

Más sobre Semántica Operacional

Yolanda Ortega Mallén

Dpto. de Sistemas Informáticos y Computación
Universidad Complutense de Madrid

Sumario

- Construcciones no secuenciales.
- Bloques y procedimientos.

Aborto

$$S ::= x := a \mid \text{skip} \mid S_1; S_2 \mid \text{if } b \text{ then } S_1 \text{ else } S_2 \mid \text{while } b \text{ do } S \mid \text{abort}$$

- No se añaden reglas semánticas.
- Paso corto: `abort` no es semánticamente equivalente a `skip` ni a `while true do skip`.
- Paso largo: `abort` no es semánticamente equivalente a `skip` pero sí a `while true do skip`.

Semántica de paso largo vs. semántica de paso corto

- 1 La semántica de paso largo **no distingue** entre **ciclar** y **terminación anormal**.
- 2 En paso corto ciclar se refleja en secuencias **infinitas** de derivación y la terminación anormal en secuencias de derivación **finitas** pero **bloqueadas**.

Ejercicio 3.2

Extender el lenguaje **While** con la sentencia `assert b before S`.

- Extender la semántica de paso corto.
- Demostrar que `assert true before S` es semánticamente equivalente a `S`, pero `assert false before S` no es semánticamente equivalente ni a `skip` ni a `while true do skip`.

Indeterminación

$$S ::= x := a \mid \text{skip} \mid S_1 ; S_2 \mid \text{if } b \text{ then } S_1 \text{ else } S_2 \mid \text{while } b \text{ do } S \mid S_1 \text{ or } S_2$$

Semántica de paso largo

$$[\text{or}_{\text{bs}}^1] \quad \frac{\langle S_1, s \rangle \rightarrow s'}{\langle S_1 \text{ or } S_2, s \rangle \rightarrow s'}$$

$$[\text{or}_{\text{bs}}^2] \quad \frac{\langle S_2, s \rangle \rightarrow s'}{\langle S_1 \text{ or } S_2, s \rangle \rightarrow s'}$$

Semántica de paso corto

$$[\text{or}_{\text{ss}}^1] \quad \langle S_1 \text{ or } S_2, s \rangle \Rightarrow \langle S_1, s \rangle$$

$$[\text{or}_{\text{ss}}^1] \quad \langle S_1 \text{ or } S_2, s \rangle \Rightarrow \langle S_2, s \rangle$$

Semántica de paso largo vs. semántica de paso corto

- 1 En paso largo la indeterminación elimina los ciclos cuando es posible.
- 2 En paso corto no.

Paralelismo

$$S ::= x := a \mid \text{skip} \mid S_1 ; S_2 \mid \text{if } b \text{ then } S_1 \text{ else } S_2 \mid \text{while } b \text{ do } S \mid S_1 \text{ par } S_2$$

Semántica de paso corto

$$[\text{par}_{ss}^1] \frac{\langle S_1, s \rangle \Rightarrow \langle S'_1, s' \rangle}{\langle S_1 \text{ par } S_2, s \rangle \Rightarrow \langle S'_1 \text{ par } S_2, s' \rangle}$$

$$[\text{par}_{ss}^2] \frac{\langle S_1, s \rangle \Rightarrow s'}{\langle S_1 \text{ par } S_2, s \rangle \Rightarrow \langle S_2, s' \rangle}$$

$$[\text{par}_{ss}^3] \frac{\langle S_2, s \rangle \Rightarrow \langle S'_2, s' \rangle}{\langle S_1 \text{ par } S_2, s \rangle \Rightarrow \langle S_1 \text{ par } S'_2, s' \rangle}$$

$$[\text{par}_{ss}^4] \frac{\langle S_2, s \rangle \Rightarrow s'}{\langle S_1 \text{ par } S_2, s \rangle \Rightarrow \langle S_1, s' \rangle}$$

Semántica de paso largo vs. semántica de paso corto

- 1 En paso largo la ejecución de las componentes es **atómica**; **no se puede expresar el intercalado**.
- 2 En paso corto el intercalado se expresa fácilmente.

Bloques y declaración de variables

Nueva categoría sintáctica: $D_V \in \mathbf{Dec}_V$,

$$D_V ::= \text{var } x := a; D_V \mid \varepsilon$$

$$S ::= x := a \mid \text{skip} \mid S_1; S_2 \mid \text{if } \dots \mid \text{while } b \text{ do } S \mid \text{begin } D_V \ S \ \text{end}$$

Generalización de la modificación de estados: $s[X \mapsto s']$

Semántica de paso largo

$$[\text{var}_{\text{bs}}] \quad \frac{\langle D_V, s[x \mapsto \mathcal{A}[[a]]s] \rangle \rightarrow_D s'}{\langle \text{var } x := a; D_V, s \rangle \rightarrow_D s'}$$

$$[\text{none}_{\text{bs}}] \quad \langle \varepsilon, s \rangle \rightarrow_D s$$

$$[\text{block}_{\text{bs}}] \quad \frac{\langle D_V, s \rangle \rightarrow_D s', \langle S, s' \rangle \rightarrow s''}{\langle \text{begin } D_V \ S \ \text{end}, s \rangle \rightarrow s''[\text{DV}(D_V) \mapsto s]}$$

Procedimientos

Nueva categoría sintáctica: $D_P \in \mathbf{Dec}_P$,

$$D_P ::= \text{proc } p \text{ is } S; D_P \mid \varepsilon$$
$$S ::= x := a \mid \dots \mid \text{while } b \text{ do } S \mid \text{begin } D_V \ D_P \ S \ \text{end} \mid \text{call } p$$

- Ámbito **dinámico** para variables y procedimientos.
- Ámbito **dinámico** para variables y **estático** para procedimientos.
- Ámbito **estático** para variables y procedimientos.

Ámbito dinámico (variables y procedimientos)

- Entorno de procedimientos: $env_P \in \mathbf{Env}_P = \mathbf{Pname} \hookrightarrow \mathbf{Stm}$

$$\text{updP}(\text{proc } p \text{ is } S; D_P, env_P) = \text{updP}(D_P, env_P[p \mapsto S])$$

$$\text{updP}(\varepsilon, env_P) = env_P$$

- Sistema de transiciones: $env_P \vdash \langle S, s \rangle \rightarrow s'$

$$[\text{block}_{bs}] \frac{\langle D_V, s \rangle \rightarrow_D s', \text{updP}(D_P, env_P) \vdash \langle S, s' \rangle \rightarrow s''}{env_P \vdash \langle \text{begin } D_V \ D_P \ S \ \text{end}, s \rangle \rightarrow s'' [DV(D_V) \mapsto s]}$$

$$[\text{call}_{bs}^{\text{rec}}] \frac{env_P \vdash \langle env_P \ p, s \rangle \rightarrow s'}{env_P \vdash \langle \text{call } p, s \rangle \rightarrow s'}$$

Ámbito estático para procedimientos

Entorno de procedimientos: $\mathbf{Env}_P = \mathbf{Pname} \hookrightarrow \mathbf{Stm} \times \mathbf{Env}_P$

$$\text{updP}(\text{proc } p \text{ is } S; D_P, \text{env}_P) = \text{updP}(D_P, \text{env}_P[p \mapsto (S, \text{env}_P)])$$

$$\text{updP}(\varepsilon, \text{env}_P) = \text{env}_P$$

$$[\text{call}_{\text{bs}}] \frac{\text{env}'_P \vdash \langle S, s \rangle \rightarrow s'}{\text{env}_P \vdash \langle \text{call } p, s \rangle \rightarrow s'}$$

siendo $\text{env}_P p = (S, \text{env}'_P)$

$$[\text{call}_{\text{bs}}^{\text{rec}}] \frac{\text{env}'_P[p \mapsto (S, \text{env}'_P)] \vdash \langle S, s \rangle \rightarrow s'}{\text{env}_P \vdash \langle \text{call } p, s \rangle \rightarrow s'}$$

Ámbito estático (variables y procedimientos)

- Entorno de variables: $env_V \in \mathbf{Env}_V = \mathbf{Var} \rightarrow \mathbf{Loc}$
- Almacén: $sto \in \mathbf{Store} = \mathbf{Loc} \cup \{\text{next}\} \rightarrow \mathbb{Z}$
- $\text{new} : \mathbf{Loc} \rightarrow \mathbf{Loc}$

$$[\text{var}_{\text{bs}}] \quad \frac{\langle D_V, (env_V[x \mapsto l], sto[l \mapsto v][\text{next} \mapsto \text{new } l]) \rangle \rightarrow_D (env'_V, sto')}{\langle \text{var } x := a; D_V, (env_V, sto) \rangle \rightarrow_D (env'_V, sto')}$$

donde $l = sto \text{ next}$ y $v = \mathcal{A}[[a]](sto \circ env_V)$

$$[\text{none}_{\text{bs}}] \quad \langle \varepsilon, (env_V, sto) \rangle \rightarrow_D (env_V, sto)$$

Entorno de procedimientos: $\mathbf{Env}_P = \mathbf{Pname} \hookrightarrow \mathbf{Stm} \times \mathbf{Env}_V \times \mathbf{Env}_P$

$$\text{updP}(\text{proc } p \text{ is } S; D_P, env_V, env_P) = \text{updP}(D_P, env_V, env_P[p \mapsto (S, env_V, env_P)])$$

$$\text{updP}(\varepsilon, env_V, env_P) = env_P$$

Ámbito estático (variables y procedimientos)

Sistema de transiciones: $env_V, env_P \vdash \langle S, sto \rangle \rightarrow sto'$

$$[ass_{bs}] \quad env_V, env_P \vdash \langle x := a, sto \rangle \rightarrow sto[env_V x \mapsto \mathcal{A}[[a]](sto \circ env_V)]$$

$$\langle D_V, (env_V, sto) \rangle \rightarrow_D (env'_V, sto'),$$

$$[block_{bs}] \quad \frac{env'_V, updP(D_P, env'_V, env_P) \vdash \langle S, sto' \rangle \rightarrow sto''}{env_V, env_P \vdash \langle \text{begin } D_V D_P S \text{ end}, sto \rangle \rightarrow sto''}$$

$$[call_{bs}] \quad \frac{env'_V, env'_P \vdash \langle S, sto \rangle \rightarrow sto'}{env_V, env_P \vdash \langle \text{call } p, sto \rangle \rightarrow sto'}$$

$$[call_{bs}^{rec}] \quad \frac{env'_V, env'_P[p \mapsto (S, env'_V, env'_P)] \vdash \langle S, sto \rangle \rightarrow sto'}{env_V, env_P \vdash \langle \text{call } p, sto \rangle \rightarrow sto'}$$

siendo $env_P p = (S, env'_V, env'_P)$

Ejercicios

Ejercicio 3.14

Las nuevas reglas dadas para el ámbito estático para la asignación, `skip`, composición secuencial, condicionales y bucles constituyen una semántica alternativa del lenguaje **While** original.

Formular y **demostrar** la **equivalencia** entre esta nueva semántica y la de paso largo vista anteriormente.

Ejercicio 3.15

Modificar la sintaxis de la declaración de procedimientos para que puedan recibir **dos parámetros de paso por valor**:

$$D_P ::= \text{proc } p(x_1, x_2) \text{ is } S; D_P \mid \varepsilon$$

$$S ::= x := a \mid \dots \mid \text{call } p(a_1, a_2)$$

Los entornos de procedimiento serán ahora elementos de:

$$\text{Env}_P = \text{Pname} \leftrightarrow \text{Var} \times \text{Var} \times \text{Stm} \times \text{Env}_V \times \text{Env}_P$$

Modificar la semántica convenientemente y dar nuevas reglas para las llamadas a procedimientos (no recursivos y recursivos).