Modélisation et vérification des systèmes informatiques

Dominique Méry

9 septembre 2015

Table des matières

1 Série 1 Annotation, modélisation et vérification

Exercice 1 *Définir les conditions de vérification de la correction partielle pour les structures suivantes. Définir un modèle TLA$^+$ pour vérifier la bonne annotation.*

Question 1.1

$\ell 1 : \{P_{\ell 1}(x, y)\}$
$\quad x := x{+}y{+}7;$
$\ell 2 : \{P_{\ell 2}(x, y)\}$

Question 1.2

$\ell : \{P_{\ell}(x, y)\}$
$\quad x, y := y, x;$
$\ell' : \{P_{\ell'}(x, y)\}$

Exercice 2 *Déterminer les conditions de vérification pour la structure de boucle bornée.*

Question 2.1 *On suppose que S ne modifie pas i.*

$\ell_1 : \{P_{\ell_1}(x)\}$
$\quad$ ***FOR*** $i := 1$ ***TO*** n ***DO***
$\quad\quad \ell_2 : \{P_{\ell_2}(i, x)\}$
$\quad\quad\quad S(x);$
$\quad\quad \ell_3 : \{P_{\ell_3}(i, x)\}$
$\quad$ ***ENDFOR***
$\ell_4 : \{P_{\ell_4}(x)\}$

Exercice 3

Montrer que chaque annotation est correcte ou incorrecte selon les conditions de vérifications énoncées comme suit
$\forall x, y, , x', y'.P_{\ell}(x, y) \ \wedge \ cond_{\ell,\ell'}(x, y) \ \wedge \ (x', y') = f_{\ell,\ell'}(x, y) \Rightarrow P_{\ell'}(x', y')$

– $\ell_1 : x = 10 \ \wedge \ y = z{+}x \ \wedge z = 2{\cdot}x$
$y := z{+}x$
$\ell_2 : x = 10 \ \wedge \ y = x{+}2{\cdot}10$

– $\ell_1 : x = 1 \ \wedge \ y = 12$
$x := 2{\cdot}y$
$\ell_2 : x = 1 \ \wedge \ y = 24$

– *On suppose que p est un nombre premier :*

$\ell_1 : x = 2^p \ \wedge \ y = 2^{p+1} \ \wedge \ x{\cdot}y = 2^{2{\cdot}p+1}$
$x := y{+}x{+}2^x$
$\ell_2 : x = 5{\cdot}2^p \ \wedge \ y = 2^{p+1}$

– $\ell_1 : x = 11 \ \wedge \ y = 13$
$z := x; x := y; y := z;$
$\ell_2 : x = 26/2 \ \wedge \ y = 33/3$

Exercice 4 Question 4.1 *Compléter l'algorithme en l'annotant.*

Question 4.2 *Vérifier la bonne annotation*

Question 4.3 *Enoncer et vérifier la correction partielle*

Exercice 5 *Il s'agit d'étudier et d'annoter le programme proposé en vu d'obtenir sa correction partielle (c'est-à-dire sans la preuve de terminaison). On appelle état un ensemble de valeurs précises (spécifié par un prédicat) des variables du programme, nous allons considérer une*

precondition : $x = a \wedge y = b \wedge a, b \in \mathbb{N}$
postcondition : $z = max(a, b)$

$\ell_0 : \{\ldots\}$
if $x < y$ **then**
  $\ell_1 : \{\ldots\}$
  $z := y$;
  $\ell_2 : \{\ldots\}$
else
  $\ell_3 : \{\ldots\}$
  $z := x$;
  $\ell_4 : \{\ldots\}$
;
$\ell_5 : \{\ldots\}$

Algorithme 1: MAX annotée

étiquette (ℓ) entre chaque instruction du programme considéré. On appelle une annotation le prédicat décrivant les valeurs possibles des variables pour un état du programme. Cette annotation est notée : $P_\ell(v)$ et exprime la propriété satisfaite par la variable v en ℓ.
On vous demande :

1. *de dessiner le graphe de transition entre les étiquettes*
2. *d'annoter toutes les étiquettes du programme*
3. *de proposer un modèle TLA$^+$ pour vérifier les annotations et la correction partielle*

precondition : $x = x_0 \wedge x_0 \in \mathbb{N}$
postcondition : $x = 0$

$\ell_0 : \{\ldots\}$
while $0 < x$ **do**
  $\ell_1 : \{\ldots\}$
  $x := x-1$;
  $\ell_2 : \{\ldots\}$
;
$\ell_3 : \{\ldots\}$

Algorithme 2: $EX2$ annotée