TD de *Génie Logiciel Avancé* n° 2
(Correction)

La méthode B

Exercice 1 (Les relation en B) *Le langage B offre un large ensemble d'opérations sur les relations. Le tableau ci-dessous définit quatre opérations sur les relations.*

Condition	*Opération*	*Définition*
$r_1 \in t \leftrightarrow u$ *et* $r_2 \in t \leftrightarrow v$	$r_1 \otimes r_2$	$\{(x,(y,z)) \mid (x,(y,z)) \in t \times (u \times v) \wedge (x,y) \in r_1 \wedge (x,z) \in r_2\}$
$r_1 \in t \leftrightarrow u$ *et* $r_2 \in v \leftrightarrow w$	$r_1 \| r_2$	$\{((x,z),(y,a)) \mid ((x,z),(y,a)) \in (t \times v) \times (u \times w) \wedge (x,y) \in r_1 \wedge (z,a) \in r_2\}$
$r \in s \leftrightarrow s$	r^n	$r^n = r; r^{n-1} \quad r^0 = \mathit{id}(s)$
$r \in s \leftrightarrow s$	r^*	$\bigcup n \cdot (n \in \mathbb{N} \mid r^n)$
$r \in s \leftrightarrow s$	r^+	$r; r^*$

Étant données les relations R_1, R_2, R_3 ci-dessous,

$$\begin{aligned} R_1 &= \{(0,2),(1,5),(2,5),(2,7)\} \\ R_2 &= \{(0,0),(2,-1),(5,8),(6,9)\} \\ R_3 &= \{(0,2),(1,0),(2,1)\} \end{aligned}$$

calculez les relations définies par les expressions suivantes:

$$\begin{aligned} R_1 \otimes R_2 &= \ldots \\ R_1 \| R_2 &= \ldots \end{aligned} \qquad \begin{aligned} R_3^0 &= \ldots \\ R_3^1 &= \ldots \\ R_3^2 &= \ldots \\ R_3^3 &= \ldots \\ R_3^4 &= \ldots \\ R_3^n &= \ldots \\ R_3^* &= \ldots \end{aligned}$$

Correction :
Faite en TD.

Exercice 2 (Un exemple de spécification en B) *En utilisant les expressions d'ensemble en B et la logique de premier ordre, spécifier l'ensemble des forêts* `foret` *construites avec des arbres binaires de l'ensemble* `arbre` *et défini informellement par:*

F1: *Un arbre est un ensemble de noeuds reliés entre eux tel que tout noeud a au plus un prédécesseur direct.*

Correction :

$ARBRES = NODES \nrightarrow NODES$

L'arbre vide est la fonction vide. On ne peut pas représenter l'arbre avec un seul noeud et aucune flèche, mais celui ci est représentable "à la C", avec un noeud pointant sur deux fils "nil".

F2: *Une forêt est un ensemble d'arbres sans partage de noeuds.*

Correction :

$FORETS = \mathbb{P}(ARBRES)$
$f : FORETS \land a1 : ARBRES \land a2 : ARBRES \Rightarrow$
$a1 \in f \land a2 \in f \Rightarrow$
$(dom(a1) \cup ran(a1)) \cap (dom(a2) \cup ran(a2)) = \emptyset$

F3: *Uniquement la racine de l'arbre n'a pas de prédécesseur.*

Correction :

$a : ARBRES \Rightarrow$
$dom(a) \neq \emptyset \Rightarrow$
$card(\{ r \mid r \in NODES \land not(r \in dom(a)) \land r \in ran(a) \}) = 1$

F4: *Un noeud dans un arbre binaire a zéro ou deux successeurs directs.*

Correction :

$ARBRESBINAIRES \subseteq ARBRES$
$a : ARBRESBINAIRES \land n : NODES \Rightarrow$
$n \in ran(a) \Rightarrow$
$card(\{ x \mid (x,n) \in a \}) = 2$

F5: *Un noeud feuille n'a pas de successeurs.*

Correction :

$feuilles: ARBRES \rightarrow \mathbb{P}(NODES)$
$a : ARBRES \Rightarrow$
$feuilles(a) = dom(a) \setminus ran(a)$

F6: *Les ancêtres d'un noeud n dans un arbre sont les noeuds entre son prédécesseur et la racine de l'arbre.*

Correction :

$ancetre : ARBRES \rightarrow (NODES \leftrightarrow NODES)$
$a : ARBRES \Rightarrow$
$(x,y) \in ancetre(a) \Leftrightarrow (x,y) \in a^{+}$

F7: *Les successeurs d'un noeud n dans un arbre sont les noeuds dont n est parmi ses ancêtres.*

Correction :

$successeur : ARBRES \rightarrow (NODES \leftrightarrow NODES)$
$a : ARBRES \Rightarrow$
$(m,n) \in successeur(a) \Leftrightarrow (n,m) \in ancetre(a)$

Alternative:

$successeur : ARBRES \rightarrow (NODES \leftrightarrow NODES)$
$a : ARBRES \Rightarrow$
$successeur(a) = ancetre(a)^{-1}$

F8: *Les noeuds frères dans un arbre ont le même prédécesseur direct.*

Correction :

frere : $ARBRES \rightarrow (NODES \leftrightarrow NODES)$
$a : ARBRES$, $(x,y) \in frere(a) \Leftrightarrow x \neq y \ \wedge \ \exists z \ (x,z) \in a \wedge (y,z) \in a$

Alternative:

frere : $ARBRES \rightarrow (NODES \leftrightarrow NODES)$
$a : ARBRES$, $(x,y) \in frere(a) \Leftrightarrow x \neq y \ \wedge \ a(x) = a(y)$

Exercice 3 (Preuves de Programmes) *Le programme suivant calcule x^y pour des valeurs entières $x > 0$ et $y \geq 0$. Prouver :*

$\{x > 0 \wedge y \geq 0\}$

```
e := x ; p := y ; z := 1 ;
while p > 0 do
        if p mod 2 <> 0 then p := p-1; z := z * e;
        e := e * e ;
        p := p div 2 ;
done;
```

$\{z = x^y\}$

Correction :

1. On prouve avec l'axiome de substitution et la règle de composition

$$\begin{array}{l} \{x > 0 \wedge y \geq 0\} \\ e := x; p := y; z := 1; \\ \{e > 0 \wedge p \geq 0 \wedge z = 1 \wedge e = x \wedge p = y\} \end{array}$$

$$\begin{array}{ll} \{x > 0 \wedge y \geq 0\} & \equiv \\ \{x > 0 \wedge y \geq 0 \wedge x = x\} & \equiv \\ e := x; & \\ \{e > 0 \wedge y \geq 0 \wedge e = x\} & \equiv \\ \{e > 0 \wedge y \geq 0 \wedge e = x \wedge y = y\} & \equiv \\ p := y; & \\ \{e > 0 \wedge p \geq 0 \wedge e = x \wedge p = y\} & \equiv \\ \{e > 0 \wedge p \geq 0 \wedge 1 = 1 \wedge e = x \wedge p = y\} & \\ z := 1; & \\ \{e > 0 \wedge p \geq 0 \wedge z = 1 \wedge e = x \wedge p = y\} & \end{array}$$

2. On pose l'invariant: $\{e^p * z = x^y \wedge p \geq 0\}$ et on prouve avec la règle du while

$$\{e > 0 \wedge p \geq 0 \wedge z = 1 \wedge e = x \wedge p = y\}\text{while } p > 0 \text{ do B1 done } \{z = x^y\}$$

où B1 est le bloc

```
if p mod 2 <> 0 then p := p-1 ; z := z * e ;
e := e * e ;
p := p div 2 ;
```

pour cela on doit vérifier:

(a) $\{e > 0 \wedge p \geq 0 \wedge z = 1 \wedge e = x \wedge p = y\} \rightarrow \{e^p * z = x^y \wedge p \geq 0\}$.
Ceci est facile :-)

(b) $\{e^p * z = x^y \wedge p \geq 0 \wedge p > 0\} B1 \{e^p * z = x^y \wedge p \geq 0\}$.

- La première étape consiste à montrer avec l'axiome de substitution et la règle de composition :

$$\{(e * e)^{p\ div\ 2} * z = x^y \wedge p\ div\ 2 \geq 0\}$$
$$e := e * e;$$
$$p := p\ div\ 2;$$
$$\{e^p * z = x^y \wedge p \geq 0\}$$

- La deuxième étape consiste à montrer avec la deuxième règle conditionnelle :

$$\{e^p * z = x^y \wedge p \geq 0 \wedge p > 0\}$$
$$\text{if } p\ mod\ 2 <> 0 \text{ then } p := p - 1; z := z * e;$$
$$\{(e * e)^{p\ div\ 2} * z = x^y \wedge p\ div\ 2 \geq 0\}$$

  – Cas pre-condition et prédicat vrai du if:

$$\{e^p * z = x^y \wedge p \geq 0 \wedge p > 0 \wedge p\ mod\ 2 <> 0\}$$
$$p := p - 1; z := z * e;$$
$$\{(e * e)^{p\ div\ 2} * z = x^y \wedge p\ div\ 2 \geq 0\}$$

On raisonne comme suit (du bas vers le haut)

$\{e^p * z = x^y \wedge p \geq 0 \wedge p > 0 \wedge p\ mod\ 2 <> 0\} implique$
$\{e^p * z = x^y \wedge p > 0 \wedge p\ mod\ 2 = 1\} ssi$
$\{e^p * z = x^y \wedge p > 0 \wedge (p-1)\ mod\ 2 = 0\} ssi$
$\{e^{(p-1)} * e * z = x^y \wedge p > 0 \wedge (p-1)\ mod\ 2 = 0\} implique$
$\{e^{(p-1)-((p-1)\ mod\ 2)} * (z * e) = x^y \wedge (p-1)\ div\ 2 \geq 0\}$ par la règle subst deux fois
$p := p - 1; z := z * e;$
$\{e^{(p-(p\ mod\ 2))} * z = x^y \wedge p\ div\ 2 \geq 0\} ssi$
$\{e^{(2*(p\ div\ 2))} * z = x^y \wedge p\ div\ 2 \geq 0\} ssi$
$\{(e * e)^{p\ div\ 2} * z = x^y \wedge p\ div\ 2 \geq 0\}$

On peut donc conclure par la règle de conséquence.

  – Cas pre-condition et prédicat faux du if: $\{e^p * z = x^y \wedge p \geq 0 \wedge p > 0 \wedge p\ mod\ 2 = 0\} \rightarrow \{(e * e)^{p\ div\ 2} * z = x^y \wedge p\ div\ 2 \geq 0\}$:
  Ceci est facile :-)

(c) Maintenant, il suffit d'appliquer la règle de composition pour completer la preuve de cette partie.

(d) $\{e^p * z = x^y \wedge p \geq 0 \wedge p \leq 0\} \rightarrow \{z = x^y\}$. De $p \geq 0 \wedge p \leq 0$ on en déduit $p = 0$, et donc $z = e^0 * z = x^y$.

3. A partir de 1 et 2 on peut conclure (avec la règle séquentielle)
$\{x > 0 \wedge y \geq 0\}$

```
e := x ; p := y ; z := 1 ;
while p > 0 do
        if p mod 2 <> 0 then p := p-1; z := z * e ;
        e := e * e ;
        p := p div 2 ;
done;
```

$\{z = x^y\}$