

# SNU 4541.664A Program Analysis

## Spring 2005

### Note 1

Prof. Kwangkeun Yi

# 귀납법 = 집합의 정의

집합을 정의하는 방법

- ▶ 되돌아서 return 歸, 바치다 dedicate 納.
- ▶ 그 집합의 원소를 가지고 그 집합의 원소를 만든다.

# 그 집합은 이것이다

- ▶ 규칙: 가정들  $X$ 와 결론  $x$ .
- ▶ “ $X$ 에 있는 것들이 정의하려는 집합에 모두 있으면,  $x$ 도 있어야 한다.”
- ▶ 그러한 집합 중에서 가장 작은 집합.
  - ▶ 규칙  $(X, x)$ 들의 모임을  $\Phi$ 라고 하자.
  - ▶ “ $\Phi$ 에 대해서 닫혀있는”( $\Phi$ -closed), “ $\Phi$ -닫힌” 집합  $A$ :

$$(X, x) \in \Phi \text{ 이고 } X \subseteq A \text{ 이면 } x \in A$$

- ▶  $\Phi$ 가 정의하는 집합: 모든  $\Phi$ -닫힌 집합들의 교집합

$$\bigcap \{A \mid A \text{는 } \Phi\text{-닫힘}\}.$$

자연수의 집합:

$$(\emptyset, 0) \quad (\{n\}, n + 1)$$

영문 소문자 알파벳으로 만들어 지는 스트링의  
집합:

$$(\emptyset, \epsilon) \quad (\{\alpha\}, x\alpha \text{ for } x \in \{a, \dots, z\})$$

# 귀납규칙 표기법

자연수 집합은

$$n \rightarrow 0 \mid n + 1$$

혹은

$$\overline{0} \quad \frac{n}{n+1}$$

위의 스트링 집합은:

$$\alpha \rightarrow \epsilon \\ \mid \quad x\alpha \quad (x \in \{a, \dots, z\})$$

혹은

$$\overline{\epsilon} \quad \frac{\alpha}{x\alpha} \quad x \in \{a, \dots, z\}$$

집합이 유한하다면 재귀가 필요없다. 집합  $\{1, 2, 3\}$ 을 규칙들로 표현하면

$$(\emptyset, 1) \quad (\emptyset, 2) \quad (\emptyset, 3)$$

혹은

$$x \rightarrow 1 \mid 2 \mid 3$$

혹은

$$\overline{1} \quad \overline{2} \quad \overline{3}$$

가 된다.

리스트의 집합:

$$\overline{\text{nil}} \quad \frac{l}{o-l}$$

혹은

$$l \rightarrow \text{nil} \mid o-l$$

말단에 정수를 가지는 두갈래 나무(*binary tree*)들의 집합:

$$\overline{n} \quad n \in \mathbb{Z} \qquad \frac{t}{\mathbf{N}(t, \mathbf{nil})}$$

$$\frac{t}{\mathbf{N}(\mathbf{nil}, t)} \qquad \frac{t_1 \quad t_2}{\mathbf{N}(t_1, t_2)}$$

혹은

$$\begin{array}{l}
 t \rightarrow n \quad (n \in \mathbb{Z}) \\
 | \quad \mathbf{N}(t, \mathbf{nil}) \\
 | \quad \mathbf{N}(\mathbf{nil}, t) \\
 | \quad \mathbf{N}(t, t)
 \end{array}$$



정수식들의 집합:

$$\overline{n} \quad n \in \mathbb{N} \qquad \frac{e}{-e}$$

$$\frac{e_1 \quad e_2}{e_1 + e_2} \qquad \frac{e_1 \quad e_2}{e_1 * e_2}$$

혹은

$$e \rightarrow n \quad (n \in \mathbb{N})$$

	$-e$
	$e + e$
	$e * e$

# 그 집합은 이렇게 만든다

규칙 집합  $\Phi$ 는 함수  $\phi$ 를 정의:

$$\phi(Y) = \{x \mid \frac{X}{x} \in \Phi, X \subseteq Y\}$$

$\Phi$  규칙들이 정의하는 집합은 함수  $\phi$ 에 의해서 닫혀있는 집합중에서

$$\bigcap \{X \mid \phi(X) \subseteq X\}$$

이다. 이 집합은  $\phi$ 의 최소고정점(least fixed point)이다.

이 집합

$$\bigcap \{X \mid \phi(X) \subseteq X\}$$

을 이렇게 만들 수 있다:

$$\phi^0 \cup \phi^1 \cup \phi^2 \cup \dots = \bigcup_{i \in \mathbb{N}} \phi^i$$

여기서

$$\phi^0 = \emptyset$$

$$\phi^n = \phi(\phi^{n-1}) \quad n \in \mathbb{N}$$

## 자연수 집합 규칙

$$n \rightarrow 0 \mid n + 1$$

그 집합은:

$$\phi^0 = \emptyset$$

$$\phi^1 = \{0\}$$

$$\phi^2 = \{0, 1\}$$

$$\phi^3 = \{0, 1, 2\}$$

...

들의 합집합.

## 리스트의 집합 규칙

$$l \rightarrow \text{nil} \mid \circ - l$$

그 집합은:

$$\phi^0 = \emptyset$$

$$\phi^1 = \{\text{nil}\}$$

$$\phi^2 = \{\text{nil}, \circ - \text{nil}\}$$

$$\phi^3 = \{\text{nil}, \circ - \text{nil}, \circ - \circ - \text{nil}\}$$

...

들의 합집합.

두갈래 나무(*binary tree*)의 집합 규칙:

$$\begin{array}{l} t \rightarrow \circ \\ | \quad \text{N}(t, \text{nil}) \\ | \quad \text{N}(\text{nil}, t) \\ | \quad \text{N}(t, t) \end{array}$$

이 집합은:

$$\phi^0 = \emptyset$$

$$\phi^1 = \{\circ\}$$

$$\phi^2 = \{\circ, \text{N}(\circ, \text{nil}), \text{N}(\text{nil}, \circ), \text{N}(\circ, \circ)\}$$

...

들의 합집합.

스트링 집합 규칙:

$$\begin{array}{l} \alpha \rightarrow \epsilon \\ | \quad x\alpha \quad (x \in \{a, \dots, z\}) \end{array}$$

이 집합은

$$\begin{aligned} \phi^0 &= \emptyset \\ \phi^1 &= \{\epsilon\} \\ \phi^2 &= \{\epsilon, a\epsilon, \dots, z\epsilon\} \\ &\dots \end{aligned}$$

들의 합집합.

# 밑바닥 없는 규칙

집합 규칙:

$$\begin{array}{l} t \rightarrow N(t, \text{nil}) \\ | \quad N(\text{nil}, t) \\ | \quad N(t, t) \end{array}$$

이 집합은  $\emptyset$ :

$$\phi^0 = \phi^1 = \dots = \emptyset$$

모든 귀납 규칙

$$\frac{X}{x}$$

에서  $X \neq \emptyset$ 이면 정의하는 집합은  $\emptyset$



# 정리

- ▶ 귀납법 = 집합의 정의 inductive definition
- ▶ 표현 방법들
- ▶ 그 집합은 무엇이지?
  - ▶ “닫혀있는 최소의 집합”
  - ▶  $\bigcap \{A \mid A \text{는 } \Phi\text{-닫힘}\}$
- ▶ 그 집합은 어떻게 만들지?
  - ▶ “ $\emptyset$ 에서 출발해서, 규칙이 만드는 원소들만을 빠짐없이 첨가”
  - ▶  $\bigcup_{i \in \mathbb{N}} \phi^i$

# 계획

- ▶ 귀납법 = 증명 proof by induction
  - ▶ 원소의 순서
  - ▶ “모든 원소들” = “모든 순서의 원소들”
- ▶ 형식논리와 추론
  - ▶ 논리식 집합의 정의 (귀납법)
  - ▶ 논리식 의미의 정의 (조립식)
  - ▶ 참논리식 집합의 정의 (귀납법)
  - ▶ 참논리식 추론의 방법 (귀납법)
  - ▶ 추론방법 평가하기

# 원소들의 순서

귀납적으로 정의된 집합  $S$

- ▶  $\phi^i$  번째에 새롭게 만들어 지는 원소:  $i$  번째 원소.
- ▶ 0 번째에 만들어진 원소들이 씨.
- ▶ “기초가 튼튼한 순서(*well-founded order*)”
- ▶ 귀납적으로 정의된 집합은 기초가 튼튼한 순서를 가지고 있다.

# 귀납법 = 증명의 방법

증명 목표:

$$\forall x \in S. P(x)$$

- ▶  $S$ 가 귀납적으로 정의됨, 즉, 모든 원소들의 순서가 있음.
- ▶  $P$ (0번째 원소)를 증명: 항상 성립
- ▶ 임의의  $i > 0$ 에 대해서

$$(\forall j < i. P(j\text{번째 원소})) \Rightarrow P(i\text{번째 원소})$$

를 증명.

(“대우법 증명”도 있음)

# 귀납 증명: 규칙들에 대한 것으로

임의의  $i > 0$ 에 대해서

$$(\forall j < i. P(j\text{번째 원소})) \Rightarrow P(i\text{번째 원소})$$

를 증명.

▶ case  $i = 1$ :  $P(1\text{번째 원소})$  증명

▶ case  $i > 1$ :

$$(\forall j < i. P(j\text{번째 원소})) \Rightarrow P(i\text{번째 원소})$$

증명

즉,

▶ “base case”: 규칙  $\frac{x}{x}$ 가 만드는  $x$ 들에 대해,  $P(x)$  증명.

▶ “inductive case”: 규칙  $\frac{X}{x}$ 가 만드는  $x$ 들에 대해,  $(\forall a \in X. P(a)) \Rightarrow P(x)$ 를 증명.

# 귀납증명 예

- ▶ 증명:  $\forall n \in \mathbb{N}. 0 + 1 + 2 + \dots + n = n(n+1)/2$
- ▶ 증명: 모든 두갈래가 꽉찬 나무(*complete binary tree*)는 말단 노드의 갯수가 내부 노드의 갯수와 같거나 하나 많다.

$$t \rightarrow o \mid N(t, t)$$

# 논리식 집합

귀납적 정의

$$\begin{array}{l} f \rightarrow T \mid F \\ | \quad \neg f \\ | \quad f \wedge f \\ | \quad f \vee f \\ | \quad f \Rightarrow f \end{array}$$

# 논리식 의미

조립식 정의 compositional definition

$$\begin{aligned} \llbracket T \rrbracket &= \text{true} \\ \llbracket F \rrbracket &= \text{false} \\ \llbracket \neg f \rrbracket &= \text{not} \llbracket f \rrbracket \\ \llbracket f_1 \wedge f_2 \rrbracket &= \llbracket f_1 \rrbracket \text{ andalso } \llbracket f_2 \rrbracket \\ \llbracket f_1 \vee f_2 \rrbracket &= \llbracket f_1 \rrbracket \text{ orelse } \llbracket f_2 \rrbracket \\ \llbracket f_1 \Rightarrow f_2 \rrbracket &= \llbracket f_1 \rrbracket \text{ implies } \llbracket f_2 \rrbracket \end{aligned}$$

임의의 논리식  $f$ 의 의미가 정의 된 셈.



$\llbracket (T \wedge (T \vee F)) \Rightarrow F \rrbracket$   
=  $\llbracket T \wedge (T \vee F) \rrbracket$  implies  $\llbracket F \rrbracket$   
= ( $\llbracket T \rrbracket$  andalso  $\llbracket T \vee F \rrbracket$ ) implies false  
= (true andalso ( $\llbracket T \rrbracket$  orelse  $\llbracket F \rrbracket$ )) implies false  
= (true andalso (true orelse false)) implies false  
= false

# 어떤 집합의 정의

쌍  $(\{g_1, \dots, g_n\}, f)$ 들의 집합

$$\begin{array}{c} \frac{}{(\Gamma, T)} \quad \frac{}{(\Gamma, f)} \quad f \in \Gamma \\ \\ \frac{(\Gamma, f_1) \quad (\Gamma, f_2)}{(\Gamma, f_1 \wedge f_2)} \qquad \frac{(\Gamma, f_1 \wedge f_2)}{(\Gamma, f_1)} \\ \\ \frac{(\Gamma, f_1)}{(\Gamma, f_1 \vee f_2)} \qquad \frac{(\Gamma, f_1 \vee f_2) \quad (\Gamma \cup \{f_1\}, f_3) \quad (\Gamma \cup \{f_2\})}{(\Gamma, f_3)} \\ \\ \frac{(\Gamma \cup \{f_1\}, f_2)}{(\Gamma, f_1 \Rightarrow f_2)} \qquad \frac{(\Gamma, f_1 \Rightarrow f_2) \quad (\Gamma, f_1)}{(\Gamma, f_2)} \\ \\ \frac{(\Gamma \cup \{f\}, F)}{(\Gamma, \neg f)} \qquad \frac{(\Gamma, f) \quad (\Gamma, \neg f)}{(\Gamma, F)} \end{array}$$

# 형식논리의 표기법으로

$$\overline{\Gamma \vdash T} \quad \overline{\Gamma \vdash f} \quad f \in \Gamma$$

$$\frac{\Gamma \vdash F}{\Gamma \vdash f} \quad \frac{\Gamma \vdash \neg\neg f}{\Gamma \vdash f}$$

$$\frac{\Gamma \vdash f_1 \quad \Gamma \vdash f_2}{\Gamma \vdash f_1 \wedge f_2}$$

$$\frac{\Gamma \vdash f_1 \wedge f_2}{\Gamma \vdash f_1}$$

$$\frac{\Gamma \vdash f_1}{\Gamma \vdash f_1 \vee f_2}$$

$$\frac{\Gamma \vdash f_1 \vee f_2 \quad \Gamma \cup \{f_1\} \vdash f_3 \quad \Gamma \cup \{f_2\} \vdash f_3}{\Gamma \vdash f_3}$$

$$\frac{\Gamma \cup \{f_1\} \vdash f_2}{\Gamma \vdash f_1 \Rightarrow f_2}$$

$$\frac{\Gamma \vdash f_1 \Rightarrow f_2 \quad \Gamma \vdash f_1}{\Gamma \vdash f_2}$$

$$\frac{\Gamma \cup \{f\} \vdash F}{\Gamma \vdash \neg f}$$

$$\frac{\Gamma \vdash f \quad \Gamma \vdash \neg f}{\Gamma \vdash F}$$

# 또 다른 시선: 증명들의 집합을 정의

증명들의 집합을 만드는 귀납규칙 (“증명규칙” inference rules).

- ▶ 예를 들어, 증명규칙

$$\frac{\Gamma \vdash f_1 \quad \Gamma \vdash f_2}{\Gamma \vdash f_1 \wedge f_2}$$

은 증명을 만드는 귀납 규칙

- ▶  $\Gamma \vdash f_1$ 와  $\Gamma \vdash f_2$ 의 증명들을 가지고  $\Gamma \vdash f_1 \wedge f_2$ 의 증명을 만든다.

# 증명 나무

$$\frac{\frac{\frac{\{p \rightarrow \neg p, p\} \vdash p}{\{p \rightarrow \neg p, p\} \vdash p \rightarrow \neg p} \quad \frac{\{p \rightarrow \neg p, p\} \vdash p}{\{p \rightarrow \neg p, p\} \vdash p}}{\{p \rightarrow \neg p, p\} \vdash \neg p} \quad \frac{\{p \rightarrow \neg p, p\} \vdash F}{\{p \rightarrow \neg p, p\} \vdash \neg p}}{\{p \rightarrow \neg p\} \vdash \neg p}$$

# 증명 규칙의 평가

기계가 만드는  $\{g_1, \dots, g_n\} \vdash f$  는 어떤 것들인가? 예)  $\llbracket g_1 \wedge \dots \wedge g_n \Rightarrow f \rrbracket = \text{true}$  인가?

- ▶ 기계의 안전성 soundness:

$$\Gamma \vdash f \text{ 이면 } \llbracket \Gamma \Rightarrow f \rrbracket = \text{true}$$

- ▶ 기계의 완전성 completeness:

$$\Gamma \vdash f \text{ 면이 } \llbracket \Gamma \Rightarrow f \rrbracket = \text{true}$$