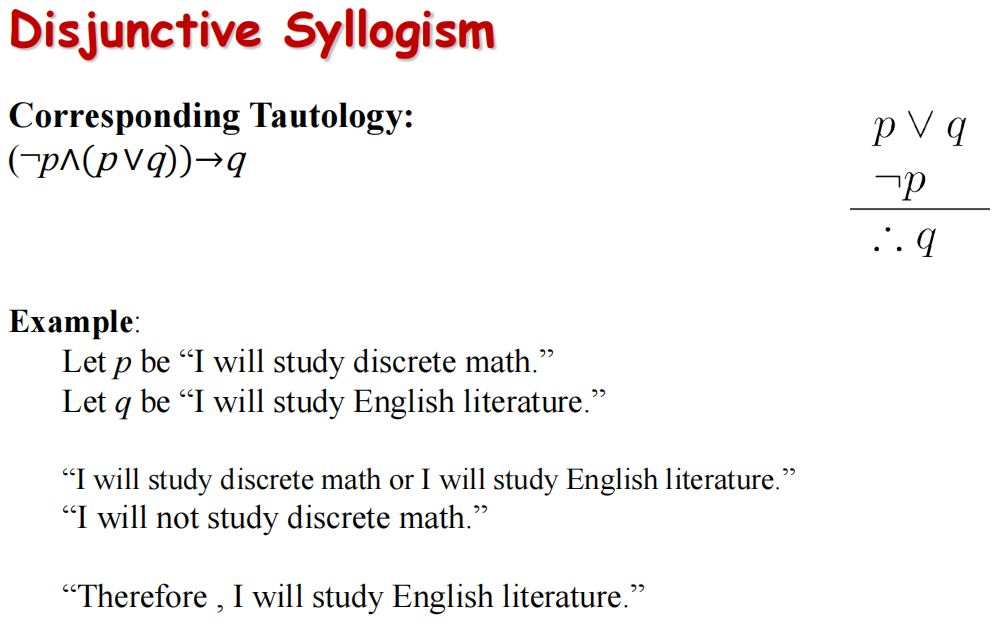
 **Addition （附加）**

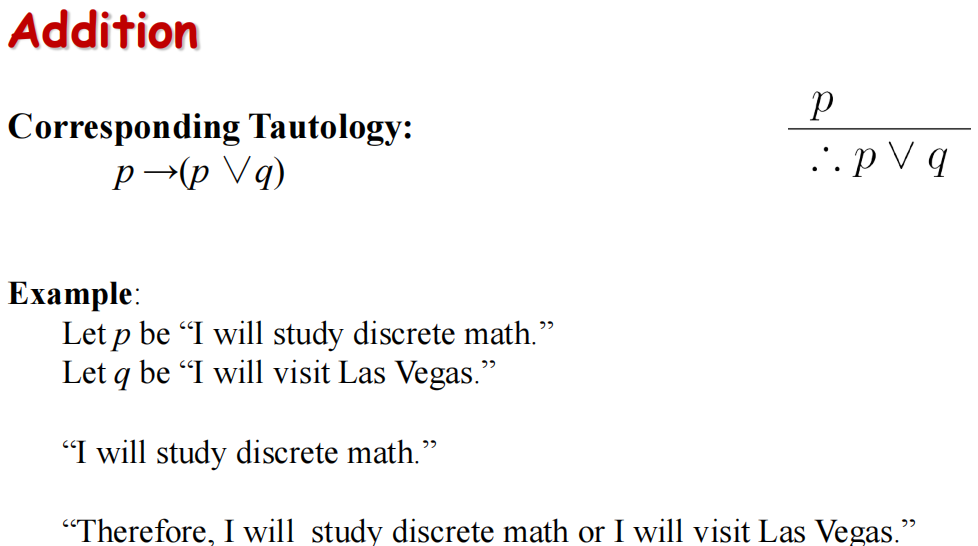
 **……**











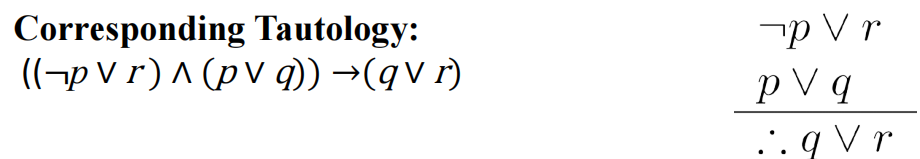
**Simplification（化简）**



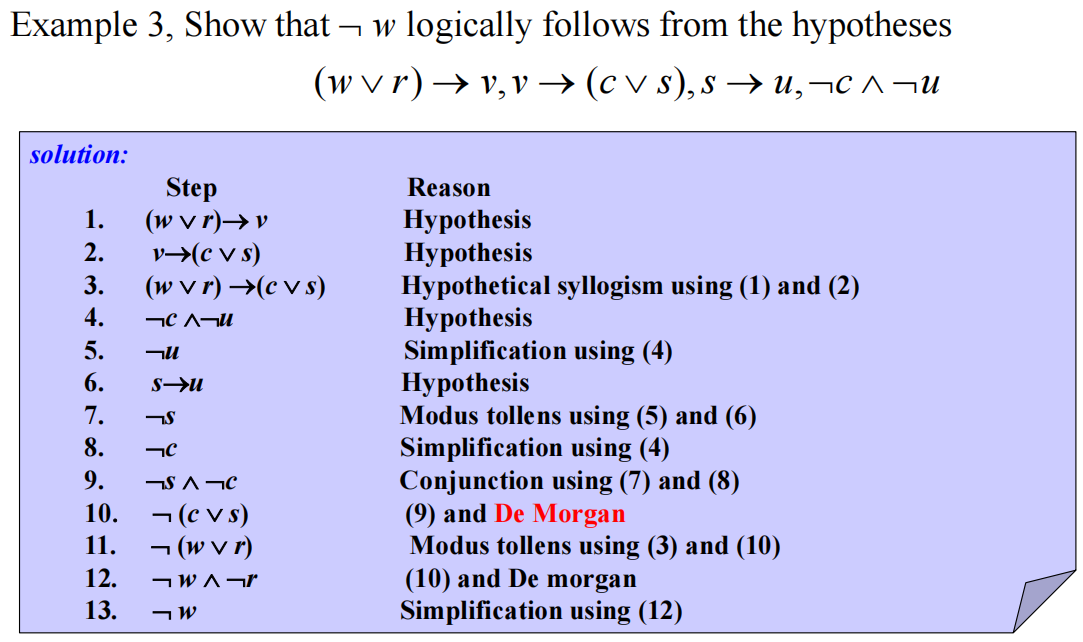
**Conjunction(合取）**



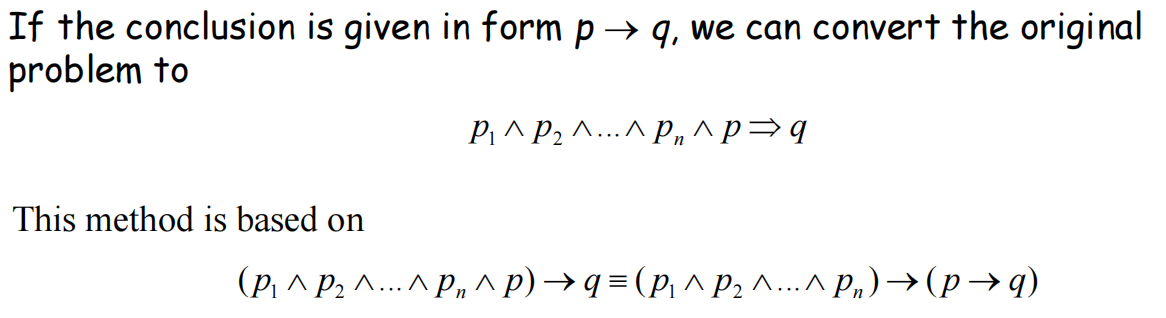
**Resolution（归结）**

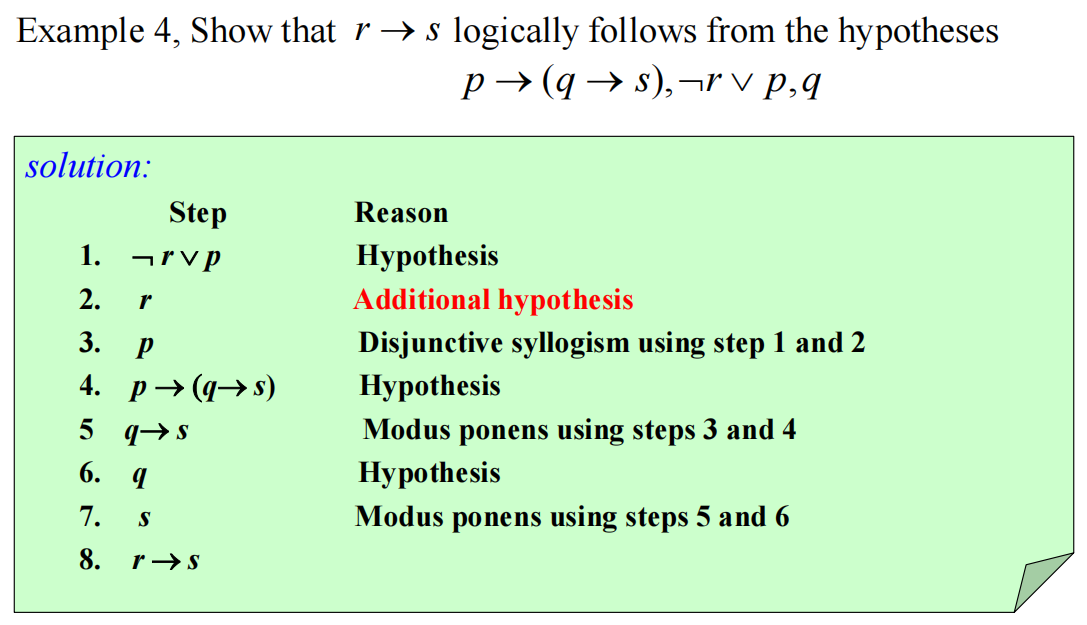


**complex example**



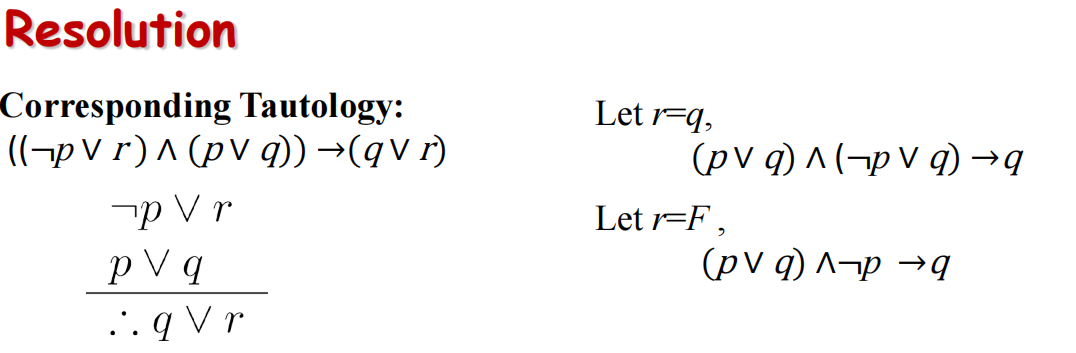
一些解题技巧





（**Additional hypothesis：附加前提**）

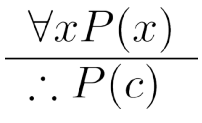
**Resolution（归结的两个推论）**



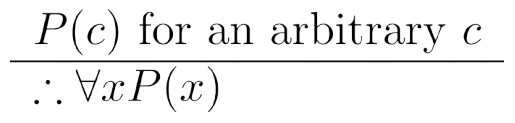


**Rules of Inference for Quantified Statements**

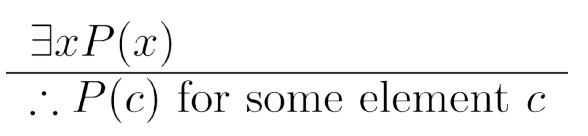
1. **Universal Instantiation (UI) 全称指定**



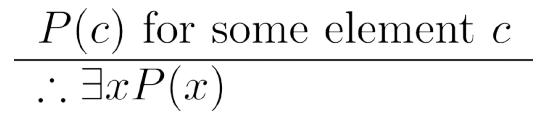
1. **Universal Generalization (UG) 全称推广**



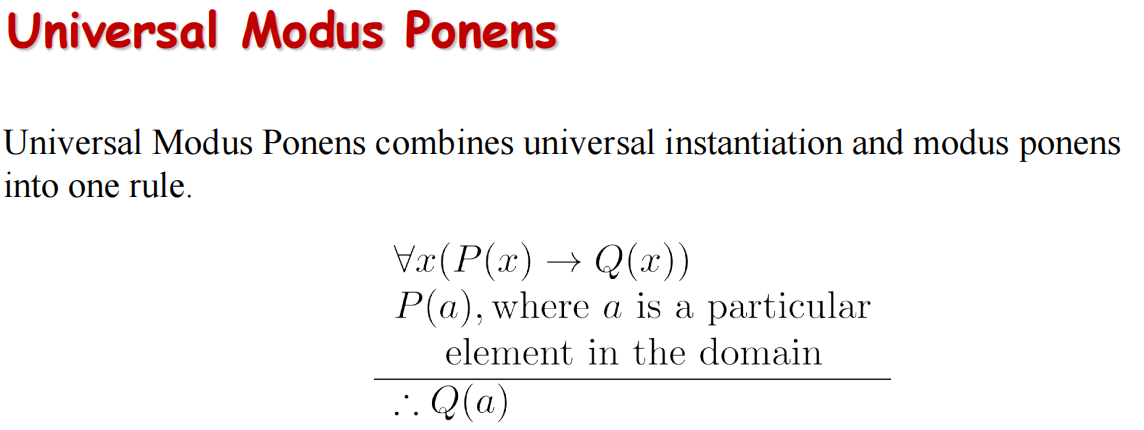
1. **Existential Instantiation (EI) 存在指定**

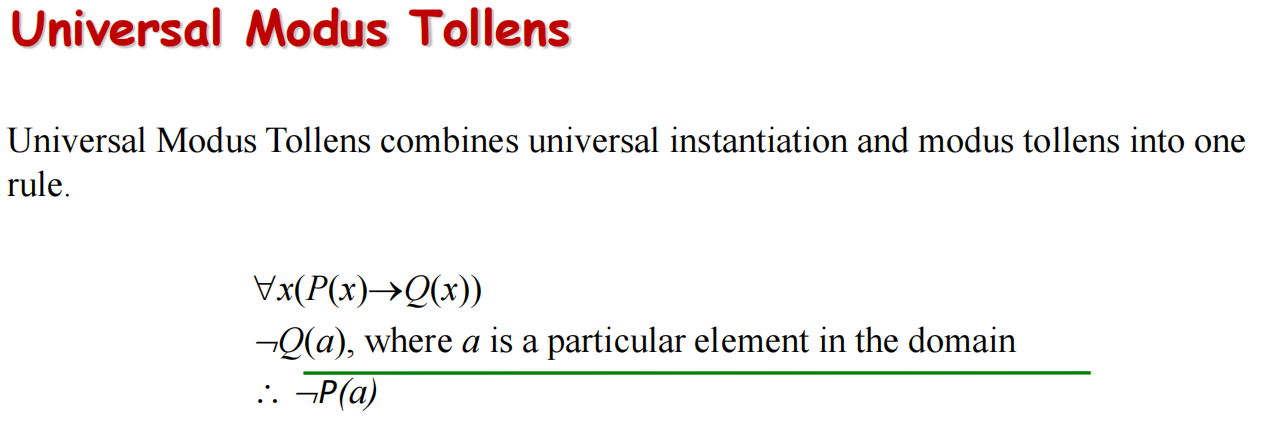


1. **Existential Generalization (EG) 存在推广**



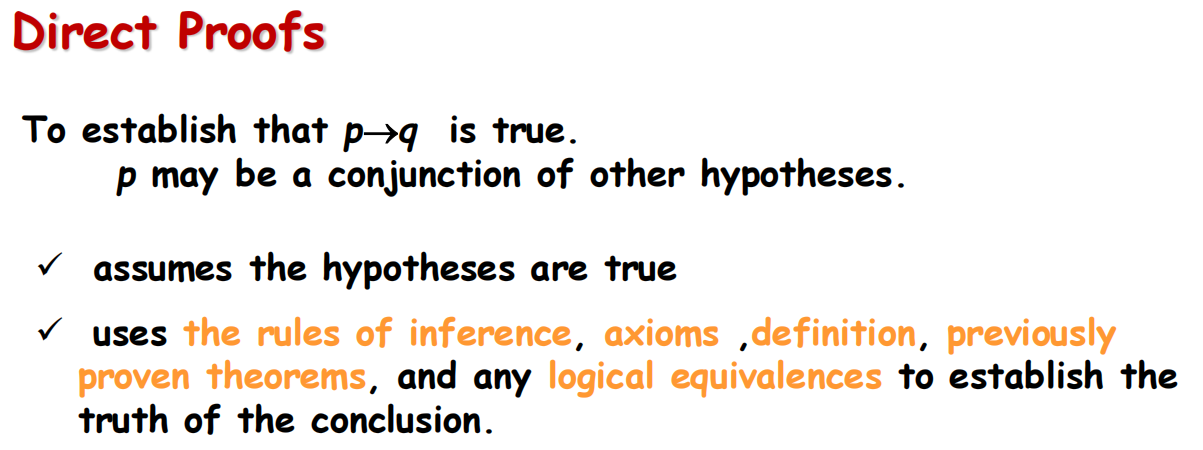
**一些推论**



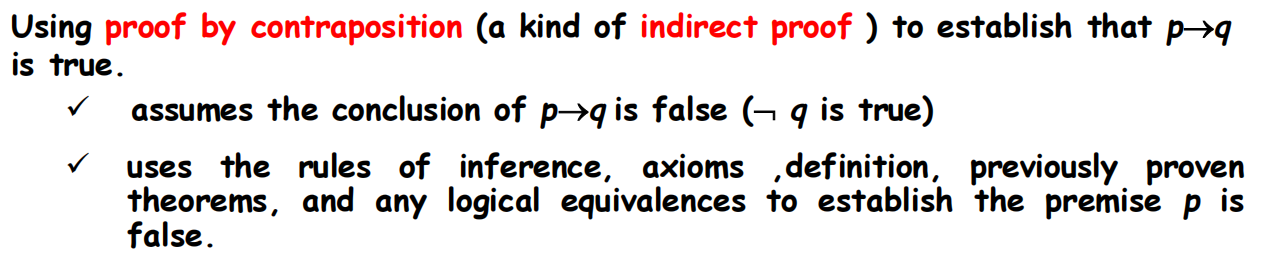


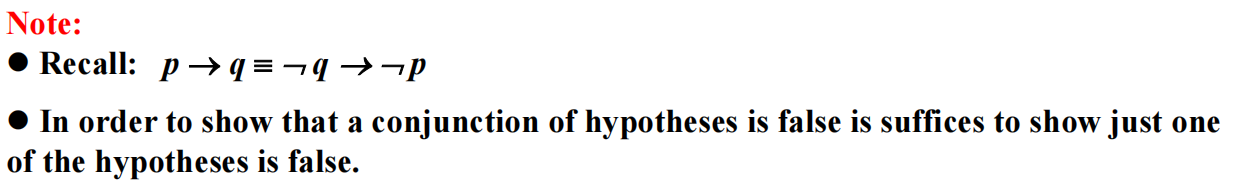
**（1.7）**

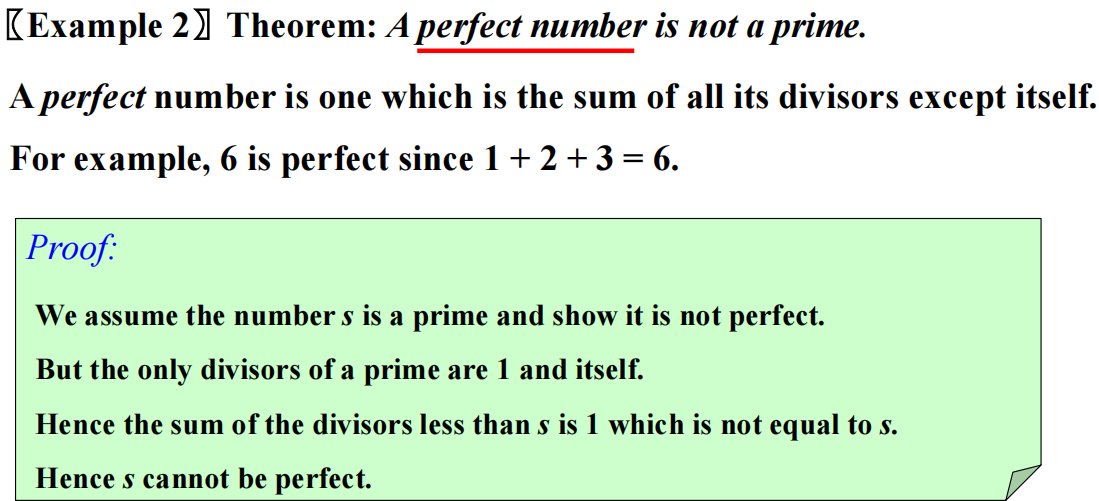
**Axioms（公理）Lemma（引理）Corollary（推论）Conjecture（假设）**



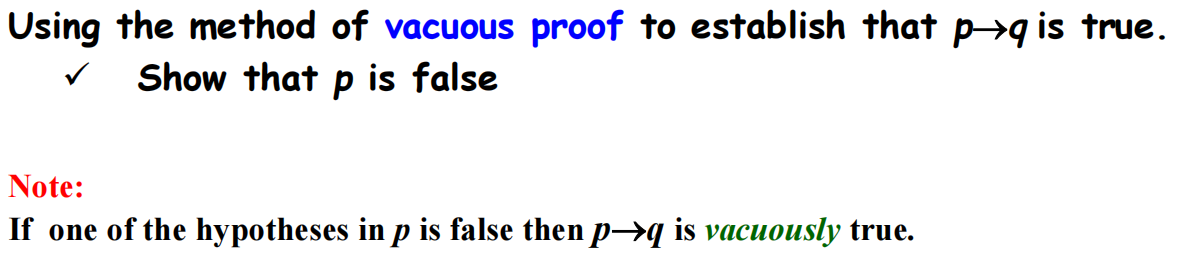
**Proof by Contraposition（反证法）**



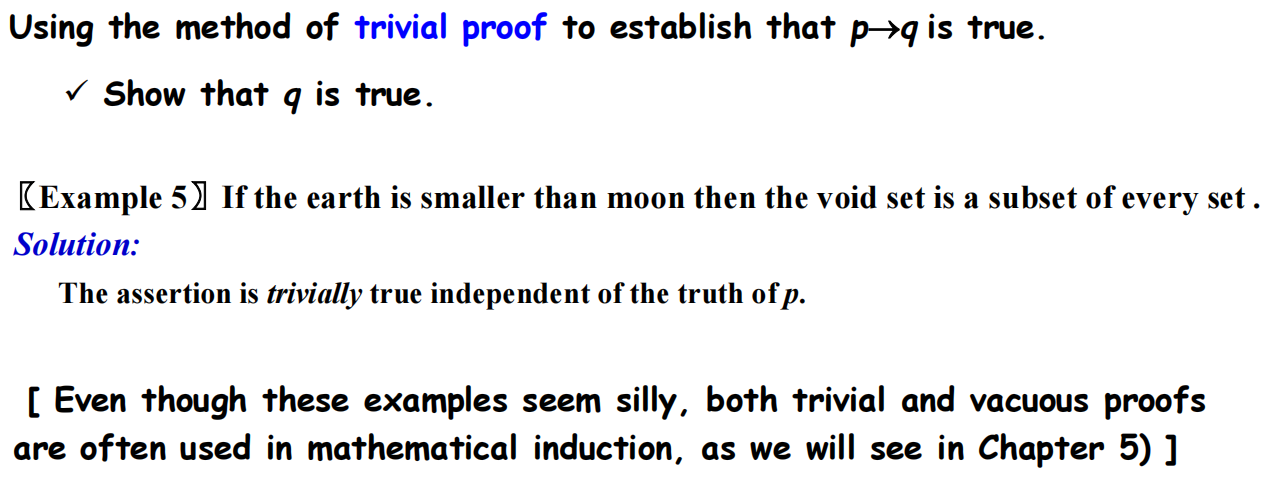




**Vacuous Proof （空证明）**

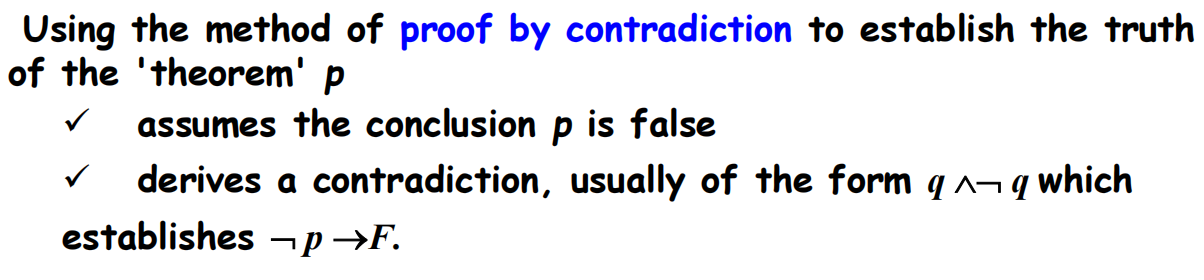


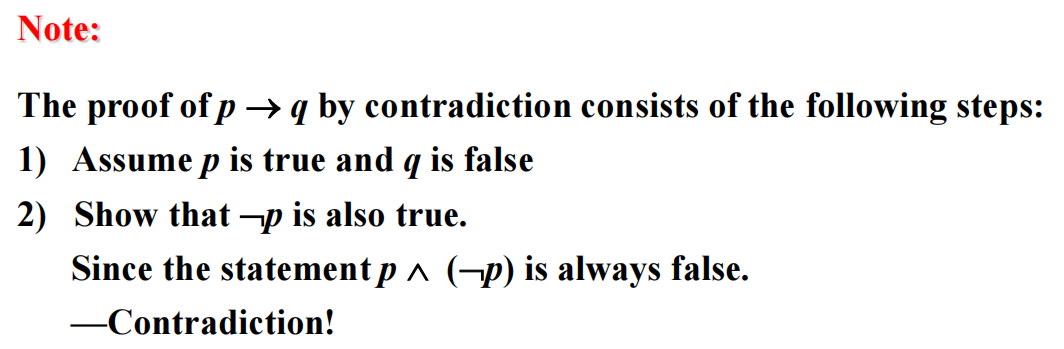
**Trivial proof （平凡证明）**

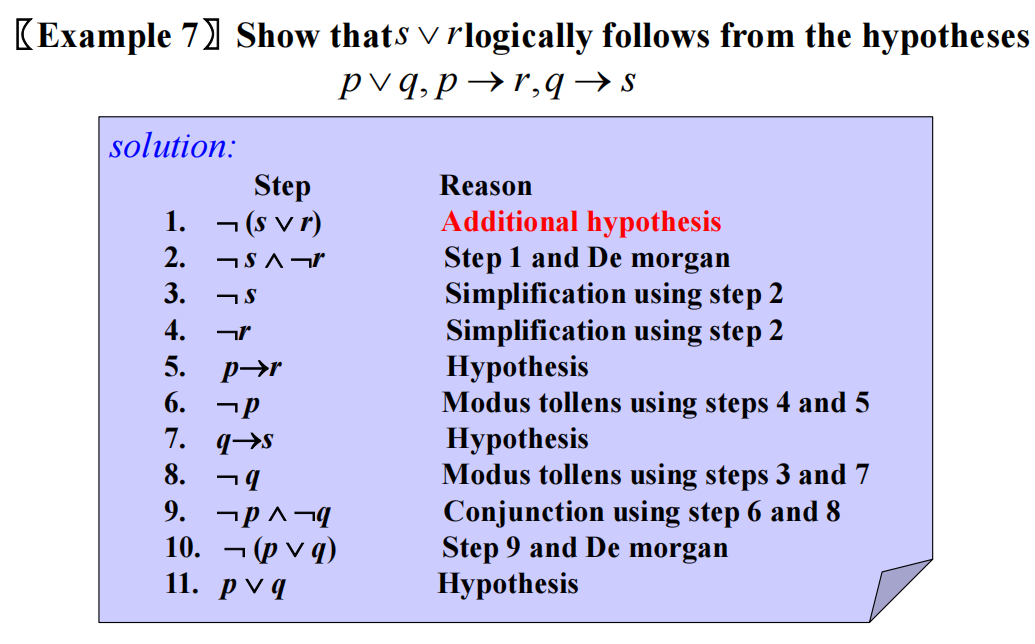


**Void set(空集) subset(子集)**

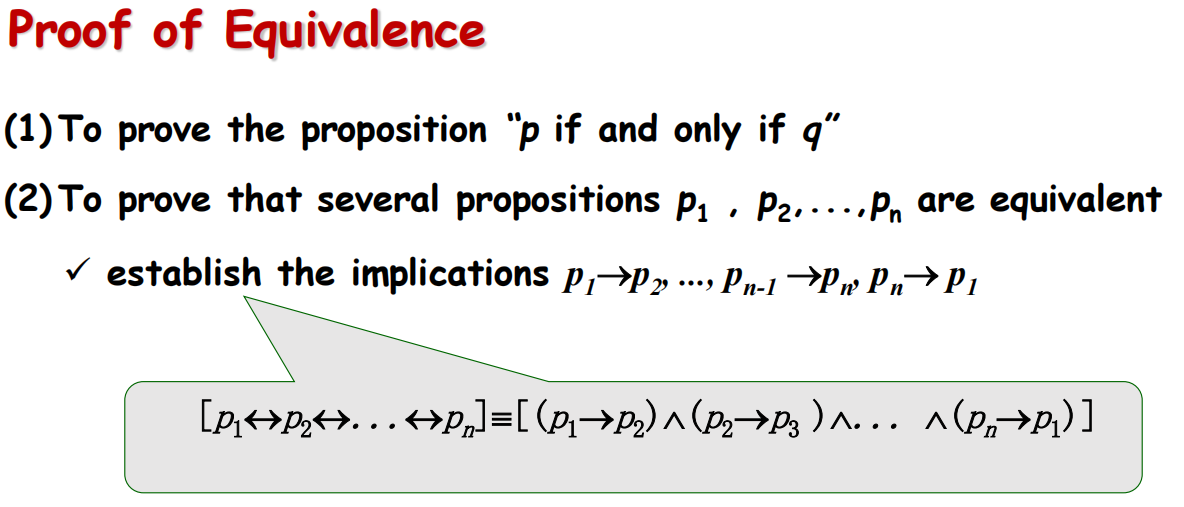
**Proof by contradiction（归谬法）**





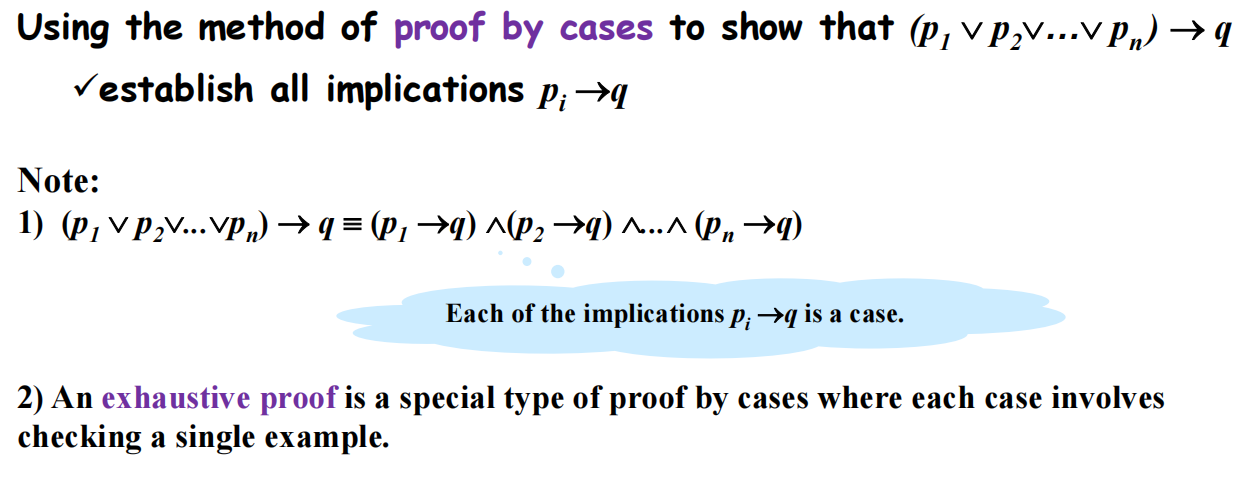


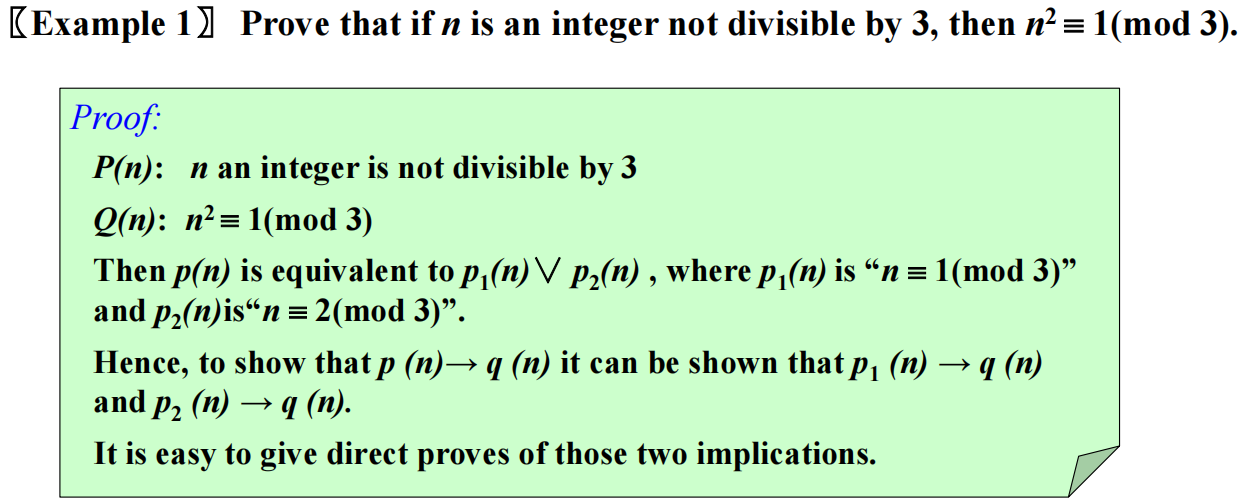
（推出某个前提为假）



**（1.8）**

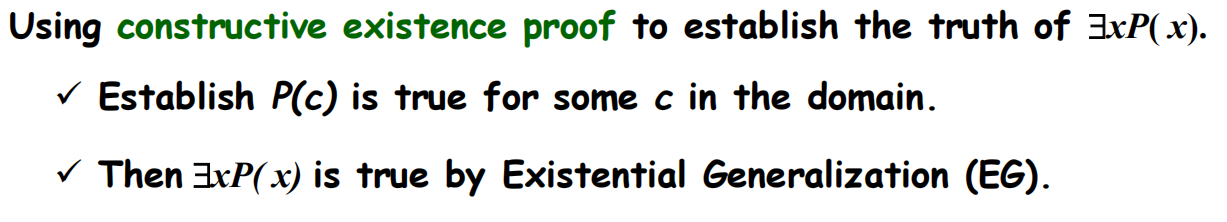
**Exhaustive Proof（穷举证明） and Proof by Cases （分情况证明）**



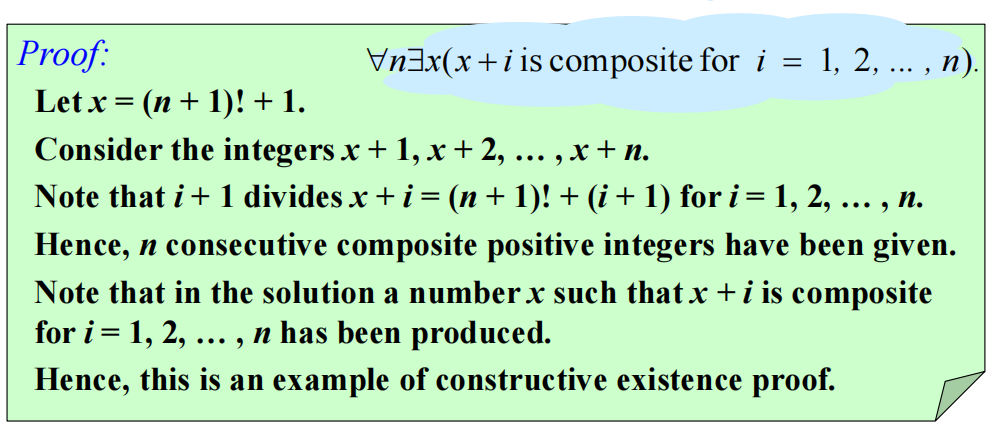


**Existence Proofs （存在性证明）**

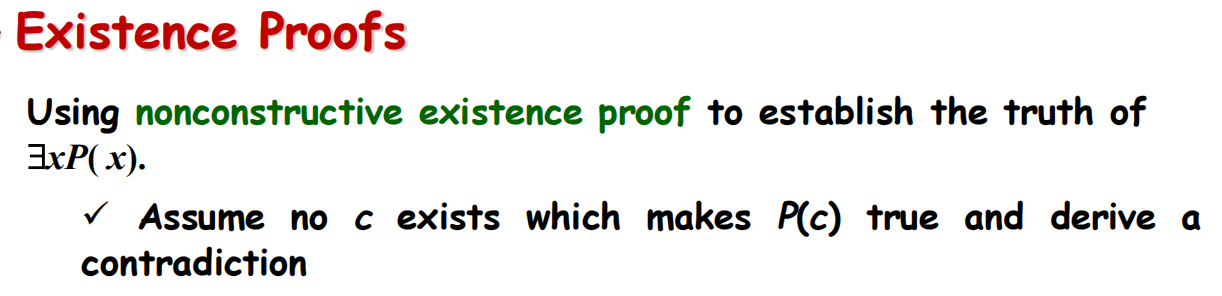
**（构造性证明）**



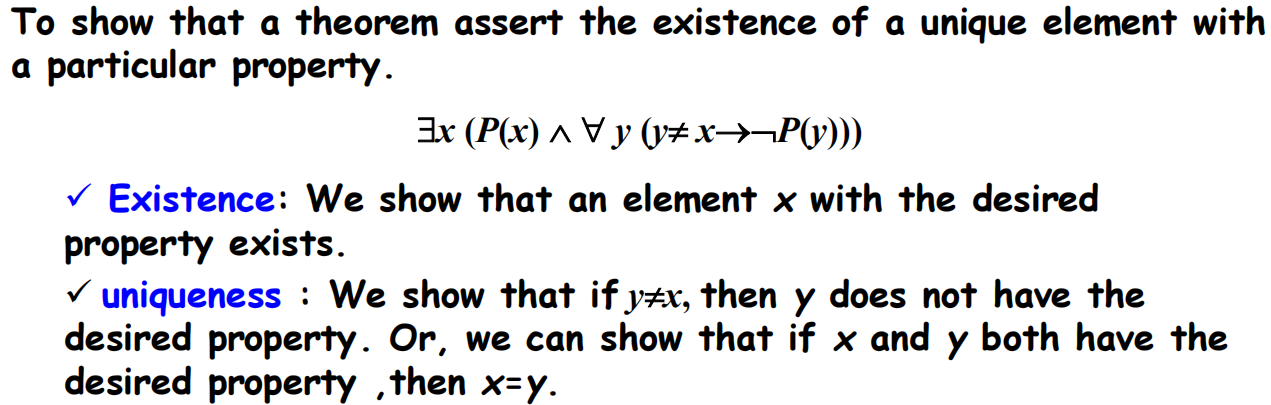
**〖Example 2〗 Show that there are *n* consecutive composite positive integers（连续正合数） for every positive integer *n*.**



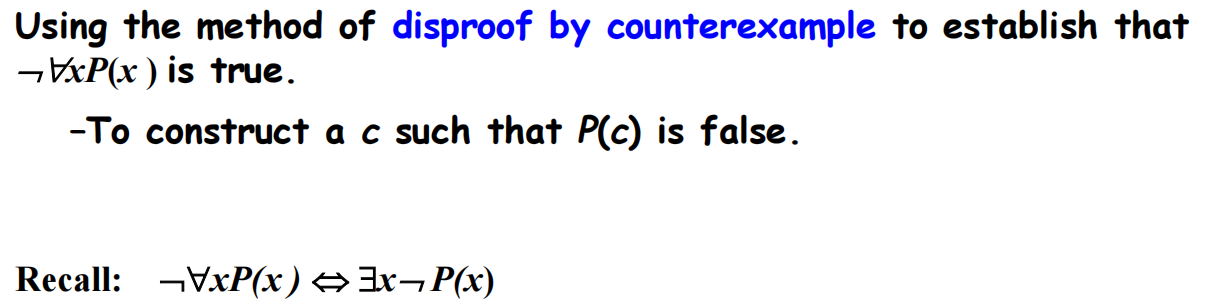
**（非构造性证明）**



**Uniqueness Proofs （唯一性证明）**



**Disproof by Counterexample（举反例）**



**Nonexistence Proofs（不存在性证明）**

