

## Contrôle

Durée : 1h30. Aucun document autorisé. Le barème est donné à titre indicatif.

**Exercice 1 : (6 points)** Si c'est possible, donner la dérivation de typage et l'exécution en sémantique naturelle et en SOS des deux programmes suivants ; sinon, indiquer pourquoi. On partira de l'environnement de typage  $\Gamma = x:\text{int}, y:\text{int}$  et de la mémoire  $\mu = [(x, 2), (y, 0)]$ .

- a) `declare v: bool := x > y`  
`begin`  
`null;`  
`while v loop x := x - 2 ; v := x > y endloop`  
`end`
- b) `declare v: bool := (y + 2) > x`  
`begin`  
`if x > v then x := x + y else null ; y := y + x endif`  
`end`

**Exercice 2 :** On se limite au sous-langage défini ci-dessous.

**Types.**

$$\tau ::= \text{int} \mid \text{bool}$$

**Termes.**

$$e ::= \bar{n} \mid e > e \mid \text{true} \mid \text{false} \mid \text{not } e$$

**Règles de typage.**

$$\frac{}{\vdash \bar{n}:\text{int}} \quad \frac{\vdash e_1:\text{int} \quad \vdash e_2:\text{int}}{\vdash e_1 > e_2:\text{bool}}$$

$$\frac{}{\vdash \text{true}:\text{bool}} \quad \frac{}{\vdash \text{false}:\text{bool}}$$

$$\frac{\vdash e:\text{bool}}{\vdash \text{not } e:\text{bool}}$$

**Sémantique opérationnelle structurée.**

$$\frac{\vdash e_1 \searrow_e e'_1}{\vdash e_1 > e_2 \searrow_e e'_1 > e_2} \quad \frac{\vdash e_2 \searrow_e e'_2}{\vdash \bar{n} > e_2 \searrow_e \bar{n} > e'_2}$$

$$\frac{n > m}{\vdash \bar{n} > \bar{m} \searrow_e \text{true}} \quad \frac{n \leq m}{\vdash \bar{n} > \bar{m} \searrow_e \text{false}}$$

$$\frac{\vdash e \searrow_e e'}{\vdash \text{not } e \searrow_e \text{not } e'} \quad \frac{}{\vdash \text{not true} \searrow_e \text{false}} \quad \frac{}{\vdash \text{not false} \searrow_e \text{true}}$$

- a) **(2 points)** Définir par induction sur la structure des expressions une fonction  $\mathcal{N}$  qui compte le nombre de `not` d'une expression  $e$ . Par exemple :

$$\mathcal{N}(\text{true}) = 0, \quad \mathcal{N}(\text{not } (\text{not } (\text{false}))) = 2$$

- b) **(4 points)** Prouver la propriété de préservation du typage.
- c) **(4 points)** Prouver la propriété de vivacité de l'évaluation.
- d) **(4 points)** Prouver que toute évaluation de toute expression termine.