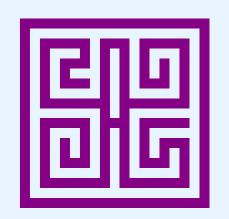


프로그램 따로분석의 이론적 기틀: 재귀적 모듈 지원하기

이준협 이광근

서울대학교 프로그래밍 연구실 (ROPAS)



문제 소개

무엇을 하고 싶은가?

- 프로그램 외부의 이름을 몰라도 분석하고 싶다. ex. OCaml에서 C 함수를 부르는 경우
- 이름이 "복잡한 방식"으로 알려질 때에도 분석하고 싶다. ex. First-class modules, Functors, Recursive modules

어떻게 할 것인가?

- "Algebraic Effects": 환경을 받는 **함수**를 **자료구조**로 나타냄.
- 환경과 상호작용을 요약하고, 나중에 환경이 들어오면 풀자!

모듈이 있는 언어의 정의

겉모습 (Untyped λ+Modules)

Identifiers
$$x \in \text{Var}$$

Expression $e \to x \mid \lambda x.e \mid e e \mid \lambda\text{-calculus}$

$$\mid e \rtimes e \quad \text{linked expression}$$

$$\mid \varepsilon \quad \text{empty module}$$

$$\mid x = e \; ; \; e \quad \text{(recursive) binding}$$

속내용

Environment
$$\sigma \in \operatorname{Env}$$

Location $\ell \in \operatorname{Loc}$

Value $v \in \operatorname{Val} \triangleq \operatorname{Env} + \operatorname{Var} \times \operatorname{Expr} \times \operatorname{Env}$

Weak Value $w \in \operatorname{WVal} \triangleq \operatorname{Val} + \operatorname{Loc} \times \operatorname{Val}$

Environment $\sigma \to \bullet$ empty stack

 $| (x, w) :: \sigma$ weak value binding

 $| (x, \ell) :: \sigma$ free location binding

Value $v \to \sigma$ exported environment

 $| (\lambda x.e, \sigma)$ closure

Weak Value $w \to v$ value

 $| \mu \ell.v$ recursive value

실행의미

 $(e,\sigma) \downarrow v$

$$\frac{ \underset{(x,\sigma)}{\text{ID}} = v}{\sigma(x) = v} \qquad \frac{ \underset{(x,\sigma)}{\text{RECID}} = \mu\ell.v}{(x,\sigma) \Downarrow v} \qquad \frac{\text{FN}}{(\lambda x.e,\sigma) \Downarrow \langle \lambda x.e,\sigma \rangle}$$

$$\frac{ \underset{(e_1,\sigma)}{\text{APP}} (e_1,\sigma) \Downarrow \langle \lambda x.e,\sigma_1 \rangle \qquad (e_2,\sigma) \Downarrow v_2 \qquad (e,(x,v_2) :: \sigma_1) \Downarrow v}{(e_1 e_2,\sigma) \Downarrow v}$$

$$\frac{ \underset{(e_1,\sigma)}{\text{LINK}} (e_1,\sigma) \Downarrow \sigma_1 \qquad (e_2,\sigma_1) \Downarrow v}{(e_1 \bowtie e_2,\sigma) \Downarrow v} \qquad \frac{\text{EMPTY}}{(\varepsilon,\sigma) \Downarrow \bullet}$$

$$\frac{\text{RecBind}}{\ell \notin \text{FLoc}(\sigma)} \quad \underbrace{(e_1,(x,\ell)::\sigma) \Downarrow v_1 \quad (e_2,(x,\mu\ell.v_1)::\sigma) \Downarrow \sigma_2}_{(x=e_1;e_2,\sigma) \Downarrow (x,\mu\ell.v_1)::\sigma_2}$$

상호작용 기록하고 나중에 풀기

상호작용 기록

상호작용 풀기 (일부 규칙들)

 $\mid w\downarrow_{\sigma_0} W$

$$\frac{\text{R-Init}}{\text{Init} \downarrow \sigma_0} \quad \frac{\overset{R-\text{Read}}{E \downarrow \Sigma} V = \Sigma(x)}{\text{Read}(E,x) \downarrow V} \quad \frac{\overset{R-\text{ReadRec}}{E \downarrow \Sigma} \mu.V = \Sigma(x)}{\text{Read}(E,x) \downarrow V^{\mu.V}}$$

$$\frac{\overset{R-\text{CallV}}{E \downarrow \langle \lambda x.e, \Sigma \rangle} \quad v \downarrow V \quad (e,(x,V) :: \Sigma) \downarrow V'}{\text{Call}(E,v) \downarrow V'} \quad \frac{\overset{R-\text{CallE}}{E \downarrow E'} \quad v \downarrow V}{\text{Call}(E,v) \downarrow \text{Call}(E',V)}$$

Conjecture (Linking)

$$\sigma \downarrow_{\sigma_0} \Sigma$$
 and $(e, \sigma) \Downarrow v$ and $(e, \Sigma) \Downarrow V \Rightarrow v \downarrow_{\sigma_0} V$

요약하기