

Dev. (11) - Sémantique d'extensions de IMP

leçon 930 sémantique des langages de programmation

réf: Nielson & Nielson

(1) Non-déterminisme.

While_{nd} est le langage défini par

$$S ::= (\dots) \mid S_1 \text{ or } S_2$$

avec, pour SN, les règles $\frac{(S_1, s) \rightarrow s'}{(S_1 \text{ or } S_2, s) \rightarrow s'} \quad \text{ou} \quad \frac{(S_2, s) \rightarrow s'}{(S_1 \text{ or } S_2, s) \rightarrow s'}$

Donner un exemple bête: $x \leftarrow 1 \text{ or } (x \leftarrow 2; x \leftarrow x+2)$.

Pour SOS, on ajoute les dérivations $(S_1 \text{ or } S_2, s) \Rightarrow (S_1, s)$
 $(S_1 \text{ or } S_2, s) \Rightarrow (S_2, s)$

Si l'une des alternatives était while true do skip,
→ avec SN on choisit l'autre alternative, la boucle infinie est supprimée.
→ avec SOS on peut choisir l'alternative qui boucle.

(2) Parallélisme

While_{par} est déf. par $S ::= (\dots) \mid S_1 \text{ par } S_2$

Reprenons l'exemple: les différentes options sont calculées en parallèle.
 $x \leftarrow 1 \text{ par } (x \leftarrow 2; x \leftarrow x+2)$ peut donner $x=1$, $x=4$ ou $x=3$.

Pour SOS, 4 règles:

$$\frac{(S_1, s) \Rightarrow (S_1', s')}{(S_1 \text{ par } S_2, s) \Rightarrow (S_1' \text{ par } S_2, s')} \quad \frac{(S_1, s) \Rightarrow s'}{(S_1 \text{ par } S_2, s) \Rightarrow (S_2, s')}$$

et idem en dérivant S_2 .

Appliqués à l'exemple.

Et pour SN? on tente de tracer, mais ça ne marche pas.