AFGL – TD 3

SÉMANTIQUE

1 Sémantique opérationnelle

Q1.1 *Évaluer* $(x * y) - 14$ *dans l'état* s *tel que* $s(x) = 7$ *et* $s(y) = 8$.

Rappels et notations :

- l'évaluation d'une expression arithmétique a dans un état s est notée par la paire $<a, s>$
- $<a, s> \longrightarrow n$ exprime le fait que l'expression a s'évalue en n dans l'état s
- un état est une fonction associant chaque variable à sa valeur

Ici, $s = \{x \mapsto 7, y \mapsto 8\}$

Voir le cours pour les règles d'évaluation. L'évaluation de l'expression donne (en remontant) :

$$\frac{\dfrac{<x, s> \longrightarrow ? \qquad <y, s> \longrightarrow ?}{<x * y, s> \longrightarrow ?} \qquad <14, s> \longrightarrow ?}{<(x * y) - 14, s> \longrightarrow ?}$$

Puis on redescend en suivant les règles A_1 (évaluation d'une expression constante), A_2 (évaluation d'une variable), A_4 (produit) et A_5 (soustraction).

$$\frac{\dfrac{<x, s> \longrightarrow s(x) = 7 \qquad <y, s> \longrightarrow s(y) = 8}{<x * y, s> \longrightarrow 7 * 8 = 56} \qquad <14, s> \longrightarrow 14}{<(x * y) - 14, s> \longrightarrow 56 - 14 = 42}$$

Q1.2 *Appliquer une suite de règles pour l'exécution des commandes suivantes :*

- `x:=x+1` dans l'état s tel que $s(x) = 2$
- `if x<0 then x:=x-1 else x:=x+1` dans l'état s tel que $s(x) = 5$
- `while x<100 do x:=x*2` dans l'état s tel que $s(x) = 50$

Les règles utilisées sont C_2 (affectation), C_4 et C_5 (alternative), C_6 et C_7 (boucle).

$$\frac{\dfrac{<x, s> \longrightarrow 2 \qquad <1, s> \longrightarrow 1}{<x + 1, s> \longrightarrow 3}}{<x := x + 1, s> \longrightarrow s[x \leftarrow 3]}$$

$s[x \leftarrow 3]$ est l'état obtenu à partir de s en (ré)associant x à 3.

Pour l'alternative, il faut évaluer l'expression booléen $x < 0$ dans l'état s :
$\mathcal{B}[[x < 0]]s = \mathcal{A}[[x]]s < \mathcal{A}[[x]]s = x < 0 = ff$, ce qui nous donne la règle C_5.

$$\dfrac{\dfrac{\dfrac{<x, s> \longrightarrow 5 \qquad <1, s> \longrightarrow 1}{<x+1, s> \longrightarrow 6}}{<x := x+1, s> \longrightarrow s[x \leftarrow 6]}}{<\text{if } x<0 \text{ then } x := x-1 \text{ else } x := x+1, s> \longrightarrow s[x \leftarrow 6]}$$

Pour la boucle, il faut évaluer l'expression booléen $x < 100$ dans l'état s puis dans l'état s_1 :
$\mathcal{B}[[x<100]]s = tt$ ce qui nous donne la règle C_7 et $\mathcal{B}[[x<100]]s_1 = ff$ ce qui nous donne la règle C_6.

$$\dfrac{\dfrac{\dfrac{<x, s> \longrightarrow 50 \qquad <2, s> \longrightarrow 2}{<x*2, s> \longrightarrow 100}}{<x := x*2, s> \longrightarrow s_1 = s[x \leftarrow 100]} \qquad <\text{while } x<100 \text{ then } x := x*2, s_1> \longrightarrow s_1}{<\text{while } x<100 \text{ then } x := x*2, s> \longrightarrow s_1}$$

2 Sémantique axiomatique

Q2.1 *Donner les obligations de preuve pour boucle suivante.*

$\{x \geq 0\}$ `while x<b do x:=x+1` $\{x \geq 0 \wedge \neg(x<b)\}$

Rappel et notations :

- I : Invariant, ici $I = x \geq 0$
- C : Condition, ici $C = x < b$
- V : Variant, ici $V = b - x$

Correction totale d'une boucle : $V \in \mathbb{N}$ est une contrainte de *typage*, $V < \gamma$ une garantie de *terminaison*.

$$\dfrac{\{I \wedge C \Rightarrow V \in \mathbb{N}\} \qquad \{I \wedge C \wedge V = \gamma\}S\{I \wedge V < \gamma\}}{\{I\} \text{ while } C \text{ do } S \ \{I \wedge \neg C\}}$$

Typage : $x \geq 0 \wedge x < b \Rightarrow b - x > 0 \wedge b - x \in \mathbb{N}$

$\{x \geq 0 \wedge x < b \wedge b - x = \gamma\}$ x := x+1 $\{x \geq 0 \wedge b - x < \gamma\}$

car $\{x \geq 0 \wedge x < b \wedge b - x = \gamma\} \Rightarrow \{x + 1 \geq 0 \wedge b - x - 1 < \gamma\}$

Q2.2 *Quelle est la précondition P pour que le jugement suivant soit prouvable ?*

$\{P\}$ `x:=0` $\{x = 0\}$

On cherche la précondition la plus faible : {True} `x:=0` $\{x = 0\}$

Q2.3 *Quelle est la postcondition Q pour que le jugement suivant soit prouvable ?*

$\{x \geq 0\}$ `x:=x+2` $\{Q\}$

Rappel : $\{P(E)\}x := E\{P(x)\}$

Notation du cours : $\{P[x \leftarrow E]\}x := E\{P\}$

Pour que P soit vraie pour x après affectation, il fallait que P soit vraie pour E avant exécution

Après changement de variable `X:=x+2` :
$x \geq 0 \Rightarrow X - 2 \geq 0 \Rightarrow X \geq 2$

On retient la postcondition la plus forte : $\{x \geq 0\}$ `x:=x+2` $\{x \geq 2\}$

Q2.4 *Prouver les deux formules suivantes.*

$\{x = 3a\}$ `a:=a+2` $\{x = 3a - 6\}$
$\{a \geq 0\}$ `x:=1` $\{a = x * a\}$

Avec un changement de variable : $\{x = 3a\}$ A`:=`a`+2` $\{x = 3A - 6\}$

$x = 3(A - 2) \Rightarrow x = 3A - 6$

Après substitution : $a = 1 * a$ est toujours vrai, donc la formule est prouvée pour toutes les préconditions (y compris $a \geq 0$).