Práctica 2: El Lenguaje EAB

Lenguajes de Programación I

López Soto Ramses Antonio
Quintero Villeda Erik

13 de septtiembre de 2019

Objetivo

Implementar la semántica del lenguaje EAB en el lenguaje de programación *Haskell*

Introducción

Semántica dinámica

Las reglas de la semántica dinámica son las siguientes:

- Estados

$$\frac{}{bool[true]\ valor}\ vtrue$$

$$\frac{}{bool[false]\ valor}\ vfalse$$

$$\frac{}{num[n]\ valor}\ vnum$$

$$\frac{t\ asa}{t\ estado}\ edo$$

- Estados iniciales

$$\frac{t\ asa\ FV(t) = \emptyset}{t\ inicial}\ ein$$

- Estados finales

$$\frac{}{bool[true]\ valor}\ vtrue$$

$$\frac{}{bool[false]\ valor}\ vfalse \quad \frac{}{num[n]\ valor}\ vnum$$

$$\frac{}{bool[true]\ final}\ ftrue$$

$$\frac{}{bool[false]\ final}\ ffalse \quad \frac{}{num[n]\ final}\ fnum$$

- Jucios de las expresiones aritméticas

$$\frac{}{suma(num[n], num[m]) \rightarrow num[n+m]}\ esumaf$$

$$\frac{}{prod(num[n], num[m]) \rightarrow num[n*m]}\ eprodf$$

$$\frac{t_1 \rightarrow t_1'}{suma(t_1, t_2]) \rightarrow suma(t_1', t_2)}\ esumai$$

$$\frac{t_2 \rightarrow t_2'}{suma(num[n], t_2]) \rightarrow suma(num[n], t_2')}\ esumad$$

$$\frac{t_1 \rightarrow t_1'}{prod(t_1, t_2]) \rightarrow prod(t_1', t_2)}\ eprodi$$

$$\frac{t_2 \rightarrow t_2'}{prod(num[n], t_2]) \rightarrow prod(num[n], t_2')}\ esumad$$

$$\frac{}{suc(num[n]) \rightarrow num[n+1]}\ esucn$$

$$\frac{}{pred(num[0]) \rightarrow num[0]}\ epred0$$

$$\frac{}{pred(num[n+1]) \rightarrow num[n]}\ epreds$$

$$\frac{t \rightarrow t'}{suc(t) \rightarrow suc(t')}\ esuc$$

$$\frac{t \rightarrow t'}{pred(t) \rightarrow pred(t')}\ epred$$

- Jucios de expresiones booleanas

$$\frac{}{if(boo[true], t_2, t_3) \rightarrow t_2}\ eiftrue$$

$$\frac{}{if(boo[false], t_2, t_3) \rightarrow t_3}\ eiffalse$$

$$\frac{t_1 \to t_1'}{if(t_1, t_2, t_3) \to if(t_1', t_2, t_3)}\ eif$$

$$\frac{t_1 \to t_1'}{and(t_1, t_2) \to and(t_1', t_2)}$$

$$\frac{t_2 \to t_2'}{and(bool[true], t_2) \to and(bool[true], t_2')}$$

$$\frac{}{and(bool[false], t_2) \to bool[false])}\ eiand$$

$$\frac{t_1 \to t_1'}{or(t_1, t_2) \to or(t_1', t_2)}$$

$$\frac{}{or(bool[true], t_2) \to bool[true]}$$

$$\frac{t_1 \to t_1'}{not(t) \to not(t')}$$

$$\frac{}{not(bool[false]) \to bool[true])}$$

$$\frac{}{not(bool[true]) \to bool[false]}$$

$$\frac{t_1 \to t_1'}{lt(t_1, t_2) \to lt(t_1', t_2)}$$

$$\frac{t_2 \to t_2'}{lt(num[n], t_2) \to lt(num[n], t_2')}$$

$$\frac{}{l(num[n], num[m]) \to bool[n < m])}$$

$$\frac{t_1 \to t_1'}{gt(t_1, t_2) \to gt(t_1', t_2)}$$

$$\frac{t_2 \to t_2'}{gt(num[n], t_2) \to gt(num[n], t_2')}$$

$$\frac{}{gt(num[n], num[m]) \to bool[n > m])}$$

$$\frac{t_1 \to t_1'}{eq(t_1, t_2) \to eq(t_1', t_2)}$$

$$\frac{t_2 \to t_2'}{eq(num[n], t_2) \to eq(num[n], t_2')}$$

$$\frac{}{eq(num[n], num[m]) \to bool[n = m])}$$

- Jucios de let

$$\frac{v \quad valor}{let(v, x.e_2) \to e_{2[x:=v]}}\ eletf$$

$$\frac{t1 \to t_1'}{let(t_1, x.t_2) \to let(t_1', x.t_2)}\ eleti$$

Semántica estática

Las reglas de la semántica estática son las siguientes

- Tipado de variables

$$\frac{}{\Gamma, x : T \vdash x : T}\ tvar$$

- Tipado de valores numéricos

$$\frac{}{\Gamma \vdash num[n] : Nat}\ tnum$$

-Tipado de valores booleanos

$$\frac{}{\Gamma \vdash bool[true] : Bool}\ ttrue$$

$$\frac{}{\Gamma \vdash bool[false] : Bool}\ tfalse$$

- Tipado de expresiones aritméticas

$$\frac{\Gamma \vdash t_1 : Nat\ \Gamma \vdash t_2 : Nat}{\Gamma \vdash suma(t_1, t_2) : Nat}\ tsum$$

$$\frac{\Gamma \vdash t_1 : Nat\ \Gamma \vdash t_2 : Nat}{\Gamma \vdash prod(t_1, t_2) : Nat}\ tprod$$

$$\frac{\Gamma \vdash t : Nat}{\Gamma \vdash suc(t) : Nat}\ tsuc$$

$$\frac{\Gamma \vdash t : Nat}{\Gamma \vdash pred(t) : Nat}\ tpred$$

- Tipado de espresiones booleanas

$$\frac{\Gamma \vdash t_1 : Bool\ \Gamma \vdash t_2 : Bool}{\Gamma \vdash and(t_1, t_2) : Bool}\ tand$$

$$\frac{\Gamma \vdash t_1 : Bool\ \Gamma \vdash t_2 : Bool}{\Gamma \vdash or(t_1, t_2) : Nat}\ tor$$

$$\frac{\Gamma \vdash t : Bool}{\Gamma \vdash not(t) : Bool}\ tnot$$

$$\frac{\Gamma \vdash t_1 : Nat\ \Gamma \vdash t_2 : Nat}{\Gamma \vdash lt(t_1, t_2) : Bool}\ tlt$$

$$\frac{\Gamma \vdash t_1 : Nat\ \Gamma \vdash t_2 : Nat}{\Gamma \vdash gt(t_1, t_2) : Bool}\ tgt$$

$$\frac{\Gamma \vdash t_1 : Nat\ \Gamma \vdash t_2 : Nat}{\Gamma \vdash eq(t_1, t_2) : Nat}\ teq$$

$$\frac{\Gamma \vdash t_1 : Bool\ \Gamma \vdash t_2 : T\ \Gamma \vdash t_3 : T}{\Gamma \vdash if(t_1, t_2, t_3) : T}\ tif$$

- Tipado de expresiones let

$$\frac{\Gamma \vdash e_1 : T\ \Gamma, x : T \vdash e_2 : T}{\Gamma \vdash let(e_1, x.e_2) : S}\ tlet$$

Desarollo

La implenentación de las relgas se realizó a través del lenguaje de programación Haskell.

En el código se visualizará de una mejor manera cómo se hizo.

Conclusión

La implenentación de la semántica de la EAB fue muy problemática pues cada que lograbamos que una función se ejecutara correctamente, salían más y más casos en los cuales se ciclaba o no lanzara el resultado esperado. La función con la cual ésto sucedió más veces fue con evals pues sólo nos devolvía los casos en los cuales la entrada era correcta, pero en otros casos se ciclaba.

En resumen, la práctica fue mucho más compleja en resolver los errores de ejecucion.

El archivo Semantic.hs sí funciona pero al hacer la llamada con make desde el archivo raíz marca un error que no supimos arreglar. Se debe ejecutar en el directorio BAE/src.