Discrete Mathematics (0034) Lecture Notes

Yulwon Rhee (202211342)

Department of Computer Science and Engineering, Konkuk University

1 Week 1

1.1 논리와 명제

논리 : 사고의 규칙
명제 논리(Propositional Logic): T/F 판별 가능한 문장 or 수식
술어 논리(Predicate Logic): 변수 포함 명제
단순 명제(simple Proposition): 하나의 문장 or 수식으로 구성된 명제
합성 명제(Composition Proposition): 단순 명제들이 논리 연산자로 연결

항진 명제(Tautology): 항상 T인 합성 명제
모순 명제(Contradiction): 항상 F인 합성 명제

1.2 논리 연산자

Table 1: Logical Operators

연산자	기호
부정(NOT)	$\sim$
논리곱(AND)	$\wedge$
논리합(OR)	$\vee$
배타적 논리합(XOR)	$\oplus$
조건(if then)	$\rightarrow$
쌍방 조건(iff)	$\leftrightarrow$

Table 2: Truth Table for the XOR, Implication and Biconditional Proposition

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

p	q	$p \to q$
T	T	T
T	F	F
F	T	T
F	F	T

p	q	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

- p이면 q이다. (if p then q, q when p, p only if q)
- p는 q의 충분조건이다. (p is sufficient for q)
- q는 p의 필요조건이다. (q is necessary for p)
- p는 q를 함축한다. (p implies q)

1.3 논리 연산자 우선순위

$$\sim \; > \; \land \; > \; \lor \; > \; \to \; > \; \leftrightarrow$$

1.4 논리 연산: 상호 관계

Table 3: $p \to q$에 대하여

역(Converse)	$q \to p$
이(Inverse)	$\sim p \to \sim q$
대우(Contrapositive)	$\sim q \to \sim p$

1.5 예제 풀이

e.g.) You cannot ride the rollercoaster if you are under 4 ft. tall unless you are older than 16 years old.
p : You can ride the rollercoaster
q : You are under 4 ft. tall
r : You are older than 16 years old
$\sim p$ if q unless r
$\Rightarrow \sim p$ if q if $\sim r$
$\Rightarrow \sim r \to (q \to \sim p)$
$\equiv (\sim r \land q) \to \sim p$

2 Week 2

2.1 비트 연산(Bit Operations)

Table 4: Logical Operators

Name	NOT	AND	OR	XOR	Implies	Iff
Propositional Logic	$\sim$	$\wedge$	$\vee$	$\oplus$	$\rightarrow$	$\leftrightarrow$
Boolean Algebra	$\overline{p}$	$p \cdot q$	$+$	$\oplus$		
C/C++/Java(Wordwise)	!	&&	||	!=		==
C/C++/Java(Bitwise)	~	&	|	^		

2.2 논리적 동치 관계

$p \leftrightarrow q$가 항진 명제 $\rightarrow$ p, q는 논리적 동치, $p \equiv q$ 또는 $p \Leftrightarrow q$

Table 5: 논리적 동치 관계

법칙 이름	동치 관계
결합 법칙	$(p \vee q) \vee r \Leftrightarrow p \vee (q \vee r)$
	$(p \wedge q) \wedge r \Leftrightarrow p \wedge (q \wedge r)$
흡수 법칙	$p \vee (p \wedge q) \Leftrightarrow p$
	$p \wedge (p \vee q) \Leftrightarrow p$
드 모르간 법칙	$\sim (p \vee q) \equiv (\sim p) \wedge (\sim q)$
	$\sim (p \wedge q) \equiv (\sim p) \vee (\sim q)$
조건 법칙	$p \rightarrow q \Leftrightarrow \sim p \vee q$
대우 법칙	$p \rightarrow q \Leftrightarrow \sim q \rightarrow \sim p$

2.3 술어 논리(Predicate Logic)

$p(x) = x$에 대한 명제술어

2.4 술어 한정자(Predicate Quantifier)

$\forall$: 모든
$\exists$: 어떤
괄호를 이용하여 모순이 없도록 범위 지정 필요
제한된 정의역 표현: 변수가 만족해야 하는 조건이 한정기호 다음에 표기
e.g.)

$$\forall x < 0(x^2 > 0), \text{정의역은 실수} \implies \forall x(x < 0 \to x^2 > 0)$$
$$\exists x > 0(x^2 = 2), \text{정의역은 실수} \implies \exists x(x > 0 \land x^2 = 2)$$

부정:

$$\sim \forall(p(x)) \equiv \exists x(\sim p(x))$$
$$\sim (\exists x p(x)) \equiv \forall x(\sim p(x))$$

2.5 중첩 한정자

$\forall x \exists y\ P(x,y)$: For every x, there is a y for which $P(x,y)$ is true.
$\exists x \forall y\ P(x,y)$: There is an x for which $P(x,y)$ is true for every y.

2.6 예제 풀이

1. If a user is active at least one network link will be available.
   $A(u)$: User u is active.
   $S(n,x)$: Network link n is in status x.
   $\exists u\, A(u) \to \exists n\, S(n, available)$

2. Everyone has exactly one best friend.
$\to$ For every person x, person x has exactly one best friend.
$\to$ 'Exactly one' means that
   1. There is a person y who is the best friend of x.
   2. For every person z, if person z is not y, then z is not the best friend of x.

   $B(x,y)$: y is the best friend of x.
   $\forall x \exists y(B(x,y))$, y의 조건: $\forall z((z \neq y) \to \sim B(x,z))$

3. There is somebody whom everybody loves.
   $\exists y \forall x L(x,y)$ ($\exists y$와 $\forall x$ 순서 유의!)

4. Nobody loves everybody
   $\sim \exists x \forall y\ L(x,y) \equiv \forall x \exists y\ \sim L(x,y)$

3 Week 3

3.1 추론

(연역적) 추론(Argument): 주어진 명제 p_n을 바탕으로 새로운 명제 q를 유도
  p_n: 전제(Premise), 가정(Hypothesis)
  q: 결론(Conclusion)
유효 추론(Valid Argument): 전제 T, 결론 T
허위 추론(Fallacious Argument): 결론 F

Table 6: 논리적 추론 법칙

법칙 이름	추론 법칙
긍정 법칙*	$p,\ p \to q \vdash q$
부정 법칙*	$\sim q,\ p \to q \vdash \sim p$
조건 삼단 법칙*	$p \to q,\ q \to r \vdash p \to r$
선언 삼단 법칙	$p \vee q,\ \sim p \vdash q$
양도 법칙	$(p \to q) \wedge (r \to s),\ p \vee r \vdash (q \vee s)$
파괴적 법칙	$(p \to q) \wedge (r \to s),\ \sim q \vee \sim s \vdash \sim p \vee \sim r$
선접 법칙	$p \vdash p \vee q$
분리 법칙	$p \wedge q \vdash p$
연접 법칙	$p,\ q \vdash p \wedge q$

* 가장 많이 사용되고 잘 알려진 3가지 법칙

3.2 대치 vs 추론

대치의 공식은 합성 명제 전체 또는 한 부분에 적용 가능
추론의 법칙은 합성 명제의 주 연산자에 사용

3.3 증명법: 한정기호를 사용한 명제의 추론규칙

전칭 예시화(Universal Instantiation): $\forall x P(x) \to \exists c P(c)$
전칭 일반화(Universal Generalisation): $P(c)$ for an arbitrary $c \to \forall x P(x)$
존재 예시화(Existential Instantiation): $\exists x P(x) \to P(c)$ for some c (c가 적어도 하나 존재)
존재 일반화(Existential Generalisation): $P(c)$ for some $c \to \exists x P(x)$

3.4 예제 풀이

1. There is someone in this class who has been to France
   Everyine who goes to France visits the Louvre.
   Therefore, someone in this class has visited the Louvre.

Solution.

x: 사람
$C(x)$: x is in this class.
$F(x)$: x has been to France.
$L(x)$: x visits to Louvre.

$\exists x(C(x) \wedge F(x)),\ \forall x(F(x) \rightarrow L(x)) \vdash \exists x(C(x) \wedge L(x))$

Some c, $C(c) \wedge F(c)$: T (존재 예시화)
$\forall x \rightarrow c \in x,\ F(c) \rightarrow L(c)$: T (전칭 예시화)
$C(c) \wedge F(c) \vdash C(c), F(c)$
$F(c),\ F(c) \rightarrow L(c) \vdash L(c)$
$C(c),\ L(c) \rightarrow C(c) \wedge L(c)$
$C(c) \wedge L(c) \rightarrow \exists x(C(x) \wedge L(x))$ (존재 일반화)

3.5 정리의 증명

정의: 논의 대상 보편화 위해 사용 용어 or 기호 의미를 확실히 규명한 문장 or 식
  e.g.) 한 내각의 크기가 직각인 삼각형은 직각삼각형, 명제는 T/F 판별 가능한 문장 or 수식
공리: 별도 증명 없이 T로 이용되는 명제
  e.g) p가 참이면 $p \vee q$도 참, $a = b$면, $a + c = b + c$
정리: 공리, 정의로 T가 확인된 명제
증명: 공리, 정의, 정리로 명제가 T임을 확인하는 과정

3.6 증명 방법

$p \rightarrow q$ 증명: p, q 모두 T or p 무조건 거짓
직접 증명법: $p \rightarrow q$ 직접 증명
간접 증명법: 동치로 $p \rightarrow q$ 변환하여 증명. 대우 증명법, 모순 증명법, 반례 증명법, 존재 증명법
기타 증명법: 수학적 귀납법

3.7 수학적 귀납법

연역법(Deduction): 사실(Fact), 공리(Axiom)에 입각해 추론(Inference)을 통해 새로운 사실 도출
귀납법(Induction): 관찰, 실험에 기반한 가설을 귀납 추론을 통해 일반적인 규칙으로 입증

4 Week 4

4.1 직접 증명법(Direct Proof)

$p \to q$가 T 증명

4.2 모순 증명법(귀류법, Contradiction Proof)

주어진 문제의 명제 부정 후 논리 전개

$$
\begin{aligned}
\sim (p \wedge (\sim q)) &\equiv \sim p \vee \sim (\sim q) \\
&\equiv \sim p \vee \sim q \\
&\equiv \sim p \vee q \\
&\equiv p \to q
\end{aligned}
$$

$p \wedge (\sim q)$가 T라고 하고, 모순 유도 시 원래 명제 T

4.3 대우 증명법(Contrapositive Proof)

$p \to q \equiv \sim q \to \sim p$에서, $\sim q \to \sim p$가 T 증명

4.4 존재 증명법(Existence Proof)

$\exists x$ such that $p(x)$ 증명

4.5 반례 증명법(Counter-Example Proof)

반례를 통해 증명
$\forall x p(x)$가 F임을 보이기 위해 $\sim \forall x p(x) \equiv \exists x \sim p(x)$에서 $p(x)$가 F인 x 적어도 하나 존재

4.6 필요충분조건 증명법(Iff Proof)

$p \to q,\ q \to p$ 증명 $\Rightarrow p \leftrightarrow q$ 증명

5 Week 5

5.1 집합

Cardinality: 원소 개수
부분 집합(Subset): A의 모든 원소가 B의 원소에 속할 때, $A \subseteq B$. 부분 집합이 아닐 때, $A \nsubseteq B$
진부분 집합(Proper Subset): $A \subseteq B$, $A \neq B \Longrightarrow A \subset B$. 진부분 집합이 아닐 때, $A \not\subset B$
멱집합(Power Set): 모든 부분 집합을 원소로 가지는 집합 $= P(S) = 2^S$. $|P(S)| = 2^{|S|}$

5.2 부분 집합의 성질

- $\forall P,\ P \subseteq P$
- $\forall P,\ \varnothing \subseteq P$

5.3 집합의 연산

Table 7: Set Operators

연산	기호
합집합	$A \cup B$
교집합	$A \cap B$
차집합	$A - B$
대칭 차집합	$A \oplus B$
곱집합	$A \times B$

서로소: $A \cap B = \varnothing$
곱집합(Cartesian Product): $a \in A,\ b \in B,\ (a,b)$ 인 모든 순서쌍의 집합
e.g) $A = 1,2,3,\ B = a,b,c$라 할 때, $A \times B = (1,a),(1,b),(1,c),(2,a),(2,b),(2,c),(3,a),(3,b),(3,c)$
드 모르간 법칙: $\overline{(A \cup B)} = \overline{A} \cap \overline{B}$, $\overline{(A \cap B)} = \overline{A} \cap \overline{B}$

5.4 집합의 분할

분할(Partition): $\exists S \neq \varnothing (\pi = \{A_1, A_2, \cdots, A_i, \cdots, A_k\})$
1. $i = 1, \cdots, k$에 대하여, $A_i \subseteq S$ $(S \neq \varnothing)$
2. $S = A_1 \cup A_2 \cup \cdots \cup A_k$
3. $i \neq j \rightarrow A_i \cap A_j = \varnothing$

c.f.) $A_i =$ 분할의 블록

6 Week 6

6.1 보수

r 진수 정수 N 에서 $r-1$ 의 보수: $(r^n - 1) - N$
r 진수 정수 N 에서 r 의 보수: $(r^n) - N = (r - 1$의 보수$) + 1$

6.2 부호화 절대치 표현

- 연산 결과가 정확하지 않음
- 0의 표현이 2가지

6.3 1의 보수 표현

- 연산 결과는 정확하지만 (초과 비트를 더해줄 때)
- 0의 표현이 2가지

6.4 2의 보수 표현

- 연산 결과가 정확함
- 0의 표현이 1가지
- 음수 값 하나 더 표현 가능 (0의 표현이 하나 줄어들어서)

6.5 초과 비트 발생 시

- 1의 보수: 초과 비트를 덧셈
- 2의 보수: 무시

7 Week 7

7.1 행렬

대각합(Trace): 대각성분의 합. $\mathrm{tr}(A) = \mathrm{trace}(A)$
교대 행렬(Skewed-Symmetric Matrix): $A = -A^T$

7.2 행렬식

$\mathrm{Det}(A) = |A|$

$$\text{let } A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$\mathrm{Det}(A) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}$$

$$\text{let } B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}$$

$$\mathrm{Det}(B) = \begin{vmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{vmatrix} = b_{11}b_{22}b_{33} + b_{12}b_{23}b_{31} + b_{13}b_{21}b_{31} - b_{11}b_{23}b_{32} - b_{12} - b_{21} - b_{33} - b_{13}b_{22}b_{31}$$

정칙 행렬(Non-Singular Matrix): $\mathrm{Det}(A) \neq 0$
특이 행렬(Singular Matrix): $\mathrm{Det}(A) = 0$

7.3 행렬식의 성질

- $n \times n$ 행렬 A에서 임의의 두 행 또는 열이 같으면 $\mathrm{Det}(A) = 0$
- $n \times n$ 행렬 A에서 임의의 두 행 또는 열을 바꾸어서 만든 행렬 B에서 $\mathrm{Det}(B) = -\mathrm{Det}(A)$
- $n \times n$ 행렬 A에서 임의의 행 또는 열의 모든 원소가 0이면 $\mathrm{Det}(A) = 0$
- $\mathrm{Det}(A) = \mathrm{Det}(A^T)$
- $\mathrm{Det}(AB) = \mathrm{Det}(A) \cdot \mathrm{Det}(B)$
- $\mathrm{Det}(kA) = k\mathrm{Det}(A)$

가역적(Nonsingular, Invertible): A, B가 정칙 행렬. $AB = BA = I$인 경우