# 1 Programs with procedures

- $P$ - set of procedures that includes an initial procedure $main$.
- $V_G$ - tuple of global variables that includes a return variable $r$.
- $V_p$ - tuple of local variables for each procedure $p \in P$ that includes a program counter variable $pc_p$.
- $init(V_G, V_{main})$ - initialization condition.
- $error_p(V_G, V_p)$ - error condition for each procedure $p \in P$.
- $step_p(V_G, V_p, V'_G, V'_p)$ - intra-procedural transitions.
- $call_{p,q}(V_G, V_p, V_q)$ - parameter passing transitions for a caller $p \in P$ and a callee $q \in P$.
- $ret_p(V_G, V_p)$ - return value passing transition for $p \in P$.
- $loc_p(V_p, V'_p)$ - evolving program counter accross call sites in $p \in P$.

# 2 Reachability

$$\frac{(g, l_{main}) \models init(V_G, V_{main})}{((g, l_{main}), \epsilon) \in Reach_{main}}$$

$$\frac{((g, l_p), st) \in Reach_p \quad ((g, l_p), (g', l'_p)) \models step_p(V_G, V_p, V'_G, V'_p)}{((g', l'_p), st) \in Reach_p}$$

$$\frac{((g, l_p), st) \in Reach_p \quad (g, l_p, l_q) \models call_{p,q}(V_G, V_p, V_q)}{((g, l_q), l_p \cdot st) \in Reach_q}$$

$$\frac{((g, l_p), l_q \cdot st) \in Reach_p \quad (g, l_p, g') \models ret_p(V_G, V_p, V'_G) \quad (l_q, l'_q) \models loc_q(V_q, V'_q)}{((g', l'_q), st) \in Reach_q}$$

# 3 Summarization

$$\frac{(g, l_{main}) \models init(V_G, V_{main})}{((g, l_{main}), (g, l_{main})) \in summ_{main}}$$

$$\frac{((g, l_p), (g', l'_p)) \in summ_p \quad ((g', l'_p), (g'', l''_p)) \models step_p(V_G, V_p, V'_G, V'_p)}{((g, l_p), (g'', l''_p)) \in summ_p}$$

$$\frac{((g, l_p), (g', l'_p)) \in summ_p \quad ((g', l'_p, l_q)) \models call_{p,q}(V_G, V_p, V_q)}{((g', l_q), (g', l_q)) \in summ_q}$$

$$\frac{\begin{array}{cc}((g, l_p), (g', l'_p)) \in summ_p & ((g', l'_p, l_q)) \models call_{p,q}(V_G, V_p, V_q) \\ ((g', l_q), (g'', l'_q)) \in summ_q \quad (g'', l'_q, q''') \models ret_q(V_G, V_q, V'_G) & (l'_p, l''_p) \models loc_p(V_p, V'_p)\end{array}}{((g, l_p), (g''', l''_p)) \in summ_p}$$

Inference rule as entailment:

$$summ_p(V_G, V_p, V'_G, V'_p) \wedge call_{p,q}(V'_G, V'_p, V_q) \wedge summ_q(V'_G, V_q, V''_G, V'_q) \wedge$$
$$ret_q(V''_G, V'_q, V'''_G) \wedge loc_p(V'_p, V''_p) \models summ_p(V_G, V_p, V'''_G, V''_p)$$