Discrete Math

Contents

Part I Discrete Math: Logic

Chapter I Propositional Logic

§ 1.1 Connectives and Truth Assingments

Definition 1.1.1 (Truth table of Connectives) (Omitted)

Definition 1.1.2 (Truth Assingments) Suppose Σ is the set of propositional variables. A mapping from Σ to $\{\mathbf{T}, \mathbf{F}\}$ called a truth assignment.

Definition 1.1.3 Suppose Σ is the set of propositional variables and $\mathcal{J} : \Sigma \to \{\mathbf{T}, \mathbf{F}\}$ is a truth assignment. The truth value of the compond proposition on $\mathcal{J}$

…

(Omitted)

Definition 1.1.4 (Tautology, contradiction) (Omitted)

Definition 1.1.5 (Contingency, Satisfiable) A contingency is a compound proposition that is neither a tautology nor a contradiction.

A compound proposition is satisfiable if it is true under some truth assignment.

§ 1.2 Consequence and Equivalent

1 The definition of consequence and logically equivalent

Definition 1.2.1 (Consequence) Suppose Φ is a set of propositions and ψ is one single proposition. We say that ψ is a consequence of Φ, written as $\Phi \models \psi$. if Φ 's being all true implies that ψ is also true.

In other words, $\Phi \models \psi$ if for any truth assignment $\mathcal{J}$, $[\![\phi]\!]_{\mathcal{J}} = \mathbf{T}$ for any $\phi \in \Phi$

implies $[\![\psi]\!]_{\mathcal{J}} = \mathbf{T}$.

Definition 1.2.2 (Logically Equivalent) ϕ is a logically equivalent to ψ, written as $\phi \equiv \psi$, if ϕ 's truth value and ψ 's truth value are the same under any situation. In other words, $\phi \equiv \psi$ if $[\![\phi]\!]_{\mathcal{J}} = [\![\psi]\!]_{\mathcal{J}}$ for any truth assignment $\mathcal{J}$.

Example 1.2.1 $\Phi = \{ \}, \psi = p \vee \neg p, \Phi \models \psi$

2 Important properties

Theorem 1.2.1

- $\phi \vee \neg\phi$ is an tautology
- $\phi \wedge \neg\phi$ is a contradiction
- $\phi, \psi \models \phi \wedge \psi$ ($\wedge$-Introduction)
- $\phi \wedge \psi \models \phi$ ($\wedge$-Elimination)
- $\phi \models \phi \vee \psi$ ($\vee$-Introduction)
- If $\Phi, \phi_1 \models \psi, \Phi, \phi_2 \models \psi$, then $\Phi, \phi_1 \vee \phi_2 \models \psi$ ($\vee$-Elimination)

Proof (Proof of the last one) Suppose $[\![\phi]\!]_{\mathcal{J}} = \mathbf{T}, [\![\phi_1 \vee \phi_2]\!]_{\mathcal{J}} = \mathbf{T}$. Then at least one of the following holds: $[\![\phi_1]\!]_{\mathcal{J}} = \mathbf{T}, [\![\phi_2]\!]_{\mathcal{J}} = \mathbf{T}$. □

Theorem 1.2.2 (Contrapositive) If $\Phi, \neg\phi \models \psi$, then $\Phi, \neg\psi \models \phi$

Theorem 1.2.3

- $\neg(\neg q) \equiv q$ (Double Negation)
- $\phi \wedge \phi \equiv \phi, \quad \phi \vee \phi \equiv \phi$ (Idempotent Laws)
- $\phi \wedge \psi \equiv \psi \wedge \psi, \quad \phi \vee \psi \equiv \psi \vee \psi$ (Commutative Laws)

- $\phi \vee (\psi \wedge \chi) \equiv (\phi \vee \psi) \wedge (\phi \vee \chi), \quad \phi \wedge (\psi \vee \chi) \equiv (\phi \wedge \psi) \vee (\phi \wedge \chi)$ (Distributive Laws)
- $\neg(q \wedge q) \equiv \neg p \vee \neg q, \quad \neg(q \vee q) \equiv \neg p \wedge \neg q$ (De Morgan's Laws)
- $\phi \wedge (\neg\phi) \equiv \mathbf{F}, \quad \phi \vee (\neg\phi) \equiv \mathbf{T}$ (Negation Laws)
- $\phi \wedge \mathbf{T} \equiv \phi, \quad \phi \vee \mathbf{F} \equiv \phi, \quad \phi \wedge \mathbf{F} \equiv \mathbf{F}, \quad \phi \vee \mathbf{T} \equiv \mathbf{T}$ (Laws of logical constants)
- $\phi \vee (\phi \wedge \psi) \equiv \phi, \quad \phi \wedge (\phi \vee \psi) \equiv \phi$ (Absorption Laws)

3 Prove Logical Equivalence

Theorem 1.2.4 (Transitivity) If $\phi \equiv \psi$ and $\psi \equiv \chi$, then $\phi \equiv \chi$.

Theorem 1.2.5 (Congruence Property)

- If $\phi \equiv \psi$, then $\neg\phi \equiv \neg\psi$
- If $\phi_1 \equiv \phi_2, \psi_1 \equiv \psi_2$, then $\phi_1 \wedge \psi_1 \equiv \phi_2 \wedge \psi_2$
- If $\phi_1 \equiv \phi_2, \psi_1 \equiv \psi_2$, then $\phi_1 \vee \psi_1 \equiv \phi_2 \vee \psi_2$

Theorem 1.2.6 (Reflexivity) $\phi \equiv \phi$

4 Relation among tautologies, contradictions, satisfiable assertions, consequence relations and logic equivalence

Theorem 1.2.7

- $\phi_1, \phi_2, \cdots \phi_n \models \psi$ iff. $\left(\bigwedge_{k=1}^{n}\right) \wedge \neg\psi$ is not satisfiable.
- $\{\ \} \models \phi$ iff. ϕ is an tautology.

- $\phi \equiv \psi$ iff. $\phi \models \psi$ and $\psi \models \phi$.

Theorem 1.2.8 If $\phi \models \psi$ and $\psi \models \chi$, then $\phi \models \chi$.

§ 1.3 Normal Forms

Definition 1.3.1 (Disjunctive Normal Form, DNF)

- A **literal** is a propositional variable or its negation.
- A **conjunctive clause** is a conjunctions of literals.
- A **compound proposition** is in disjunctive normal form if it is a disjunction of conjunctive clauses.

Definition 1.3.2 (Conjunctive Normal Form, CNF)
(Similar as above)

Example 1.3.1

- literals $x, y, z, p, q, r, \neg q$
- conjunctive clauses $p, p \wedge q, \neg p \wedge q$
- DNF $p, p \vee (\neg q \wedge r), \neg p \vee (q \wedge p \wedge r)$

Theorem 1.3.1 Every compound proposition is logically equivalent to some compound proposition in DNF.

Proof (Proof 1) Suppose that the compound proposition ϕ consists of the literals $p_1, p_2, \cdots, p_n$.
For all $\mathcal{J}$ as a interpretation, we only need to prove that

$$\phi \equiv \bigvee_{[\![\phi]\!]_{\mathcal{J}}=\mathbf{T}} \left(\bigwedge_{\mathcal{J}(p_i)=\mathbf{T}} p_i \wedge \bigwedge_{\mathcal{J}(p_i)=\mathbf{F}} \neg p_i \right) = \mathbf{T}$$

Consider a specific interpretation $\mathcal{J}_0$, if $[\![\phi]\!]_{\mathcal{J}} = \mathbf{T}$, then

$$\left[\!\!\left[\bigvee_{[\![\phi]\!]_{\mathcal{J}}=\mathbf{T}} \left(\bigwedge_{\mathcal{J}(p_i)=\mathbf{T}} p_i \wedge \bigwedge_{\mathcal{J}(p_i)=\mathbf{F}} \neg p_i\right)\right]\!\!\right]_{\mathcal{J}_0} = \left[\!\!\left[\left(\bigwedge_{\mathcal{J}_0(p_i)=\mathbf{T}} p_i \wedge \bigwedge_{\mathcal{J}_0(p_i)=\mathbf{F}} \neg p_i\right)\right]\!\!\right]_{\mathcal{J}_0}$$

If $\mathcal{J}_0(p_i) = \mathbf{T}$, then $[\![p_i]\!]_{\mathcal{J}_\prime} = \mathbf{T}$,
if $\mathcal{J}_0(p_i) = \mathbf{F}$, then $[\![\neg p_i]\!]_{\mathcal{J}_\prime} = \mathbf{T}$.
So

$$\left[\!\!\left[\left(\bigwedge_{\mathcal{J}_0(p_i)=\mathbf{T}} p_i \wedge \bigwedge_{\mathcal{J}_0(p_i)=\mathbf{F}} \neg p_i\right)\right]\!\!\right]_{\mathcal{J}_0} = \mathbf{T}$$

□

Proof (Proof 2) Define $DNF(\phi)$ as follow and prove that $DNF(\phi) \equiv \phi$.

Definition 1.3.3

- $DNF(\phi) \triangleq DNF_2(DNF_1(\phi))$

- $DNF_1(\neg\neg\phi) = DNF_1(\phi)$.
  (The De Morgan's law)
  $DNF_1(\phi \wedge \psi) = DNF_1(\phi) \wedge DNF_1(\psi)$ ($\vee$ is the same)
  $DNF_1(l) = l$ l is a literal.

- $DNF_2(l) = l$ l is a literal,
  $DNF_2(\phi \vee \psi) = DNF_2(\phi) \vee DNF_2(\psi)$
  If $\phi = \bigvee_{i=1}^{n} \phi_i, \psi = \bigvee_{j=1}^{m} \psi_j$, then

$$DNF_2(\phi \wedge \psi) = \bigvee_{i=1}^{n} \bigvee_{j=1}^{m} (\phi_i \wedge \psi_j)$$

Then it's obvious that $\phi \equiv DNF(\phi)$ and $DNF(\phi)$ is a DNF. □

Theorem 1.3.2 Every compound proposition is logically equivalent to some compound proposition in CNF.

Proof (Similar as above)

Example 1.3.2 (*) The CDCL algorithm.
(Suspended now)

Chapter II First Order Logic, FOL

§ 2.1 The syntax of first order language

Definition 2.1.1

- Predicate Logic's Language
  - Variables $x, y, z, \cdots$
  - Constants $c_1, c_2, \cdots$
  - Prelicates $P, Q, R, \cdots$
  - Functions $f, g, h, \cdots$
  - Logic patterns $\exists, \forall, \wedge, \vee, \neg$
- Terms $x, y, c_1, c_2, f(x), g(x, y), \cdots$
- propositions $P(x), Q(f(x, g(x, y))), \exists x \forall y R(x, g(y)), \cdots$

§ 2.2 The semantics of first order language

1 Structure

Definition 2.2.1 (S-structure)
Given a sumbol set S, an S-structure $\mathcal{A} = (A, \alpha)$ contains

- a domain A, which is a non-empty set.
- an interpretation of every predicate symbol.
  Example 2.2.1 if P is a symbol of binary predicate, then $\alpha(P)$ is a mapping from $A \times A$ to $\{\mathbf{T}, \mathbf{F}\}$.

- an interpretation of every function symbol.

  Example 2.2.2 if f is a symbol of unary function, then $\alpha(f)$ is a mapping from A to A.

- an interpretation of every constant symbol.

  Example 2.2.3 if s is a constant symbol, $\alpha(c)$ is an element in domain A.

With a structure, we can determine the truth of an closed proposition.

2 Interpretation

Definition 2.2.2 (S-interpretation)
Given a symbol set S, a S-interpretation $\mathcal{J} = (\mathcal{A}, \beta)$ is

- a S-structure $\mathcal{A} = (A, \alpha)$
- a S-assignment β: a mapping from variables to elements in the domain A

For $\mathcal{J} = (\mathcal{A}, \beta)$ and $\mathcal{A} = (A, \alpha)$, we usually use $\mathcal{J}(P)$ and $\mathcal{A}(P)$ to represent $\alpha(P)$, use $\mathcal{J}(f)$ and $\mathcal{A}(f)$ to represent $\alpha(f)$, use $\mathcal{J}(c)$ and $\mathcal{A}(c)$ to represent $\alpha(c)$, and use $\mathcal{J}(x)$ to represent $\beta(x)$.

Definition 2.2.3 (Terms' denotation)
For S-interpretation $\mathcal{J}$ and a S-term t,

- $[\![x]\!]_{\mathcal{J}} = \mathcal{J}(x)$
- $[\![c]\!]_{\mathcal{J}} = \mathcal{J}(c)$
- $[\![f(t_1, t_2, \ldots, t_n)]\!]_{\mathcal{J}} = \mathcal{J}(f)\big([\![t_1]\!]_{\mathcal{J}}, [\![t_2]\!]_{\mathcal{J}}, \ldots, [\![t_n]\!]_{\mathcal{J}}\big)$

Definition 2.2.4 (Propositions' truth)
For S-interpretation $\mathcal{J}$ and a S-proposition t,

- $[\![P(t_1, t_2, \ldots, t_n)]\!]_{\mathcal{J}} = \mathcal{J}(P)\big([\![t_1]\!]_{\mathcal{J}}, [\![t_2]\!]_{\mathcal{J}}, \ldots, [\![t_n]\!]_{\mathcal{J}}\big)$

- $[\![\varphi \wedge \psi]\!]_{\mathcal{J}} = [\![\wedge]\!]([\![\varphi]\!]_{\mathcal{J}}, [\![\psi]\!]_{\mathcal{J}})$
- $[\![\neg\varphi]\!]_{\mathcal{J}} = [\![\neg]\!]([\![\varphi]\!]_{\mathcal{J}})$
- $[\![\forall x \varphi]\!]_{\mathcal{J}} = \mathbf{T}$ if and only if for every a in $\mathcal{A}$'s domain, $[\![\varphi]\!]_{\mathcal{J}[x \mapsto a]} = \mathbf{T}$
- $[\![\exists x \varphi]\!]_{\mathcal{J}} = \mathbf{T}$ if and only if for at least one a in $\mathcal{A}$'s domain, $[\![\varphi]\!]_{\mathcal{J}[x \mapsto a]} = \mathbf{T}$

where $\mathcal{J}[x \mapsto a]$ is a S-interpretation which keeps all other interpretations in $\mathcal{J}$ and interprets x by a.

§ 2.3 Quantiers with restricted domains

1 The truth of ”if-then”

Theorem 2.3.1

- $\phi \to (\psi \to \phi) \equiv \mathbf{T}$.
- $(\phi \to \psi \to \chi) \to (\phi \to \psi) \to (\phi \to \chi) \equiv \mathbf{T}$.
- $\phi \to \psi \equiv \neg\phi \vee \psi$

Part II Discrete Math: Set Theory

Chapter III The definition of set

(Omitted)

Chapter IV Relations

§ 4.1 Relations

1 Properties of relations

Definition 4.1.1 Given R, a relation on A,

- **Reflexive** on A if it holds that $\forall a \in A, (aRa) \Leftrightarrow I_A \subseteq R$
- **Symmetric** on A if it holds that $\forall a, b \in A$ if aRb, then $bRa \Leftrightarrow R^{-1} = R$
- **Transitive** on A if it holds that $\forall a, b, c \in A$ if aRb, bRc, then $aRc \Leftrightarrow R \circ R \subseteq R$
- **Antisymmetric** on A if it holds that $\forall a, b \in A$ if aRb, bRa, then $a = b \Leftrightarrow R \cap R^{-1} = I_A$

2 Equivalence relations

Definition 4.1.2 If $R \subseteq A \times A$ is reflexive, symmetric and transitive, then R is called a **equivalence relation** on A

§ 4.2 Relations and Sets

1 Equivalence classes and Partitions

Definition 4.2.1 R is an equivalence relation on A, $a \in A$, then we define the equivalence class $[a]_R$ of A by

$$[a]_R = \{b \in A | bRa\}$$

Theorem 4.2.1 aRb iff. $[a]_R = [b]_R$

2 Transitive Closures and Reflexive Transitive Closures

Definition 4.2.2 (Transitive Closures) Suppose R is a relation on A, R' is a transitive closure of R if

- $R \subseteq R'$
- R' is transitive
- $\forall T$, T is transitive, $R \subseteq T$, then $R' \subseteq T$.

Definition 4.2.3 (Another definition) $R^+ = \bigcup\limits_{n=1}^{\infty} R^n$ is the transitive closure

Proof Let's prove that the two definitions are equivalent.

- $R \subseteq R^+$
- If aR^+b, bR^+c, then there exists m, n, $aR^m b$, $bR^n c$, then $aR^{m+n}c$, R^+ is transitive.
- If $R \subseteq T$ and T is transitive, if $R^n \subseteq T$, then $R^{n+1} = R^n \circ R \subseteq T \circ T \subseteq T$, so $R^+ = \bigcup\limits_{n=1}^{\infty} R^n \subseteq T$.

So such R^+ is a transitive closure. □

Chapter V Functions

§ 5.1 Functions

§ 5.2 Funcions and Sets

1 Injection and Surjection

Definition 5.2.1

$F : A \to B$,

- **Injection**(one-to-one map): $\forall a, a' \in A$, if $F(a) = F(a')$, then $a = a'$.
- **Surjection**(onto map): $\forall b \in B, \exists a \in A, F(a) = b$.
- **Bijection**(one-to-one correspondence): both one-to-one and onto.

Theorem 5.2.1

- If F, G are both injections, then $F \circ G$ is also an injection.
- If F, G are both surjection, then $F \circ G$ is also a surjection.
- If $F \circ G$ is an injection, then G is also an injection.
- If F is an bijection, then F^{-1} is also a bijection.

Theorem 5.2.2 (Berstern's Theorem) If there exist an injection from A to B and an injection from B to A, then there exists a bijection between A and B

Proof Suppose F is an one-to-one function from A to B, G is an one-to-one function from B to A.

Then we can construct a sequence of set as follow:

$$C_0 = \{a \in A | \forall b \in B, G(b) \neq a\} = A \setminus \{a | \exists b \in B, G(b) = a\},$$
$$D_0 = \{F(a) | a \in C_0\} = B \setminus \{b \in B | \exists a \in A \setminus C_0, b = F(a)\}$$

$\forall n \geqslant 1$,

$$C_n = \left\{ a \in A | \forall b \in B \setminus \bigcup_{i=0}^{n-1} D_i, G(b) \neq a \right\}$$
$$D_n = \{F(a) | a \in C_n\}$$

Now we define a function H, where

$$H(a)=\begin{cases} F(a), & a\in\bigcup\limits_{n=0}^{\infty}C_n \\ b\ (a=G(b)), & a\notin\bigcup\limits_{n=0}^{\infty}C_n \end{cases}$$

Let $C=A\setminus\bigcup\limits_{n=0}^{\infty}C_n$, $D=B\setminus\bigcup\limits_{n=0}^{\infty}D_n$

Now we prove that H is well-defined and is a bijection.

- Firstly we prove that such b exists.

  $\forall a\in C, a\notin C_0$, so $\exists b\in B$, $G(b)=a$. If $b\in D_n$, then $a=G(b)\in C_{n+1}$, contradictive! So $b\in D$. Due to G is an injection, such b is unique.

- Then we prove that H is an injection.

  $\forall a\in\bigcup\limits_{n=0}^{\infty}C_n, F(a)\in\bigcup\limits_{n=0}^{\infty}D_n$, and due to F is an injection on $\bigcup\limits_{n=0}^{\infty}C_n$, H is an injection.

  $\forall a\in C, \exists b\in D, a=G(b)$, due to G is an injection on C, H is an injection.

- Finallty we prove that H is a surjection.

  $\forall b\in\bigcup\limits_{n=0}^{\infty}D_n$ according to the define.

  $\forall b\in D, \exists a\in A, G(b)=a$, so $a\notin C_0$. If $a\in C_n(n\geqslant 1)$, then $b\in D_{n-1}$, contradictive! So $a\in C$.

□

2 Equinumerous Sets

Definition 5.2.2

- If there exists an injection from A to B, then we write $A\preccurlyeq B$.

- If there exists a bijection between A, B, then we call A, B are equinumerous, i.e. $A \approx B$

Definition 5.2.3 Denote the set of function (or its cardinality) $\{F \mid F: A \to B\}$ by B^A

Theorem 5.2.3 $\mathcal{P}(A) \approx \{F \mid F: A \to \{0,1\}\}$

Proof Let function $H: \mathcal{P}(A) \to \{F \mid F: A \to \{0,1\}\}$,
$\forall X \in \mathcal{P}(A), H(X)(a) = 1$ iff. $a \in X$.
For any $F \in \{F \mid F: A \to \{0,1\}\}$, $X = \{a \mid F(a) = 1\} \in \mathcal{P}(A)$, $H(X) = F$.
If $H(X_1) = H(X_2) = F$, then $X_1 = X_2 = \{a \mid F(a) = 1\}$. □

Theorem 5.2.4 If $A_1 \approx A_2, B_1 \approx B_2$, then $(A_1 \to B_1) \approx (A_2 \to B_2)$, i.e. $B_1^{A_1} \approx B_2^{A_2}$

Proof There exist $f \in (A_1 \to A_2), g \in (B_1 \to B_2)$, f, g are both bijections.
Then let $H: (A_1 \to B_1) \to (A_2 \to B_2)$, for any $F: A_1 \to B_1$, $H(F) = g \circ F \circ f^{-1}$
$H(F_1) = H(F_2) \Rightarrow g \circ F_1 \circ f^{-1} = g \circ F_2 \circ f^{-1} \Rightarrow F_1 \circ f^{-1} = F_2 \circ f^{-1}$. According to $\forall b \in A_2, \exists a \in A_1, f(a) = b$. So $F_1 \circ f^{-1} = F_2 \circ f^{-1} \Rightarrow \forall b \in A_2, F_1 \circ f^{-1}(b) = F_2 \circ f^{-1}(b) \Rightarrow F_1(a) = F_2(a) \Rightarrow F_1 = F_2$.
$\forall F_2 \in (A_2 \to B_2)$, let $F_1 = g^{-1} \circ F_2 \circ f$. □

Theorem 5.2.5 $(A \times B \to C) \approx (A \to (B \to C))$, i.e. $C^{A \times B} \approx \left(C^B\right)^A$

Proof Let $H: (A \times B \to C) \to (A \to (B \to C))$, $H(F)(a)(b) = F(a,b)$.
Omit the following proof. □

Theorem 5.2.6 (Cantor's Theorem) $\mathcal{P}(A)$'s cardinality is bigger than A's.

Proof Prove by contradiction.

Assume that exists A, $\mathcal{P}(A) \approx A$, then there exists an bijection θ from A to $\mathcal{P}(A)$.
Let $X = \{x \in A \mid x \in \theta(x)\} \subseteq A$.
Consider $x = \theta^{-1}(X)$.

- If $x \in \theta(x) = X$, then according to the definition of X, $x \notin X$, impossiable!
- If $x \notin \theta(x) = X$, then according to the definition of X, $x \in X$, impossiable!

□

3 Countable Infinity and Uncountable Infinity

Example 5.2.1

- $\mathbb{N}$, $\mathbb{N} \times \mathbb{N}$, $\underbrace{\mathbb{N} \times \cdots \times \mathbb{N}}_{n \text{ times}}$ is countable.
- The set of all finit sequence of $\mathbb{N}$ is countable.
  (equal to $\bigcup\limits_{n=1}^{+\infty} \underbrace{\mathbb{N} \times \cdots \times \mathbb{N}}_{n \text{ times}}$)
- $\mathbb{Q}$ is countable.
  $\mathbb{Q} \preccurlyeq \mathbb{Z}^+ \times \mathbb{Z} \approx \mathbb{N} \times \mathbb{N} \approx \mathbb{N}$, $\mathbb{N} \preccurlyeq \mathbb{Q}$

Example 5.2.2

- $2^{\mathbb{N}} \approx \mathbb{N}^{\mathbb{N}}$
- $\mathbb{R} \approx 2^{\mathbb{N}}$
- $\mathbb{R}^{\mathbb{R}} \approx 2^{\mathbb{R}} \approx \mathcal{P}(\mathbb{R})$

§ 5.3 ZFC Set Theory

1 The Definition of ”=”

Definition 5.3.1 Assembling a prelicate.

- (Axiom of reflexivity) $\forall x(x = x)$
- (Axiom of symmetry) (Omitted)
- (Axiom of transitivity) (Omitted)
- (Axiom of substitution) $\forall a \forall b(a = b \to (\phi[x \mapsto a] \to \phi[x \mapsto b]))$

2 The Axioms of ZFC Set Theory

Theorem 5.3.1

- (Axiom of Extension) $\forall A \forall B(A = B \Leftrightarrow \forall x(x \in A \leftrightarrow x \in B))$
- (Axiom of Union) $\forall \mathcal{A} \exists B \forall x(x \in B \leftrightarrow \exists C(C \in \mathcal{A} \wedge x \in C))$, we denote B as $\bigcup \mathcal{A}$
- (Axiom of Power Set) $\forall A \exists \mathcal{B} \forall C(C \in \mathcal{B} \leftrightarrow C \subseteq A)$, we denote $\mathcal{B}$ as $\mathcal{P}(A)$
- (Axiom of Empty Set) $\exists X \forall x(\neg x \in X)$, we denote such X as $\varnothing$
- (Axiom of Infinity) $\exists X(\varnothing \in X \wedge \forall y(y \in X \to y \cup \{y\} \in X))$, we call such X **inducive set**.
- (Axiom Schema of Specification) $\forall A \exists B \forall x(x \in B \leftrightarrow (x \in A \wedge \phi(x)))$, we denote such B as $\{x \in A \mid \phi(x)\}$
- (Axiom of Regularity) $\forall A \exists y(y \in A \wedge y \cap A = \varnothing) \Leftrightarrow \forall A \exists y(y \in A \wedge \forall x(x \in A \to \neg x \in y))$

3 The Re-definition of Certain Concepts with ZFC

Definition 5.3.2 (The definition of nature numbers)

$0\ :\ \varnothing$

$1\ :\ 0 \cup \{0\}$

$2\ :\ 1 \cup \{1\}$

$\ldots$

We define $\mathbb{N}$ as the smallest inducive set, i.e. for any inducive set T, $\mathbb{N} \subseteq T$. Obviously all the numbers we defined w is the elements of $\mathbb{N}$.

Definition 5.3.3 (The definition of ordered pairs)

We define (a, b) as $\{\{a\}, \{a, b\}\}$.

Definition 5.3.4 (The options of nature numbers) The sum of $m, n \in \mathbb{N}$ is r iff. $(m, n, r) \in T$ where T is the least set such that

$$\forall n, \quad (n, 0, n) \in T$$

$$\forall n \forall m \forall r\, ((n, m, r) \in T \to (n, m \cup \{m\}, r \cup \{r\}) \in T)$$

Definition 5.3.5 (Define transitive closures with ZFC) For any $R \subseteq A \times A$, we write $aR^n b$ iff. $(a, b, t) \in T$ where T is the least set such that

$$\forall a \forall b (aRb \to (a, b, 1) \in T)$$

$$\forall n \forall a \forall b \forall c (aRb \wedge (b, c, n) \in T \to (a, c, n \cup \{n\}) \in T)$$

Definition 5.3.6 For any $R \subseteq A \times A$, $R^+ = \bigcup\limits_{n \in \mathbb{N}^+} R^n$ defines the following set according to the axiom of separating.

$$\{(a, b) \in A \times A \mid \exists n((a, b, n) \in T)\}$$

where T is the set defined in **Define**5.3.5.

§ 5.4 Inference Rules and Proof Theory

Definition 5.4.1 (The natural deduction system)
$\Phi \vdash \psi$ iff. it can be established by the following proof rules in finite steps:

- $\phi[x \mapsto t] \vdash \forall x \phi$; $\forall x \phi \vdash \phi[x \mapsto t]$
- If $\Phi \vdash \psi$ and x does not freely occur in Φ, then $\Phi \vdash \forall x \psi$.
- If $\Phi, \psi \vdash \chi$ and x does not freely occur in Φ or χ, then $\Phi, \forall x\, \psi \vdash \chi$.
- $\phi, \psi \vdash \phi \wedge \psi$; $\quad \phi \wedge \psi \vdash \phi$; $\quad \phi \wedge \psi \vdash \psi$
- $\phi \vdash \phi \vee \psi$; $\quad \psi \vdash \phi \vee \psi$
- If $\Phi, \phi_1 \vdash \psi$ and $\Phi, \phi_2 \vdash \psi$, then $\Phi, \phi_1 \vee \phi_2 \vdash \psi$
- If $\Phi, \psi \vdash \chi$ and $\Phi, \neg\psi \vdash \chi$, then $\Phi \vdash \chi$
- If $\Phi, \neg\psi \vdash \chi$ and $\Phi, \neg\psi \vdash \neg\chi$, then $\Phi \vdash \psi$
- If $\phi \in \Phi$, then $\Phi \vdash \phi$
- If $\Phi \subseteq \Psi$ and $\Phi \vdash \phi$, then $\Psi \vdash \phi$
- If $\Phi \vdash \psi$ and $\Phi \vdash \psi \to \chi$, then $\Phi \vdash \chi$.
- If $\Phi, \psi \vdash \chi$, then $\Phi \vdash \psi \to \chi$.

Definition 5.4.2 (Soundness) A first order logic (“$\vdash$”) is sound if $\Phi \vdash \psi$ implies $\Phi \models \psi$.

Definition 5.4.3 (Completeness) A first order logic (“$\vdash$”) is complete if $\Phi \models \psi$ implies $\Phi \vdash \psi$.

Part III Graph Theory

Chapter VI Graph in General

(Mostly omitted)

§ 6.1 Basic definitions

1 Edges and Degrees

Definition 6.1.1 (Adjacency and Incidence)

- If $G = (V, E)$ is an undirected graph, two vertices $u, v \in V$ are adjacent (or neighbours) in G if there is an edge $e \in E$ such that the endpoints of e are u, v .
  If the endpoints of an edge e are u, v , then e is incident with u, v .
- If $G = (V, E)$ is a directed graph and $e \in E$ is from u to v , then: u is adjacent to v , and v is adjacent from u ; u is the initial vertex of the edge, while v is the terminal (or end) vertex of the edge.

Definition 6.1.2 (Neighbourhoods)

The neighbourhood $\mathcal{N}(v)$ is the set of all neighbours of v .

$\mathcal{N}(A) := \bigcup\limits_{v \in A} \mathcal{N}(v)$ for $A \subseteq V$.

Definition 6.1.3 (Degrees)

- If $G = (V, E)$ is an undirected graph, the degree of a vertex $v \in V$ is the number of edges incident with it, for which a loop associated with v contributes twice to the degree of v . Notation:

$$\begin{aligned}\deg(v) &= |\{e \mid v \text{ is } e\text{'s first endpoint}\}| + |\{e \mid v \text{ is } e\text{'s second endpoint}\}| \\ &= \sum_{v \text{ is } e\text{'s first endpoint}} 1 + \sum_{v \text{ is } e\text{'s second endpoint}} 1\end{aligned}$$

- If $G = (V, E)$ is a directed graph and $v \in V$, we define:

  $\deg^-(v) := |\{e \in E \mid e \text{ is associated with } (u_1, v_1) \text{ and } v = v_1\}|$ (in-degree);

  $\deg^+(v) := |\{e \in E \mid e \text{ is associated with } (u_1, v_1) \text{ and } v = u_1\}|$ (out-degree).

2 Loops and Circuits

(Omitted)

§ 6.2 Subgraph and Connected Components

1 Connectivity of Undirected Graph

Definition 6.2.1 (Subgraph) (Omitted)

Definition 6.2.2 (Induced Subgraph) Suppose $G = (V, E)$ is a graph and $W \subseteq V$ is a subset of vertices. The subgraph induced by W consists of all the vertices from W and all the edges from E whose endpoints both lie in W .

Definition 6.2.3 (Connected Components) Suppose $G = (V, E)$ is an undirected graph. A connected component of G is a connected subgraph that is not a proper subgraph of another connected subgraph of G .

Theorem 6.2.1 If G is a nonempty undirected graph, then G 's connected components are induced subgraphs of equivalence classes of the connectivity relation.

2 Reachability of Directed Graph

Definition 6.2.4 (Reachability) A vertex $v \in V$ is reachable from $u \in V$ if there is at least a path from u to v in G .
Two vertices $u, v \in V$ are mutually reachable if there are paths both from u to v and from v to u in G .
If $u, v \in V$ are mutually reachable, then we call u and v are strongly connected.

Definition 6.2.5 (Strongly-Connected Components) Suppose $G = (V, E)$ is a directed graph. A strongly-connected component of G is a strongly-connected subgraph of G that is not a proper subgraph of another strongly-connected subgraph of G .

Theorem 6.2.2 Given a directed graph G , mutual reachability in G is an equivalence relation.

Given a nonempty directed graph G , its strongly-connected components are induced subgraphs of equivalence classes of mutual reachability.

3 Connectivity against Vertices and Edges Removal

(Omitted)

Chapter VII Tree

§ 7.1 Unrooted tree

Definition 7.1.1 (Trees) A tree is a connected (nonempty) undirected graph without simple circuits.

Definition 7.1.2 (Leaves (in an unrooted tree)) A vertex is a leaf in a tree if its degree is 1.

Theorem 7.1.1 All statements below about an undirected graph $G = (V, E)$ are equivalent:

- G is a tree, i.e. G is connected but contains no simple circuits.
- G is connected and $|E| = |V| - 1$
- G contains no simple circuits and $|E| = |V| - 1$
- G is connected, but every edge in E is a cut edge.
- G contains no simple circuits, adding any edge to G generates a simple circuit.

- There is a unique simple path between any two G's vertices.

§ 7.2 Rooted tree

Definition 7.2.1 (Rooted trees) A rooted tree is a tree with a designated vertex (as the root)

Definition 7.2.2 (Levels) If $G = (V, E)$ is a rooted tree with the root r , we define that:

- the level of a vertex v is the length of the unique simple path from r to v ;
- the height of the tree is the maximal level of its vertices.

Definition 7.2.3 (Parents and Children) If $G = (V, E)$ is a rooted tree with the root r , the parent of a vertex $v \neq r$ is the unique vertex u such that $(u, v) \in E$ and this edge appears on the unique simple path between r and v . Also, v is called a child of u .

Definition 7.2.4 (Siblings) If $G = (V, E)$ is a rooted tree with the root r , vertices u and v are siblings if they have the same parent.

Definition 7.2.5 (Ancestors and Descendants) If $G = (V, E)$ is a rooted tree with the root r , the ancestors of a vertex $v \neq r$ are the vertices on the simple path from r to v , excluding v but including r , and the descendants of a vertex v are all vertices that have v as one of their ancestors.

Definition 7.2.6 (Subtree) If $G = (V, E)$ is a rooted tree with the root r , the subtree at a vertex v is the induced subgraph of the vertex set consisting of the vertex v (as its root) and all the descendants of v .

Chapter VIII Spanning tree, Eular path and Hamilton paths

§ 8.1 Spanning tree

1 Spanning tree

Definition 8.1.1 (Spanning Trees) Suppose G is an undirected graph. A spanning tree of G is a subgraph of G that is a tree containing every vertex of G .

Theorem 8.1.1 An undirected graph is connected if and only if it has a spanning tree.

Proof

- If a spanning tree exists, then there is a path in the spanning tree between any two vertices. Thus the graph is also connected.
- Suppose $G_0 = G$ is connected. If G_0 contains a simple circuit, then removing one edge from the simple circuit, which generates graph G_1 . G_1 must be connected. Repeat this process and construct $G_2, G_3, \cdots$. In this process, the number of edges keep decreasing. Thus, there must be a natural number k such that the number of vertices in G_k is one more than its edges. That must be a tree.

2 Minimum spanning tree

Definition 8.1.2 (Minimum spanning tree) A minimalspanning tree of G is a spanning tree whose sum of all weights in the tree is the minimum among all spanning trees.

3 Algorithms to construct minimum spanning tree

(Omitted)

§ 8.2 Eular paths and circuits

Definition 8.2.1 Given $G = (V, E)$ is a graph.

- an Euler circuit in G is a simple circuit containing every edge of G.
- an Euler path in G is a simple path containing every edge of G.

Theorem 8.2.1 A connected undirected graph with at least two vertices has an Euler circuit if and only if all of its vertices have even degree.

Proof By induction on the number of edges.

- **Base Step:** two edges
- **Inductive Step:**
  - A simple path of maximal length is a circuit, or otherwise the parity of the degrees at the start and end does not match.
  - A simple circuit of maximal length is an Euler circuit, or otherwise we get an undirected multigraph whose vertices have even degree when we remove the circuit, from which we can construct an Euler circuit in a connected component that can be merged back to the original one.

§ 8.3 Hamilton paths and circuits

Definition 8.3.1 (Hamilton Paths and Circuits) Given an undirected graph $G = (V, E)$,

- a Hamilton path is a simple path $v_0, v_1, \cdots, v_n$ of distinct vertices in G such that $V = \{v_0, v_1, \cdots, v_n\}$;

- a Hamilton circuit is a simple circuit $v_0, v_1, \cdots, v_n, v_0$ ($n \geq 0$) such that $V = \{v_0, v_1, \cdots, v_n\}$ and $v_0, v_1, \cdots, v_n$ is a Hamilton path.

Part IV Counting Theory

Chapter IX Counting Theory

§ 9.1 Inclusion-Exclusion Principle

Theorem 9.1.1 (Inclusion-Exclusion Principle) If $A_1, \cdots, A_n$ are finite sets, then

$$\left|\bigcup_{i=1}^{n} A_i\right| = \sum_{\ell=1}^{n} (-1)^{\ell+1} \sum_{1 \leqslant k_1 < \cdots < k_\ell \leqslant n} \left|\bigcap_{j=1}^{\ell} A_{k_j}\right|$$