Raphaël Charrondière
raphael.charrondiere@inria.fr
Semaine 12 - échéance S13

Correction TD 10: Logique de Hoare

Tester c'est bien, prouver c'est mieux

Dans le cadre de ce TD, les entiers sont naturels (c'est à dire positifs).

Exercice 1 Sommer et multiplier les entiers

On considère les programmes suivants :

$x := 0$; $z := 1$; **while** z $\leqslant$ y **do** ($x := x + z$; $z := z + 1$) **P1**

$u := 0$;
while x>1 **do if** pair(x) **then** ($x := \frac{x}{2}$; $y := 2 \times y$) **else** ($x := x - 1$; $u := u + y$);
$y := y + u$ **P2**

---

1 Utiliser la logique de Hoare pour montrer

— $\{y = n \wedge n \geqslant 0\}$**P1**$\{x = \frac{n\times(n+1)}{2}\}$
— $\{x = x_0 \wedge y = y_0 \wedge x_0 > 0\}$**P2**$\{y = x_0 \times y_0\}$

`Correction :`

$$\{y = n \wedge n \geqslant 0\} \implies \{y = n \wedge 0 = 0 \wedge 1 \leqslant y + 1\}$$
$$x := 0$$
$$\{y = n \wedge x = \frac{1(1-1)}{2} \wedge 1 \leqslant y + 1\}$$
$$z := 1$$
$$\{y = n \wedge x = \frac{z(z-1)}{2} \wedge z \leqslant y + 1\}$$
$$\textbf{while}(z \leqslant y)$$
$$\{y = n \wedge x = \frac{z(z-1)}{2} \wedge z \leqslant y + 1 \wedge z \leqslant y\} \implies \{y = n \wedge x + z = \frac{z(z+1)}{2} \wedge z + 1 \leqslant y + 1\}$$
$$x := x + z$$
$$\{y = n \wedge x = \frac{z(z+1)}{2} \wedge z + 1 \leqslant y + 1\}$$
$$z := z + 1$$
$$\{y = n \wedge x = \frac{z(z-1)}{2} \wedge z \leqslant y + 1\}$$
$$\{y = n \wedge x = \frac{z(z-1)}{2} \wedge z > y \wedge z \leqslant y + 1\} \implies \{x = \frac{n(n+1)}{2}\}$$

Raphaël Charrondière
raphael.charrondiere@inria.fr

$\{x = x_0 \wedge y = y_0 \wedge x_0 > 0\} \implies \{xy + 0 = x_0 + y_0 \wedge x > 0\}$
$u := 0$
$\{xy + u = x_0 + y_0 \wedge x > 0\}$
while$(x > 1)$
$\{xy + u = x_0 + y_0 \wedge x > 0 \wedge x > 1\} \implies \{xy + u = x_0 + y_0 \wedge x > 0\}$
if $\text{pair}(x)$
$\{\text{pair}(x) \wedge xy + u = x_0 + y_0 \wedge x > 0\} \implies \{\frac{x}{2}2y + u = x_0 + y_0 \wedge \frac{x}{2} > 0\}$
$x := \frac{x}{2}$
$\{x2y + u = x_0 + y_0 \wedge x > 0\}$
$y := 2y$
$\{xy + u = x_0 + y_0 \wedge x > 0\}$
else
$\{\neg\text{pair}(x) \wedge xy + u = x_0 + y_0 \wedge x > 0\} \implies \{(x-1)y + u + y = x_0 + y_0 \wedge (x-1) > 0\}$
$x := x - 1$
$\{xy + u + y = x_0 + y_0 \wedge x > 0\}$
$u := u + y$
$\{xy + u = x_0 + y_0 \wedge x > 0\}$
$\{xy + u = x_0 + y_0 \wedge x > 0\}$
$\{xy + u = x_0 + y_0 \wedge x <= 1 \wedge x > 0\} \implies \{y + u = x_0 \times y_0\}$
$y := y + u$
$\{y = x_0 \times y_0\}$

2➤ (P2) utilise la condition *pair*, hors celle-ci n'est pas définie en cours, ajoutez la règle d'inférence correspondante. On peut penser à deux variantes, l'une est triviale, l'autre récursive.

`Correction :`

Version triviale : $\dfrac{}{\sigma, \text{pair}(x) \looparrowright b}\ b \equiv x\%2 == 0$

Version recursive : $\dfrac{}{\sigma, \text{pair}(0) \looparrowright \mathbb{1}}\ \dfrac{}{\sigma, \text{pair}(1) \looparrowright \mathbb{0}}\ \dfrac{\sigma, \text{pair}(l) \looparrowright b}{\sigma, \text{pair}(k) \looparrowright b}$ l=k-2

Exercice 2 On continue ?

On donne les spécifications suivantes :

— $\{y > 0\}$**S1**$\{z = x \times y\}$
— $\{y > 0\}$**S2**$\{z = x^y\}$

1➤ Écrire S1 qui n'utilise que des additions et soustractions, puis S2.

`Correction :`

S1 : $i := 0$**;** $z := 0$**; while** i¡y **do** $i := i + 1$**;** $z := z + x$

2➤ Démontrer votre réponse à l'aide de la logique de Hoare.

Raphaël Charrondière
raphael.charrondiere@inria.fr

`Correction :`

$$
\begin{array}{l}
\{y > 0\} \implies \{0 = 0x \wedge 0 < y+1\} \\
i := 0 \\
\{0 = ix \wedge i < y+1\} \\
z := 0 \\
\{z = ix \wedge i < y+1\} \\
\textbf{while}(i < y) \\
\quad \{z = ix \wedge i < y+1 \wedge i < y\} \implies \{z + x = (i+1)x \wedge i+1 < y+1\} \\
\quad i := i+1 \\
\quad \{z + x = ix \wedge i < y+1\} \\
\quad z := z + x \\
\quad \{z = ix \wedge i < y+1\} \\
\{z = ix \wedge i < y+1 \wedge i >= y\} \implies \{z = x \times y\}
\end{array}
$$

Exercice 3 Thème et variations

On se propose d'ajouter deux commandes à IMP :
- **for** i=1 to <expr> **do** <commande>
- **repeat** <commande> **until** <condition>

1) Donner une(des) règle(s) d'inférence pour ces commandes.

`Correction :` Pour le for on a un cas de base et des itérations :

$$\frac{\sigma, i \looparrowright k \qquad \sigma, a \looparrowright k}{\sigma, \textbf{for } i = k \textbf{ to } a \textbf{ do } c \Downarrow \sigma}$$

$$\frac{\sigma, i \looparrowright k \qquad \sigma, a \looparrowright q \qquad \sigma, c \Downarrow \sigma' \qquad \sigma', \textbf{for } i = k_n \textbf{ to } a \textbf{ do } c \Downarrow \sigma''}{\sigma, \textbf{for } i = k \textbf{ to } a \textbf{ do } c \Downarrow \sigma''}\ k < q \wedge k_n = k+1$$

Pour le repeat, c'est encore plus simple

$$\frac{\sigma, c \Downarrow \sigma' \qquad \sigma', \textbf{while } \mathrm{b} \textbf{ do } c \looparrowright \sigma''}{\sigma, \textbf{repeat } c \textbf{ until } b \Downarrow \sigma''}$$

2) Donner une règle de Hoare pour ces commandes.

$$\frac{\vdash \{A \wedge i \in [1, a]\} c \{A\}}{\vdash \{A\} \textbf{for } i = 1 \textbf{ to } a \textbf{ do } c \{A \wedge i = a\}}$$

$$\frac{\vdash \{A\} c \{A \wedge B\}}{\vdash \{A\} \textbf{repeat } c \textbf{ until } b \{A \wedge B \wedge b\}}$$

Exercice 4 Amusons-nous un peu

On se propose d'introduire du non-déterminisme dans IMP. On introduit ainsi la commande **maybe do** <commande> **otherwise do** <commande>.

1) Donner la(les) règle(s) d'inférence pour la sémantique à grands pas.

`Correction :`

$$\frac{\sigma, c_1 \Downarrow \sigma'}{\sigma, \textbf{maybe do } c_1 \textbf{ otherwise do} c_2 \Downarrow \sigma'} \qquad \frac{\sigma, c_2 \Downarrow \sigma'}{\sigma, \textbf{maybe do } c_1 \textbf{ otherwise do} c_2 \Downarrow \sigma'}$$

2) Donner une règle de Hoare pour cette commande.

Raphaël Charrondière
raphael.charrondiere@inria.fr

`Correction :` $$\frac{\vdash \{A\}c_1\{B\} \quad \vdash \{A\}c_2\{B\}}{\vdash \{A\}\textbf{maybe do } c_1 \textbf{ otherwise do} c_2\{B\}}$$