§5. Dynamic Behavior of Programs

5.1 Creation

In the beginning, we must create two sets \( Q_0 \) and \( V_0 \).

5.1.1 All \( \langle \text{expression} \rangle \)'s in standard \( \langle \text{declaration} \rangle \)'s are rewritten in the form of normal (as defined in the next paragraph).

5.1.2 Let \( V_0 \) stand for the set of the all \( \langle \text{identifier} \rangle \)'s declared by some standard \( \langle \text{declaration} \rangle \), and let \( a(V) \) be able, and \( f(V) \) be an abstract element, different from each other, for each \( \langle \text{variable} \rangle \) \( V \) in \( V_0 \).

5.1.3 Let \( Q_0 \) stand for the set
\[
\{ \ f(V) \mid V \in V_0 \},
\]
and let \( v(Q) \) be the value of \( Q \) for each \( Q \in Q_0 \).

(If standard \( \langle \text{declaration} \rangle \) of a \( \langle \text{variable} \rangle \) \( V \in V_0 \) is of the form

\"let \ V \ be \ procedure \ (T_1, \ldots, T_n) \ by \ ((V_1, \ldots, V_n)E)\"

and \( f(V) = Q \), then

\( v(Q) \) is \( (V_1, \ldots, V_n)E. \).)

5.1.4 Let \( Q_0 \) stand for an abstract element \( \notin Q_0 \), and let \( L_0 \) stand for a \( \langle \text{label} \rangle \).

5.2 Normalization

Let \( E_1 \) stand for a legal program, \( V_1 \) stand for the set of the all \( \langle \text{identifier} \rangle \)'s contained in \( E_1 \), and \( L_1 \) stand for the set of the all \( \langle \text{label} \rangle \)'s contained in \( E_1 \).

1) Let \( V \) stand for \( V_0 \cup V_1 \), and let \( a(V) \) be \( \text{inable} \) for \( V \in V \setminus V_0 \).

2) Let \( L \) stand for \( L_1 \cup L_0 \).

3) Let \( Q \) stand for \( Q_0 \cup Q_0 \), and let \( v(Q) \) be \( \text{done} \).

4) \( \tau(E_1) \Rightarrow E_2 \).

5) Let \( D \) be a \( \langle \text{form declaration} \rangle \) in \( E_2 \) of the form

\"let \ G \ represent \ F\"

with a \( \langle \text{form} \rangle \) \( G \) and an \( \langle \text{expression} \rangle \) \( F \).

Dynamic Behavior of Programs-1
\[ \varepsilon (\forall) \Rightarrow \forall \;
\]

Replace in \( E_2 \) \( D \) with

"let \( V \) be \( F \);

let \( G \) represent \( V \); ".

6) Let \( E' \) be a \(<\text{form call}>\) in \( E_2 \), of the form

"\( P_0 \, E_1 \, P_1 \, E_2 \, P_2 \, \ldots \, P_{n-1} \, E_n \, P_n \)"

where \( E_1, \ldots, E_n \) be \(<\text{expression}>\)'s and \( P_i \) be empty or a sequence of
\(<\text{mark}>\)'s for \( i = 0, 1, \ldots, n \). If the operator form

\( (P_0 \, t(E_1) \, P_1 \, t(E_2) \, P_2 \, \ldots \, P_{n-1} \, t(E_n) \, P_n) \)

is declared by a declaration of the form

"let \( G \) represent \( F \),

then, replace \( E' \) in \( E_2 \) with

"\( (F(E_1), \ldots, E_n) \)".

7) When \( T \) is a \(<\text{typifier}>\) in \( E_2 \), replace \( T \) in \( E_2 \) with \( t(T) \).

8) Eliminate all \(<\text{form declaration}>\) and \(<\text{mark declaration}>\) in \( E_2 \), and let
\( E \) stand for the result.

An \(<\text{expression}>\) of the form as \( E \) is called normal.

5.3 Elaboration of a Normal Program

\( \varepsilon (E) ; \)

if the result is a quantity \( Q \), then the elaboration of \( E \) is thus completed, but
if the result is a \(<\text{label}>\) \( L \), then the elaboration of \( E \) is undefined.