L2/L3 MASS/MIAGE **Année universitaire 2013–2014**
Université Paris 1

Contrôle continu : preuves de programmes
Sujet 1

On rappelle que la notation $\texttt{prog}_{i \to j}$ désigne les lignes i à j (incluses) du programme `prog`.

Exercice 1

On étudie le programme suivant :

`prog`

```
1  x ← x + y + 1
2  y ← x - y + 1
3  x ← x - y + 3
```

Question 1 En utilisant notamment la règle de Floyd, déterminez une formule Q la plus forte possible telle que le triplet $\langle (x = u) \wedge (y = v) \rangle$ `prog` $\langle Q \rangle$ soit totalement correct.

Correction

On sait que la règle de Floyd donne la plus forte post condition pour un triplet de hoare. On commence par l'appliquer à la première ligne du programme, en notant $P = (x = u) \wedge (y = v)$, ce qui donne

$$\frac{}{\langle P \rangle \ \texttt{x} \leftarrow \texttt{x + y + 1} \ \langle (\exists x_0, (x = x_0 + y + 1) \wedge ((x_0 = u) \wedge (y = v))) \rangle} \leftarrow \text{Floyd}$$

On constate que P_1, la post condition obtenue, est sémantiquement équivalente à

$$P_1 \equiv (x = u + y + 1) \wedge (y = v) = Q_1.$$

En effet, en effectuant une substitution de x_0 par u, on a

$$\frac{\vdash (x = u + y + 1) \wedge ((u = u) \wedge (y = v))}{\vdash (\exists x_0, (x = x_0 + y + 1) \wedge ((x_0 = u) \wedge (y = v)))} \exists \text{ d}$$

et donc la valuation de la formule $(x = u + y + 1) \wedge ((u = u) \wedge (y = v))$ est toujours la même que celle de P_1 (et $u = u$ est tautologique). Notons que la justification formelle de cette équivalence sémantique n'est pas demandée.

On applique maintenant la règle de Floyd à la deuxième instruction du programme en partant de Q_1 (car on souhaite appliquer la règle de la séquence). On obtient

$$\frac{}{\langle Q_1 \rangle \ \texttt{y} \leftarrow \texttt{x - y + 1} \ \langle (\exists y_0, (y = x - y_0 + 1) \wedge ((x = u + y_0 + 1) \wedge (y_0 = v))) \rangle} \leftarrow \text{Floyd}$$

De la même façon que précédemment, on constante que la post condition obtenue, P_2 est sémantiquement équivalente à

$$P_2 \equiv ((y = x - v + 1) \wedge (x = u + v + 1)).$$

En manipulant les égalités, on obtient une formule plus pratique (car faisant apparaître moins souvent les variables manipulées par le programme) :

$$P_2 \equiv ((y = u + 2) \wedge (x = u + v + 1)) = Q_2.$$

On applique encore la règle de Floyd, en partant de Q_2 et pour la troisième instruction du programme, ce qui donne

$$\frac{}{\langle Q_2 \rangle \ \texttt{x} \leftarrow \texttt{x - y + 3} \ \langle (\exists x_1, (x = x_1 - y + 3) \wedge ((y = u + 2) \wedge (x_1 = u + v + 1)) \rangle} \leftarrow \text{Floyd}$$

On transforme de nouveau la post condition obtenue, P_3 en une formule plus simple mais sémantiquement équivalente, soit

$$P_3 \equiv ((x = v + 2) \wedge (y = u + 2)) = Q_3.$$

En utilisant la règle ;, on obtient

$$\frac{\dfrac{}{\langle P \rangle \ \texttt{x} \leftarrow \texttt{x + y + 1} \ \langle Q_1 \rangle} \leftarrow \text{Floyd} \quad \dfrac{}{\langle Q_1 \rangle \ \texttt{y} \leftarrow \texttt{x - y + 1} \ \langle Q_2 \rangle} \leftarrow \text{Floyd}}{\langle P \rangle \ \texttt{prog}_{1\to 2} \ \langle Q_2 \rangle} \ ;$$

puis

$$\frac{\langle P \rangle \ \texttt{prog}_{1\to 2} \ \langle Q_2 \rangle \quad \dfrac{}{\langle Q_2 \rangle \ \texttt{x} \leftarrow \texttt{x - y + 3} \ \langle Q_3 \rangle} \leftarrow \text{Floyd}}{\langle P \rangle \ \texttt{prog} \ \langle Q_3 \rangle} \ ;$$

Notons que nous avons implicitement utiliser le fait qu'une formule peut toujours être remplacée par une formule sémantiquement équivalente dans changer la nature du triplet. Il s'agit d'un cas particulier de la règle $\Rightarrow$ qu'on applique sans justification.

Comme à chaque étape de la preuve, la post condition est la plus forte possible, on en déduit que $Q = Q_3$.

Question 2 Déterminez de la même façon une formule R (non triviale) telle que le triplet $\langle\rangle$ `prog` $\langle R \rangle$ soit totalement correct.

Correction

Cette question était mal formulée. Comme nous l'avons vu en cours, l'application de la formule de Floyd à une précondition vide conduit à une tautologie. Dans sa formulation littérale, cette tautologie donne des indications sur ce que fait le programme, mais sans apporter de réelle preuve de son fonctionnement « correct ». L'objectif pédagogique était simplement de montrer qu'on ne peut pas enlever les quantificateurs existentiels dans cette situation. Il est facile de voir qu'on obtient d'abord

$$\frac{}{\langle\rangle \ \texttt{x} \leftarrow \texttt{x + y + 1} \ \left\langle \underbrace{(\exists x_0 (x = x_0 + y + 1))}_{P'_1} \right\rangle} \leftarrow \text{Floyd}$$

puis

$$\frac{}{\langle P_1' \rangle \ \texttt{y} \ \leftarrow \texttt{x - y + 1} \ \left\langle \underbrace{(\exists y_0 \exists x_0 (x = x_0 + y_0 + 1) \wedge (y = x - y_0 + 1))}_{P_2'} \right\rangle} \leftarrow \text{Floyd}$$

On constate aussi par simple manipulation algébrique que

$$P_2' \equiv (\exists y_0 \exists x_0 (x = x_0 + y_0 + 1) \wedge (y = x_0 + 2)) = Q_2'.$$

On obtient ensuite

$$\frac{}{\langle Q_2' \rangle \ \texttt{x} \ \leftarrow \texttt{x - y + 3} \ \left\langle \underbrace{(\exists x_1 \exists y_0 \exists x_0 (x = x_1 - y + 3) \wedge (x_1 = x_0 + y_0 + 1) \wedge (y = x_0 + 2))}_{P_3'} \right\rangle} \leftarrow \text{Floyd}$$

De nouvelles manipulations algébriques simples donnent

$$P_3' \equiv (\exists x_1 \exists y_0 \exists x_0 (x = y_0 + 2) \wedge (x_1 = x_0 + y_0 + 1) \wedge (y = x_0 + 2)) = Q_3'$$

En appliquant ensuite le même enchaînement de règles que dans la question 1, on voit que $R = Q_3'$ répond à la question.

Question 3 On cherche une précondition P la plus faible possible telle que le triplet

$$\langle P \rangle \ \texttt{prog} \ \langle (x = a) \wedge (y = b) \rangle$$

soit totalement correct. Montrez tout d'abord que le triplet

$$\langle (x - y + 3 = a) \wedge (y = b) \rangle \ \texttt{prog}_{3 \to 3} \ \langle (x = a) \wedge (y = b) \rangle$$

est totalement correct, **sans utiliser la règle de Floyd**.

Correction

Il suffit d'appliquer la règle de Hoare. On a

$$(x = a) \wedge (y = b)[x \leftarrow x - y + 3] = (x - y + 3 = a) \wedge (y = b),$$

et donc

$$\frac{}{\langle (x - y + 3 = a) \wedge (y = b) \rangle \ \texttt{x} \ \leftarrow \texttt{x - y + 3} \ \langle (x = a) \wedge (y = b) \rangle} \leftarrow \text{Hoare}$$

On sait que la règle de Hoare produit la pré condition la plus faible et on a donc répondu à la question.

Question 4 Pourquoi n'utilise-t-on pas la règle de Floyd dans la question précédente ?

Correction

La règle de Hoare produit la pré condition la plus faible, alors que la règle de Floyd produit la post condition la plus forte. On doit donc appliquer Hoare.

Question 5 En procédant comme à la question 3, montrez que le triplet

$$\langle (x = b-2) \wedge (y = a-2) \rangle \ \texttt{prog}_{1\to 3} \ \langle (x = a) \wedge (y = b) \rangle$$

est totalement correct, **sans utiliser la règle de Floyd**.

Correction

En appliquant une nouvelle fois la règle de Hoare, on obtient

$$\frac{}{\left\langle \underbrace{(x - (x - y + 1) + 3 = a) \wedge (x - y + 1 = b)}_{H_1} \right\rangle \ \texttt{y} \leftarrow \texttt{x - y + 1} \ \left\langle \underbrace{(x - y + 3 = a) \wedge (y = b)}_{H_0} \right\rangle} \leftarrow \text{Hoare}$$

De simples manipulations algébriques donnent

$$H_1 \equiv (y + 2 = a) \wedge (x - y + 1 = b) = J_1,$$

et une nouvelle application de Hoare conduit à

$$\frac{}{\left\langle \underbrace{(y + 2 = a) \wedge ((x + y + 1) - y + 1 = b)}_{H_2} \right\rangle \ \texttt{x} \leftarrow \texttt{x+y+1} \ \langle (y + 2 = a) \wedge (x - y + 1 = b) \rangle} \leftarrow \text{Hoare}$$

Il est clair qu'on a

$$\begin{aligned} H_2 &\equiv ((y + 2 = a) \wedge (x + 2 = b)), \\ &\equiv ((x = b - 2) \wedge (y = a - 2)) = J_2. \end{aligned}$$

On combine les différents résultats grâce à la règle ;, ce qui donne

$$\frac{\dfrac{}{\langle J_2 \rangle \ \texttt{x} \leftarrow \texttt{x + y + 1} \ \langle J_1 \rangle} \leftarrow \text{Hoare} \quad \dfrac{}{\langle J_1 \rangle \ \texttt{y} \leftarrow \texttt{x - y + 1} \ \langle H_0 \rangle} \leftarrow \text{Hoare}}{\langle J_2 \rangle \ \texttt{prog}_{1\to 2} \ \langle H_0 \rangle} \ ;$$

puis

$$\frac{\langle J_2 \rangle \ \texttt{prog}_{1\to 2} \ \langle H_0 \rangle \qquad \dfrac{}{\langle H_0 \rangle \ \texttt{x} \leftarrow \texttt{x - y + 3} \ \langle (x = a) \wedge (y = b) \rangle} \leftarrow \text{Hoare}}{\langle (x = b - 2) \wedge (y = a - 2) \rangle \ \texttt{prog} \ \langle (x = a) \wedge (y = b) \rangle} \ ;$$

Exercice 2

Soit le programme suivant :

prog

```
1  if (x > 0)
2    y ← 1/x
3  else
4    if (x < 0)
5      y ← -(1/x)
```

```
6     else
7       y ← 0
8     endif
9   end if
```

Dans ce programme, `x` et `y` sont des variables réelles et la division `1/x` est supposée donner un résultat exact (c'est-à-dire que la formule $\forall x\ ((x \neq 0) \Rightarrow (x(1/x) = 1))$ est une tautologie).

Question 1 Déterminez une formule Q non triviale (non tautologique en particulier) telle que le triplet suivant soit totalement correct

$$\langle x \leq 0 \rangle\ \texttt{prog}_{4\to 8}\ \langle Q \vee ((x = 0) \wedge (y = 0)) \rangle$$

Correction

L'objectif est donc de traiter le `else` correspondant au premier `if`. Pour cela, on étudie l'instruction `y ← -(1/x)`, sous les conditions $x \leq 0$ (hypothèse de la question) et $x < 0$ (hypothèse du second `if`). En appliquant Floyd, on obtient

$$\frac{}{\langle (x \leq 0) \wedge (x < 0) \rangle\ \texttt{y ← -(1/x)}\ \left\langle \underbrace{(\exists y_0, (x \leq 0) \wedge (x < 0) \wedge (y = -(1/x)))}_{P_1} \right\rangle} \leftarrow \text{Floyd}$$

Il est évident que $P_1 \equiv (x < 0) \wedge (y = -(1/x)) = Q_1$ car y_0 n'intervient pas dans la sous-formule concernée et en supprimant la condition redondante $(x \leq 0)$.

En appliquant Floyd à l'autre branche du second `if`, on obtient

$$\frac{}{\langle (x \leq 0) \wedge (\neg(x < 0)) \rangle\ \texttt{y ← 0}\ \left\langle \underbrace{(\exists y_0, (x \leq 0) \wedge (\neg(x < 0)) \wedge (y = 0))}_{P_2} \right\rangle} \leftarrow \text{Floyd}$$

Il aussi évident que $P_2 \equiv ((x = 0) \wedge (y = 0)) = Q_2$ en simplifiant les conditions sur x et en constatant que y_0 n'intervient pas dans la sous-formule.

On sait que $Q_1 \vdash Q_1 \vee Q_2$ et que $Q_2 \vdash Q_1 \vee Q_2$, donc on peut appliquer la règle $\Rightarrow$ (deux fois) ce qui donne

$$\frac{\dfrac{}{\langle (x \leq 0) \wedge (x < 0) \rangle\ \texttt{y ← -(1/x)}\ \langle Q_1 \rangle} \leftarrow \text{Floyd} \qquad Q_1 \vdash Q_1 \vee Q_2}{\langle (x \leq 0) \wedge (x < 0) \rangle\ \texttt{y ← -(1/x)}\ \langle Q_1 \vee Q_2 \rangle} \Rightarrow$$

et

$$\frac{\dfrac{}{\langle (x \leq 0) \wedge (\neg(x < 0)) \rangle\ \texttt{y ← 0}\ \langle Q_2 \rangle} \leftarrow \text{Floyd} \qquad Q_1 \vdash Q_1 \vee Q_2}{\langle (x \leq 0) \wedge (\neg(x < 0)) \rangle\ \texttt{y ← -(1/x)}\ \langle Q_1 \vee Q_2 \rangle} \Rightarrow$$

On peut maintenant appliquer la règle du « if then » qui donne

$$\frac{\langle (x \leq 0) \wedge (x < 0) \rangle \ \texttt{prog}_{5\to 5} \ \langle Q_1 \vee Q_2 \rangle \qquad \langle (x \leq 0) \wedge (\neg(x < 0)) \rangle \ \texttt{prog}_{7\to 7} \ \langle Q_1 \vee Q_2 \rangle}{\langle (x \leq 0) \rangle \ \texttt{prog}_{4\to 8} \ \langle Q_1 \vee Q_2 \rangle} \text{ if then}$$

La condition Q recherchée est donc $Q = Q_1 = (x < 0) \wedge (y = -(1/x))$.

Question 2 Montrez que le $\langle\rangle \ \texttt{prog} \ \langle ((x \neq 0) \wedge (y|x| = 1)) \vee ((x = 0) \wedge (y = 0)) \rangle$ est totalement correct.

Correction

On procède selon une stratégie similaire à celle utilisée dans la question précédente, sachant que le `else` du `if` est déjà traité. On étudie donc d'abord l'instruction 2, `y ← 1/x`. En appliquant a règle de Floyd, on a

$$\frac{}{\langle (x > 0) \rangle \ \texttt{y} \ \leftarrow \texttt{1/x} \ \left\langle \underbrace{(\exists y_0, (x > 0) \wedge (y = 1/x))}_{P_3} \right\rangle} \leftarrow \text{Floyd}$$

Il est clair que $P_3 \equiv ((x > 0) \wedge (y = 1/x)) = Q_3$. On a aussi $Q_3 \vdash (Q_1 \vee Q_2 \vee Q_3)$ où Q_2 et Q_3 sont les formules obtenues dans la question précédente. En appliquant la règle $\Rightarrow$ on obtient

$$\frac{\dfrac{}{\langle (x > 0) \rangle \ \texttt{y} \ \leftarrow \texttt{1/x} \ \langle Q_3 \rangle} \leftarrow \text{Floyd} \qquad Q_3 \vdash Q_1 \vee Q_2 \vee Q_3}{\langle (x > 0) \rangle \ \texttt{y} \ \leftarrow \texttt{1/x} \ \langle Q_1 \vee Q_2 \vee Q_3 \rangle} \Rightarrow$$

De la même façon on a $(Q_1 \vee Q_2) \vdash (Q_1 \vee Q_2 \vee Q_3)$ et $(x \leq 0) \equiv (\neg(x > 0))$. En appliquant de nouveau la règle implique combinée aux résultats de la question précédente, on a

$$\frac{(\neg(x > 0)) \vdash (x \leq 0) \qquad \langle (x \leq 0) \rangle \ \texttt{prog}_{4\to 8} \ \langle Q_1 \vee Q_2 \rangle \qquad (Q_1 \vee Q_2) \vdash Q_1 \vee Q_2 \vee Q_3}{\langle (\neg(x > 0)) \rangle \ \texttt{prog}_{4\to 8} \ \langle Q_1 \vee Q_2 \vee Q_3 \rangle} \Rightarrow$$

En appliquant la règle du `if`, on obtient

$$\frac{\langle (x > 0) \rangle \ \texttt{y} \ \leftarrow \texttt{1/x} \ \langle Q_1 \vee Q_2 \vee Q_3 \rangle \qquad \langle (\neg(x > 0)) \rangle \ \texttt{prog}_{4\to 8} \ \langle Q_1 \vee Q_2 \vee Q_3 \rangle}{\langle\rangle \ \texttt{prog} \ \langle Q_1 \vee Q_2 \vee Q_3 \rangle} \text{ if then}$$

Il reste donc à étudier $Q_1 \vee Q_2 \vee Q_3$. On constate que Q_2 est l'une des deux parties de la post condition à prouver. Reste donc à étudier $Q_1 \vee Q_3$. On constante que

$$\begin{aligned}
Q_1 &= (x < 0) \wedge (y = -(1/x)) \\
&\equiv (x < 0) \wedge (xy = -x(1/x)) && \text{car } x \neq 0 \\
&\equiv (x < 0) \wedge (-xy = x(1/x)) \\
&\equiv (x < 0) \wedge (-xy = 1)) && \text{en utilisant l'axiome} \\
&\equiv (x < 0) \wedge (|x|y = 1)) && \text{car } |x| = -x \text{ quand } x < 0
\end{aligned}$$

On applique un raisonnement similaire sur Q_3 qui donne

$$\begin{aligned} Q_3 &= (x > 0) \wedge (y = 1/x) \\ &\equiv (x > 0) \wedge (xy = x(1/x)) \\ &\equiv (x > 0) \wedge (xy = 1) \\ &\equiv (x > 0) \wedge (|x|y = 1) \end{aligned}$$

Enfin, on a

$$\begin{aligned} Q_1 \vee Q_3 &\equiv ((x < 0) \wedge (|x|y = 1)) \vee ((x > 0) \wedge (|x|y = 1)), \\ &\equiv (((x < 0) \vee (x > 0)) \wedge (|x|y = 1)), \\ &\equiv ((x \neq 0) \wedge (|x|y = 1)), \end{aligned}$$

ce qui permet de conclure.

Exercice 3

On étudie le programme suivant :

prog

```
1 y ← x
2 u ← 1
3 r ← 0
4 while(y >= 2)
5   r ← r + u*(y mod 2)
6   u ← 2*u
7   y ← y div 2
8 end while
```

Dans ce programme, toutes les variables sont des entiers de $\mathbb{Z}$. L'opérateur `*` désigne la multiplication dans $\mathbb{Z}$, L'opération `mod` donne le reste de la division euclidienne alors que `div` donne le quotient. La formule suivante est une tautologie

$$\forall z \in \mathbb{Z}\ (z = 2(z \texttt{ div } 2) + (z \texttt{ mod } 2)).$$

On pourra aussi utiliser la tautologie suivante :

$$\forall z \in \mathbb{Z}\ ((z \texttt{ mod } 2) \leq 1) \wedge ((z \texttt{ mod } 2) \geq 0).$$

Question 1 Soit F la formule logique $(uy + r = x) \wedge (y \geq 1)$. Déterminez une précondition P telle que le triplet $\langle P \rangle\ \texttt{prog}_{1\to 3}\ \langle F \rangle$ soit totalement correct.

Correction

On procède en utilisant la règle de Hoare. On a

$$\frac{}{\left\langle \underbrace{(uy + 0 = x) \wedge (y \geq 1)}_{F_1} \right\rangle\ \texttt{r} \leftarrow \texttt{0}\ \langle F \rangle} \leftarrow \text{Hoare}$$

Il est clair que $F_1 \equiv (uy = x) \wedge (y \geq 1)$. On applique de nouveau la règle de Hoare, ce qui donne

$$\frac{}{\langle (y = x) \wedge (y \geq 1) \rangle \ \texttt{u} \leftarrow \texttt{1} \ \langle (uy = x) \wedge (y \geq 1) \rangle} \leftarrow \text{Hoare}$$

Une troisième application donne

$$\frac{}{\left\langle \underbrace{(x = x) \wedge (x \geq 1)}_{F_2} \right\rangle \ \texttt{y} \leftarrow \texttt{x} \ \langle (y = x) \wedge (y \geq 1) \rangle} \leftarrow \text{Hoare}$$

Comme $F_2 \equiv (x \geq 1)$, une première application de la règle ; donne

$$\frac{\dfrac{}{\langle (x \geq 1) \rangle \ \texttt{y} \leftarrow \texttt{x} \ \langle (y = x) \wedge (y \geq 1) \rangle} \leftarrow \text{Hoare} \qquad \dfrac{}{\langle (y = x) \wedge (y \geq 1) \rangle \ \texttt{u} \leftarrow \texttt{1} \ \langle (uy = x) \wedge (y \geq 1) \rangle} \leftarrow \text{Hoare}}{\langle (x \geq 1) \rangle \ \texttt{prog}_{1 \to 2} \ \langle (uy = x) \wedge (y \geq 1) \rangle} \ ;$$

puis une autre application donne

$$\frac{\langle (x \geq 1) \rangle \ \texttt{prog}_{1 \to 2} \ \langle (uy = x) \wedge (y \geq 1) \rangle \qquad \dfrac{}{\langle (uy = x) \wedge (y \geq 1) \rangle \ \texttt{r} \leftarrow \texttt{0} \ \langle F \rangle} \leftarrow \text{Hoare}}{\langle (x \geq 1) \rangle \ \texttt{prog}_{1 \to 3} \ \langle F \rangle} \ ;$$

Donc la condition cherchée est $P = (x \geq 1)$.

Question 2 Montrez que le triplet suivant est totalement correct

$$\langle F \wedge (y \geq 2) \rangle \ \texttt{prog}_{5 \to 6} \ \langle (uy + 2r = 2x + u(y \ \texttt{mod} \ 2)) \wedge (y \geq 2) \rangle \, .$$

Correction

On procède encore par la règle de Hoare en notant $G = (uy + 2r = 2x + u(y \ \texttt{mod} \ 2)) \wedge (y \geq 2)$. On a

$$\frac{}{\langle G_1 \rangle \ \texttt{u} \leftarrow \texttt{u*2} \ \langle G \rangle} \leftarrow \text{Hoare}$$

avec

$$\begin{aligned} G_1 &= G[u \leftarrow 2u] \\ &= (2uy + 2r = 2x + 2u(y \ \texttt{mod} \ 2)) \wedge (y \geq 2) \\ &\equiv (uy + r = x + u(y \ \texttt{mod} \ 2)) \wedge (y \geq 2) = G_2 \end{aligned}$$

On a ensuite, en appliquant Hoare de nouveau,

$$\frac{}{\langle G_3 \rangle \ \texttt{r} \ \leftarrow \texttt{r + u*(y mod 2)} \ \langle G_2 \rangle} \leftarrow \text{Hoare}$$

avec

$$\begin{aligned} G_3 &= G_2[r \leftarrow r + u(y \texttt{ mod } 2)] \\ &= (uy + r + u(y \texttt{ mod } 2) = x + u(y \texttt{ mod } 2)) \wedge (y \geq 2) \\ &\equiv (uy + r = x) \wedge (y \geq 2) = G_4 \end{aligned}$$

Donc d'après la règle de la séquence

$$\frac{\dfrac{}{\langle G_4 \rangle \ \texttt{prog}_{5\to5} \ \langle G_2 \rangle} \leftarrow \text{Hoare} \qquad \dfrac{}{\langle G_2 \rangle \ \texttt{prog}_{6\to6} \ \langle G \rangle} \leftarrow \text{Hoare}}{\langle (uy + r = x) \wedge (y \geq 2) \rangle \ \texttt{prog}_{5\to6} \ \langle G \rangle} \ ;$$

Or, il est clair que $F \wedge (y \geq 2) \vdash (uy + r = x) \wedge (y \geq 2)$ et donc par l'application de la règle implique, on a

$$\frac{F \wedge (y \geq 2) \vdash (uy + r = x) \wedge (y \geq 2) \qquad \langle (uy + r = x) \wedge (y \geq 2) \rangle \ \texttt{prog}_{5\to6} \ \langle G \rangle}{\langle F \wedge (y \geq 2) \rangle \ \texttt{prog}_{5\to6} \ \langle G \rangle} \Rightarrow$$

Ce qui permet de conclure.

Question 3 Montrez que le triplet suivant est totalement correct

$$\langle (uy + 2r = 2x + u(y \texttt{ mod } 2)) \wedge (y \geq 2) \rangle \ \texttt{prog}_{7\to7} \ \langle F \rangle \, .$$

La preuve nécessite l'utilisation de la tautologie donnée ci-dessus.

Correction

On utilise encore une fois la règle de Hoare qui donne

$$\frac{}{\left\langle \underbrace{(u(y \texttt{ div } 2) + r = x) \wedge ((y \texttt{ div } 2) \geq 1)}_{H} \right\rangle \ \texttt{y} \leftarrow \texttt{y div 2} \ \langle F \rangle} \leftarrow \text{Hoare}$$

En appliquant la tautologie à y, on obtient $y = 2(y \texttt{ div } 2) + (y \texttt{ mod } 2)$ et donc

$$2(y \texttt{ div } 2) = y - (y \texttt{ mod } 2).$$

Or

$$H \equiv (2u(y \texttt{ div } 2) + 2r = 2x) \wedge (2(y \texttt{ div } 2) \geq 2),$$

et donc en substituant, on obtient

$$\begin{aligned} H &\equiv (uy - u(y \texttt{ mod } 2) + 2r = 2x) \wedge (y - (y \texttt{ mod } 2) \geq 2), \\ &\equiv (uy + 2r = 2x + u(y \texttt{ mod } 2)) \wedge (y \geq 2 + (y \texttt{ mod } 2)) \end{aligned}$$

L'inégalité $y \geq 2 + (y \texttt{ mod } 2)$ doit être traitée avec attention. En effet, comme on a obtenu H par la règle de Hoare, on ne peut que *renforcer* la condition, pas la relâcher. Or, constate que si $y \geq 2$, on a alors $y \geq 2 + (y \texttt{ mod } 2)$. En effet, comme y est entier, $(y \geq 2) \Rightarrow (y = 2) \vee (y \geq 3)$ est une tautologie. Or $(y = 2) \Rightarrow (y \geq 2 + (y \texttt{ mod } 2))$ est aussi une tautologie puisque quand $y = 2$, $(y \texttt{ mod } 2) = 0$. De plus $(y \geq 3) \Rightarrow y \geq 2 + (y \texttt{ mod } 2)$ est aussi une tautologie car $\forall y \in \mathbb{Z}$ $(y \texttt{ mod } 2) \leq 1$ est aussi une tautologie.

Donc finalement, $\forall y \in \mathbb{Z}\ (y \geq 2) \Rightarrow (y \geq 2 + (y \texttt{ mod } 2))$ est une tautologie. On a donc

$$(uy + 2r = 2x + u(y \texttt{ mod } 2)) \wedge (y \geq 2) \vdash H,$$

ce qui donne bien le résultat recherché en appliquant la règle implique.

Question 4 Déterminez Q telle que le triplet $\{F\}\ \texttt{prog}_{4\to 8}\ \{Q\}$ soit *partiellement* correct. Quelles sont les valeurs possibles pour `y` quand Q est vraie ?

Correction

On cherche à appliquer la règle du while partiel. Il faut donc identifier un invariant. Or, en appliquant la règle de la séquence aux résultats des deux questions précédentes, on obtient, en notant $G = (uy + 2r = 2x + u(y \texttt{ mod } 2)) \wedge (y \geq 2)$

$$\frac{\langle F \wedge (y \geq 2)\rangle\ \texttt{prog}_{5\to 6}\ \langle G\rangle \qquad \langle G\rangle\ \texttt{prog}_{7\to 7}\ \langle F\rangle}{\langle F \wedge (y \geq 2)\rangle\ \texttt{prog}_{5\to 7}\ \langle F\rangle}\ ;$$

Ceci montre que F est un invariant pour la boucle $\texttt{prog}_{4\to 8}$. On peut donc appliquer la règle correspondante qui donne

$$\frac{\langle F \wedge (y \geq 2)\rangle\ \texttt{prog}_{5\to 7}\ \langle F\rangle}{\{F\}\ \texttt{prog}_{4\to 8}\ \left\{\underbrace{(F \wedge (\neg(y \geq 2)))}_{Q_1}\right\}}\ \text{while partiel}$$

On a

$$\begin{aligned} Q_1 &= (uy + r = x) \wedge (y \geq 1) \wedge (\neg(y \geq 2)) \\ &\equiv (uy + r = x) \wedge (y \geq 1) \wedge (y < 2) \\ &\equiv (uy + r = x) \wedge (y = 1) && \text{car } y \text{ est entier} \\ &\equiv (u + r = x) \wedge (y = 1) = Q \end{aligned}$$

On constate que seul $y = 1$ peut rendre Q vraie.

Question 5 Proposez un *variant* pour la boucle $\texttt{prog}_{4\to 8}$ et déterminez R telle que le triplet $\langle R\rangle\ \texttt{prog}_{4\to 8}\ \langle Q\rangle$ soit *totalement* correct avec la même post-condition Q qu'à la question précédente.

Correction

On sait qu'un variant est un terme dont la valeur décroît strictement à chaque itération de la boucle, tout en restant positif ou nul. Ici, on constate que la variable y diminue à chaque tour. On peut donc poser $T := y$ comme invariant potentiel. On utilise alors la règle de Floyd qui donne

$$\frac{}{\langle (y = n) \wedge (y \geq 2)\rangle\ \texttt{r } \leftarrow \texttt{r + u*(y mod 2)}\ \langle J_1\rangle} \leftarrow \text{Floyd}$$

avec $J_1 = (\exists r_0, (y = n) \wedge (y \geq 2) \wedge (r = r_0 + u * (y \texttt{ mod } 2)))$. Il est clair que $J_1 \vdash (y =$

$n) \wedge (y \geq 2)$, et donc que $\langle (y = n) \wedge (y \geq 2) \rangle$ `r ←r + u*(y mod 2)` $\langle (y = n) \wedge (y \geq 2) \rangle$ est totalement correct (par application de la règle implique). On obtient de la même façon

$$\frac{}{\langle (y = n) \wedge (y \geq 2) \rangle \texttt{ u } \leftarrow \texttt{2*u } \langle J_2 \rangle} \leftarrow \text{Floyd}$$

avec $J_2 = (\exists u_0, (y = n) \wedge (y \geq 2) \wedge (u = 2u_0))$. Comme $J_2 \vdash (y = n) \wedge (y \geq 2)$, on trouve que $\langle (y = n) \wedge (y \geq 2) \rangle$ `u ←2*u` $\langle (y = n) \wedge (y \geq 2) \rangle$ est totalement correct (toujours en appliquant la règle implique).
De plus

$$\frac{}{\langle (y = n) \wedge (y \geq 2) \rangle \texttt{ y } \leftarrow \texttt{y div 2 } \langle J_3 \rangle} \leftarrow \text{Floyd}$$

avec

$$\begin{aligned} J_3 &= \exists y_0,\ (y_0 = n) \wedge (y_0 \geq 2) \wedge (y = (y_0 \texttt{ div } 2)) \\ &\equiv (n \geq 2) \wedge (y = (n \texttt{ div } 2)) \end{aligned}$$

Or, on sait que $(n \geq 2) \Rightarrow ((n \texttt{ div } 2) \geq 1)$ est une tautologie, de même que $(n \geq 1) \Rightarrow ((n \texttt{ div } 2) < n)$. On en déduit donc que $J_3 \vdash (y < n) \wedge (y \geq 0)$ et donc que $\langle (y = n) \wedge (y \geq 2) \rangle$ `u ←2*u` $\langle (y < n) \wedge (y \geq 0) \rangle$ est totalement correct.
En appliquant deux fois la règle de la séquence, on en déduit que

$$\langle (y = n) \wedge (y \geq 2) \rangle \ \texttt{prog}_{5 \to 7} \ \langle (y < n) \wedge (y \geq 0) \rangle$$

est totalement correct. En appliquant la règle du et à ce triplet et au triplet prouvé dans la section précédente, on en déduit que le triplet

$$\langle (y = n) \wedge (y \geq 2) \wedge F \rangle \ \texttt{prog}_{5 \to 7} \ \langle (y < n) \wedge (y \geq 0) \wedge F \rangle$$

est totalement correct. Ceci montre que $T := y$ est un variant de la boucle. On peut donc alors appliquer la règle du while total qui donne

$$\frac{\langle (F \wedge (y \geq 2) \wedge (y = n)) \rangle \ \texttt{prog}_{5 \to 7} \ \langle (y < n) \wedge (y \geq 0) \wedge F \rangle}{\langle F \rangle \ \texttt{prog}_{4 \to 8} \ \langle (F \wedge (\neg (y \geq 2))) \rangle} \text{ while total}$$

Par le même raisonnement que dans la question précédente, on constate que la post condition peut remplacée par $Q = (u + r = x) \wedge (y = 1)$.

Question 6 Déterminez S telle que le triplet $\langle S \rangle$ `prog` $\langle Q \rangle$ soit *totalement* correct (toujours avec la même post-condition).

Correction

On applique la règle de la séquence aux résultats de la question 1 et de la question précédente, ce qui donne

$$\frac{\langle (x \geq 1) \rangle \ \texttt{prog}_{1 \to 3} \ \langle F \rangle \qquad \langle F \rangle \ \texttt{prog}_{4 \to 8} \ \langle Q \rangle}{\langle (x \geq 1) \rangle \ \texttt{prog} \ \langle Q \rangle} \ ;$$

et donc $S = (x \geq 1)$.

Question 7 Expliquez brièvement comment on pourrait montrer qu'à la fin du programme, `u` contient une puissance de 2.

Correction

Il faut pour cela utiliser un invariant de boucle. On pourrait par exemple inclure dans F une sous-formule additionnelle de la forme $(\exists k,\ u = 2^k)$. Cette sous-formule est clairement vraie avant la boucle puisque $u = 1$ d'après l'instruction numéro 2. Il faudrait ensuite montrer que les lignes 5 à 7 laissent cette sous-formule vraie. Comme u est multipliée par 2, cela ne pose pas de problème particulier. Plus formellement, on doit utiliser la règle de Floyd sous la forme

$$\frac{}{\left\langle (\exists k,\ u = 2^k) \right\rangle \ \texttt{u} \ \leftarrow \texttt{2*u} \ \left\langle (\exists u_0,\ (\exists k,\ u_0 = 2^k) \wedge (u = 2u_0)) \right\rangle} \leftarrow \text{Floyd}$$

On constate que de simples manipulations logique permet de déduire de la post condition $(\exists k,\ u = 2^k)$ et donc d'obtenir que $(F \wedge (\exists k,\ u = 2^k))$ reste un invariant de la boucle.