

# SNU 프로그래밍언어 특강

(1.0)

이 광근

[kwangkeunyi.snu.ac.kr](http://kwangkeunyi.snu.ac.kr)

# 계획

무엇 의미구조 denotational semantics 세계에서 증명방법

- ▶ 고정점에 대한 증명법
  - ▶ 고정점 인덕 fixpoint induction 으로 증명하기
  - ▶ 고정점 정의를 이용해서 증명하기
- ▶ 겉보기 증명 extentional proof 스타일

(미리보기) 실행 의미구조 operational semantics 세계에서 증명 대비 장점

# 무엇 의미구조 denotational semantics 증명법 I: 고정점에 대한

- ▶ 프로그램 의미는 연속함수의 최소고정점
- ▶ 프로그램 의미의 성질 = 최소고정점의 성질
- ▶ 최소고정점의 성질 증명 방법
  1. 고정점 인덕 fixpoint induction으로 (만능 아님)
  2. 고정점 정의를 이용하기

# 고정점 인덕션 fixpoint induction

- ▶ CPO  $D$ , 연속함수  $f : D \rightarrow D$

$$lfp f = \bigsqcup C \quad (C \stackrel{\text{let}}{=} \{f^i \perp_D \mid i \in \mathbb{N}\})$$

증명할것:  $P(lfp f)$  (성질  $P$ )

- ▶  $\forall x \in C. P(x)$ 를 보이면,  $P(lfp f)$ 을 보인것임?
- ▶ 아님;  $lfp f \notin C$  인 경우가 흔하다(무한 CPO)

단, 성질  $P$ 가 품에넣는 성질, inclusive assertion 인 경우라면 오케이:

- ▶ 고정점 체인  $C$  집합을 만드는 인덕규칙:

$$\perp \quad \frac{x}{f(x)}$$

이므로

- ▶  $P(\perp)$ 임을 보이고
- ▶  $P(x) \Rightarrow P(f(x))$ 을 보이면,
- ▶  $P(\text{lfp } f)$ 도 사실임 (" $\text{lfp } f$ 도 그 품에 들어감").

품에넣는 성질  $P$ : 관심 체인  $C \subseteq D$ 에 대해서

$$(\forall x \in C. P(x)) \implies P(\bigsqcup C)$$

인 성질.

▶  $P$ 가 품에넣는 성질? 생김새로 판단하기

$$P \rightarrow P \wedge P \mid \forall \vec{y}. EP$$

$$EP \rightarrow EP \vee EP \mid Q(\vec{y}) \mid f(\vec{y}) \sqsubseteq g(\vec{y})$$

$Q$       1차 논리식 first order predicate

$f, g$       연속함수

(p.215, *Denotational Semantics: The Scott-Strachey Approach to Programming Language Theory*, Joseph E. Stoy)

(“Inductive Method for Proving Properties of Programs”, Manna, Ness, Vuillemin)

▶  $P$ 가 품에넣는 성질? 내용으로 판단하기

- ▶ 관심 체인  $C \subseteq D$ 에 대해서

$$(\forall x \in C. P(x)) \implies P(\bigsqcup C)$$

인지 확인

- ▶ 예) 연속함수  $f, g \in D \rightarrow D$ , 증명할것:  $P(\text{lfp } f, \text{lfp } g)$ ,

$$P(a, b) \stackrel{\text{let}}{=} \forall x. (a(x) \sqsubseteq k_1 \implies b(x) \sqsubseteq k_2).$$

위의  $P$ 는 품에넣는 성질이다

- ▶ 관심 체인  $C = \{(f^i \perp, g^i \perp) \mid i \in \mathbb{N}\}$ 에 대해

$$(\forall (a, b) \in C. P(a, b)) \implies P(\bigsqcup C)$$

이므로. (왜?)

# 고정점 인덕션 fixpoint induction 예

$$\begin{aligned} \llbracket \text{while } E \ C \rrbracket &= \text{lfp } \lambda X. (\lambda M. \llbracket E \rrbracket M \ ? \ X (\llbracket C \rrbracket M) : M) \\ \llbracket \text{repeat } C \ E \rrbracket &= \text{lfp } \lambda X. ((\lambda M. \llbracket E \rrbracket M \ ? \ X M : M) \circ \llbracket C \rrbracket) \end{aligned}$$

증명:  $\llbracket C ; \text{while } E \ C \rrbracket = \llbracket \text{repeat } C \ E \rrbracket$  즉,

$$\begin{aligned} \llbracket C ; \text{while } E \ C \rrbracket &= (\text{lfp} F) \circ \llbracket C \rrbracket \\ \llbracket \text{repeat } C \ E \rrbracket &= \text{lfp} G \\ F, G &= \dots \end{aligned}$$

이므로, 증명할 것은

$$P(\text{lfp} F, \text{lfp} G) \stackrel{\text{let}}{=} (\text{lfp} F \circ \llbracket C \rrbracket = \text{lfp} G).$$

- ▶ 품에 넣는 성질, inclusive predicate 이므로
- ▶ 고정점 인덕, fixpoint induction 으로:

$P(\perp, \perp)$ 를 보이고,  $P(f, g) \Rightarrow P(F(f), G(g))$ 를 보인다.

# 고정점 정의를 이용한 증명법

최소고정점 정의를 이용해서

$$lfp f \stackrel{\text{def}}{=} \bigcap \{x \mid f(x) \sqsubseteq x\}$$

- ▶ 조건  $f(A) \sqsubseteq A$ 를 만족하는  $A$ 를 찾고
- ▶  $P(A)$  임을 보이면,  $P(lfp f)$ 이 사실이다.
  - ▶ 단,  $P$ 가  $(P(x) \wedge y \sqsubseteq x \implies P(y))$ 이어야
- ▶ 왜:  $lfp f$ 은 그런  $A$ 보다 작은 원소이므로.

## 고정점 정의를 이용하는 증명 예

$f(n) = \text{if } n=0 \text{ then } \infty \text{ else } (f(n-1) \parallel \text{true})$

▶ 증명할 것:  $\llbracket f \rrbracket \sqsubseteq \lambda x. \text{true}$

▶ 즉,  $\text{lfp}(F \stackrel{\text{let}}{=} \lambda f. \lambda n. n = 0? \perp : (f(n-1) \text{ or}^* \text{true}))$   
 $\sqsubseteq \lambda x. \text{true}.$

증명하기:

1. 찾아라  $F(f) \sqsubseteq f$  인  $f$ , 그리고
2.  $f$ 가, 확인하려는 성질을 만족함을 확인하라:  
 $f \sqsubseteq \lambda x. \text{true}$ , 그러면
3.  $\text{lfp} F \sqsubseteq \lambda x. \text{true}$ 이다.

그런  $f$ 는  $(\lambda n. n = 0? \perp : \text{true})$

# 고정점 정의를 이용하는 증명 예

$f(n) = \text{if } n=0 \text{ then } \infty \text{ else } (f(n-1) \parallel \text{true})$

▶ 증명할 것:  $\llbracket f \rrbracket \sqsubseteq \lambda x. \text{true}$

▶ 즉,  $\text{lfp}(F \stackrel{\text{let}}{=} \lambda f. \lambda n. n = 0? \perp : (f(n-1) \text{ or}^* \text{true}))$   
 $\sqsubseteq \lambda x. \text{true}.$

증명하기:

1. 찾아라  $F(f) \sqsubseteq f$  인  $f$ , 그리고
2.  $f$ 가, 확인하려는 성질을 만족함을 확인하라:  
 $f \sqsubseteq \lambda x. \text{true}$ , 그러면
3.  $\text{lfp} F \sqsubseteq \lambda x. \text{true}$ 이다.

그런  $f$ 는  $(\lambda n. n = 0? \perp : \text{true})$

(참고) 실행 의미구조 operational semantics 세계에서 증명한다면 상대적으로 지루함. 인덕 증명 + 실행과정 추적:  $\forall n \in \mathbb{N}. f(n)$ 이 안멈추거나  $\text{true}$  값을 계산함.

# 무엇 의미구조 denotational semantics 증명법 II: 걸보기

## 증명 extentional proof 스타일

확인하고자 하는 성질  $P$ 가 의미 세계안에서 드러나는 “걸모습”에 관한 것일때.

- ▶ 예) 프로그램변환 맞음  $\Rightarrow P(\llbracket \text{출발프로그램} \rrbracket, \llbracket \text{도착프로그램} \rrbracket)$ .
  - ▶  $P(a, b) \stackrel{\text{def}}{=} a = b$  (걸모습 = 무엇의의미가 같음)
  - ▶  $P(a, b) \stackrel{\text{def}}{=} a \sim b$  (걸모습 = 무엇의의미 세계에서 어떤 관계)
- ▶ 걸모습 증명의 효용: 좀더 확신에 가까워지기 (짬)
  - ▶ 마치, SW테스트에서 “간접테스트 metamorphic test” 같은

# 겉보기 증명 extentional proof 스타일: 예 1

이고가기 lifting 변환에 대해서

- ▶ 함수의 자유변수를 인자로 이고가도록 변환
- ▶  $f(x) = x+a \longrightarrow f(x,a) = x+a$
- ▶ 함수정의를 맨바깥으로 옮기고
- ▶ 함수호출을 한상차림으로

이런 이고가기 변환은 맞는가?

- ▶ 이고가기 변환  $\text{liftExp}$ :

$$\begin{aligned} \text{liftExp} : (A \rightarrow X \rightarrow Y) \text{Exp} \\ \rightarrow ((A \times X) \rightarrow Y) \text{Exp} \end{aligned}$$

- ▶ 대응하는 의미세계의 함수  $\text{lift}$ :

$$\begin{aligned} \text{lift} : (A \rightarrow (X \rightarrow Y) \rightarrow (X \rightarrow Y)) \\ \rightarrow (A \times X \rightarrow Y) \rightarrow (A \times X \rightarrow Y) \end{aligned}$$

다음 성질을 만족한다(가정):

$$(\text{lift } F) f (a, x) = F a (\lambda x'. f(a, x')) x$$

- ▶ 다음 걸모습을 증명하자: (의미세계에서 이고가기의 올바름)

$$(\text{lfp}(F a)) x = (\text{lfp}(\text{lift } F))(a, x).$$

증명목표:

$$(lfp(Fa)) x = (lfp(lift F))(a, x)$$

고정점 인덕으로 증명:  $P(lfp(F a), lfp(lift F))$

$$P(f, g) \stackrel{\text{let}}{=} (f x = g (a, x)).$$

▶  $\perp_{X \rightarrow Y} x = \perp_{A \times X \rightarrow Y} (a, x)$ ? 네.

▶  $f x = g (a, x)$ 이면  $P((F a) f, (lift F) g)$ ? 네, 왜냐면

$$\begin{aligned}(lift F) g (a, x) &= F a (\lambda x'. g(a, x')) x && \text{(성질)} \\ &= F a (\lambda x'. f x') x && \text{(인덕가정)} \\ &= F a f x.\end{aligned}$$

## 겉보기 증명 extentional proof 스타일: 예 2

마저할일전달(continuation-passing-style) 변환에 관해서.  
(마저할일전달 강의 후)

# 딱맞는 의미구조 full abstraction semantics

무엇 의미구조 denotational semantics 평가하기

- ▶ 정의한 의미구조가 실제세계와 딱맞아야
- ▶ 무엇 의미가 같으면 실행 의미도 같고, 다르면 달라야

정의)  $\llbracket \cdot \rrbracket$ 는 딱맞는 의미구조 full abstraction semantics, fully abstract:

$$\llbracket E \rrbracket = \llbracket E' \rrbracket \Leftrightarrow \forall \text{context } C \llbracket \cdot \rrbracket. C[E] \stackrel{\text{behave}}{=} C[E'].$$

정의)  $E \stackrel{\text{behave}}{=} E' : \dots$

- ▶  $\llbracket f(n)=\text{if } n=0 \text{ then } 1 \text{ else } f(n-1) \rrbracket = \llbracket g(n)=1 \rrbracket$ ?  
네. 굿.
- ▶  $\llbracket \infty \parallel \text{true} \rrbracket = \llbracket \text{true} \rrbracket$ ? 음.
  - ▶  $\llbracket \parallel \rrbracket$ 을 조심히 정의해야
  - ▶ 모든 경우마다 실행의미와 일치하도록

# 값중심언어의 무엇 의미구조 denotational semantics

$E$	$\rightarrow$	$n$	자연수
		$x$	변수
		$\text{fn } x E$	함수값
		$\text{rec } x E$	재귀값
		$E E$	함수적용

$$\text{값 } \mathbb{V} = \mathbb{N}_\perp + (\mathbb{V} \rightarrow \mathbb{V})$$

$$\text{환경 } Env = Var_\perp \rightarrow \mathbb{V}$$

$$\llbracket E \rrbracket \in Env \rightarrow \mathbb{V}$$

$$\llbracket n \rrbracket \sigma = n$$

$$\llbracket x \rrbracket \sigma = \sigma(x)$$

$$\llbracket \text{fn } x E \rrbracket \sigma = \lambda v. (\llbracket E \rrbracket \sigma \{x \mapsto v\})$$

$$\llbracket \text{rec } x E \rrbracket \sigma = \text{lfp } \lambda v. (\llbracket E \rrbracket \sigma \{x \mapsto v\})$$

$$\llbracket E_1 E_2 \rrbracket \sigma = (\llbracket E_1 \rrbracket \sigma) \cdot (\llbracket E_2 \rrbracket \sigma)$$