

Prova scritta di Linguaggi - 25.01.2018 (4 esercizi in totale)

Si consideri il linguaggio *Lang*:

$$\begin{aligned}
 E \in \text{Lang} ::= & x \mid l \mid n \mid \{lab_1 = E_1, \dots, lab_k = E_k\} \mid \\
 & \text{ref } E \mid E_1 + E_2 \mid E_1 \wedge E_2 \mid E_1 \geq E_2 \mid \#lab E \mid \\
 & !E \mid E_1 := E_2 \mid \text{let } x : T = E_1 \text{ in } E_2 \mid E_1; E_2 \mid \\
 & \text{skip} \mid \text{fun}(x : T) \Rightarrow E \mid E_1 E_2 \mid \text{fix}.E
 \end{aligned}$$

1. (3 punti) Formalizzare la *statica* del linguaggio *Lang* fornendo: i) una grammatica dei tipi ammessi nel linguaggio; ii) un sistema per il tipaggio; iii) **un sistema per il sottotipaggio** (dare *solo* le regole che userete nel prossimo esercizio).
2. (10 punti) Dire, giustificando *formalmente* la risposta, se, in presenza di sottotipaggio, il termine

$$\begin{aligned}
 & \text{let } f : T_1 = \text{fun}(x : T_2) \Rightarrow \{a = \#a x, b = (\#a x) \geq 0, c = \#c x\} \text{ in} \\
 & \left((\text{fun}(g : \text{ref}\{b : \text{bool}\}) \Rightarrow \#b g)(\text{ref}(f\{a = 5, b = 6, c = 7\})) \right) \\
 & \quad \wedge \\
 & \left(\text{fun}(g : \text{ref}\{a : \text{int}, c : \text{int}\}) \Rightarrow (\#a g) \geq (\#c g)(\text{ref}(f\{a = 7, c = 8\})) \right)
 \end{aligned}$$

è ben tipato o meno, al variare dei tipi T_1 e T_2 . Nel caso sia tipabile fornire il tipo dell'intera espressione. Usare *esclusivamente* le regole di sottotipaggio viste a lezione.

3. (12 punti) Si consideri il seguente linguaggio concorrente:

$$\begin{aligned}
 E \in \text{Lang} ::= & l \mid \text{true} \mid \text{false} \mid !l \mid l := E \\
 & E_1 \parallel E_2 \mid E_1 \oplus E_2 \mid \text{await } E_1 \text{ protect } E_2 \text{ end}
 \end{aligned}$$

dove l'operatore \oplus denota la scelta non deterministica. Supponendo di aver definito type system e semantica operativa anche per i costrutti concorrenti. Provare **formalmente** se le seguenti leggi algebriche sono vere o false per E_1 e E_2 arbitrari. Se le leggi non dovessero valere, si fornisca un **controesempio**, e si mostri formalmente se la simulazione è verificata in uno o entrambi i sensi.

- (a) $\alpha; (\text{skip}; E_1 \parallel \alpha; E_2) \approx_{\Gamma} \text{skip}; \alpha; \alpha; (E_1 \parallel E_2)$
dove α è un arbitrario assegnamento semplice.
- (b) $(\text{await } \text{true} \text{ protect } (E_1 \parallel E_2) \text{ end}) \approx_{\Gamma} (\text{await } \text{true} \text{ protect } E_2 \text{ end}) \parallel (\text{await } \text{true} \text{ protect } E_1 \text{ end})$
- (c) $\alpha; \alpha; (\text{skip}; E_1 \oplus \text{skip}; E_2) \oplus (\alpha; E_1) \approx_{\Gamma} \alpha; (\text{skip}; (\alpha; E_1 \oplus \alpha; E_2)) \oplus (\alpha; \alpha; E_2)$
dove α è un arbitrario assegnamento semplice.

4. (6 punti) Si consideri il seguente linguaggio:

$$E \in \text{Lang} ::= x \mid n \mid E_1 + E_2 \mid \text{fun}(x : T) \Rightarrow E \mid E_1 E_2$$

Se ne dia una semantica operativa in stile CBV (ovviamente left-to-right). Si riporti formalmente la proprietà di Determinacy (Determinatezza) per il linguaggio e si fornisca una prova formale di tale proprietà. Nel caso la proprietà non dovesse valere si fornisca un controesempio.