CS412 Exercise sheet 3 - solutions

Functions

1. If $f_1 \in X \rightarrowtail\!\!\!\!\!\!+ Y, f_2 \in X \rightarrowtail\!\!\!\!\!\!+ Y, f_3 \in Y \rightarrowtail\!\!\!\!\!\!+ Z$ which of the following are necessarily functions? Give a brief justification or a counterexample as appropriate.

   (a) $f_1 \cup f_2$ not necessarily a function. Eg: $f_1 = \{x \mapsto y_1\}$, $f_2 = \{x \mapsto y_2\}$

   (b) $f_1 \cap f_2$ has a subset of elements of f_1 (and of f_2) and therefore functional property is preserved.

   (c) $f_1^{\sim}$ not necessarily a function. Eg: $f_1 = \{x_1 \mapsto y, x_2 \mapsto y\}$

   (d) $f_1 \mathbin{;} f_3$ is functional - there can be no diverging arrows at either stage. Can prove this by considering the definitions. We know that:

   $$\forall x, y_1, y_2 \bullet ((x : X \wedge y_1 : Y \wedge y_2 : Y \wedge f_1(x) = y_1 \wedge f_1(x) = y_2) \Rightarrow y_1 = y_2) \qquad \mathbf{1}$$
   $$\forall y, z_1, z_2 \bullet ((y : Y \wedge z_1 : Z \wedge z_2 : Z \wedge f_2(y) = z_1 \wedge f_2(y) = z_2) \Rightarrow z_1 = z_2) \qquad \mathbf{2}$$

   Consider $x : X;\ z_1, z_2 : Z$. If $(f_1 \mathbin{;} f_3)(x) = z_1$ and $(f_1 \mathbin{;} f_3)(x) = z_2$ then there exist $y_1, y_2 : Y$ such that:

   $$f_1(x) = y_1 \quad \text{and} \quad f_3(y_1) = z_1$$
   $$f_1(x) = y_2 \quad \text{and} \quad f_3(y_2) = z_2$$

   By **1**, $y_1 = y_2$, and therefore by **2**, $z_1 = z_2$. Therefore, $f_1 \mathbin{;} f_3$ is functional.

   (e) $f_1 \mathbin{;} (f_2^{-1})$ not necessarily a function.
   Eg: $f_1 = \{x_1 \mapsto y_1\}$, $f_2 = \{x_1 \mapsto y_1, x_2 \mapsto y_1\}$

2. (a) $\{0 \mapsto 0, 1 \mapsto 1, 2 \mapsto 4, 3 \mapsto 9, 4 \mapsto 16\}$

   (b) $\{x, y \mid x \in \mathbb{N}_1 \wedge y \in \mathbb{N} \wedge y = x - 1\}$

   (c) $\{x, y \mid x \in \mathbb{N} \wedge y \in \mathbb{N} \wedge \exists\, n \bullet (n \in \mathbb{N} \wedge x = 2 * n) \wedge y = 0 \,..\, x\}$

3. (a) $reg \subseteq IDENTIFIER \wedge sname \in IDENTIFIER \rightarrowtail\!\!\!\!\!\!+ NAME \wedge sdeg \in IDENTIFIER \rightarrowtail\!\!\!\!\!\!+ DEGREE \wedge dom(sname) = reg \wedge dom(sdeg) = reg$ It's not necessaary to have the reg component - could just use the domain of $sname$ say.

   (b) $reg, sname, sdeg := \{\}\,\{\}, \{\}$

   (c) $nn \leftarrow getname(ss) \mathrel{\widehat{=}}$
   $PRE\ ss : IDENTIFIER \wedge ss \in reg$
   $THEN\ nn := sname(ss)$
   END

   (d) $dd \leftarrow notakers \mathrel{\widehat{=}} dd := DEGREE - ran(sdeg)$

(e) $ii \leftarrow addstudent(nn, dd) \mathrel{\widehat{=}}$

$$
\begin{array}{l}
\quad PRE\ nn : NAME \wedge dd : DEGREE \\
\quad THEN \\
\quad\quad ANY\ xx\ WHERE\ xx \in IDENTIFIER - reg \\
\quad\quad THEN\ reg := reg \cup \{xx\} \\
\quad\quad\quad\quad sname := sname \cup \{xx \mapsto nn\} \\
\quad\quad\quad\quad sdeg := sdeg \cup \{xx \mapsto dd\} \\
\quad\quad\quad\quad ii := xx \\
\quad\quad END \\
\quad END
\end{array}
$$

(f) $nn \leftarrow shownames(dd) \mathrel{\widehat{=}}$

$$
\begin{array}{l}
\quad PRE\ dd : DEGREE \\
\quad THEN\ nn := sname[sdeg^{-1}[\{dd\}]] \\
\quad END
\end{array}
$$

4. Requirements are a bit vague, so you may come up with a variety of things. Here's one approach. As a lorry should only be loaded once for a delivery round, this uses a flag to record whether or not loading has happened. When it comes to working out what goods are to be loaded onto a lorry it's useful to have a function which can take a delivery and sum the total number for each element of $GOODS$. We can do this by declaring a constant (here called *sumgoods*) and giving its definition in the PROPERTIES section. (A more natural way to do this is by using a DEFINITIONS section.)

MACHINE *deliveries*

SETS $GOODS;\ ADDRESS;\ LORRY;\ FLAG = \{yes, no\}$

CONSTANTS $Order, sumgoods$

PROPERTIES

$$\begin{array}{ll}
Order = GOODS \nrightarrow \mathbb{N}_1 \wedge & \\
sumgoods = & \{dd, tt \mid \\
& \quad dd \in ADDRESS \nrightarrow Order \wedge tt \in GOODS \nrightarrow \mathbb{N} \wedge \\
& \quad \mathrm{dom}(tt) = \{xx \mid xx \in GOODS \wedge \exists\, yy \bullet (yy \in ADDRESS \wedge xx \in \mathrm{dom}(dd(yy)))\} \wedge \\
& \quad \forall\, gg \bullet (gg \in \mathrm{dom}(tt) \Rightarrow \\
& \qquad\qquad tt(gg) = \sum aa \bullet (aa \in \mathrm{dom}\, dd \wedge gg \in \mathrm{dom}(dd(aa)) \mid dd(aa)(gg))) \\
& \}
\end{array}$$

VARIABLES $todeliver, loadnow$

INVARIANT $todeliver \in LORRY \rightarrow (ADDRESS \nrightarrow Order) \wedge$
$loadnow \in LORRY \rightarrow FLAG$

INITIALISATION
$todeliver := LORRY \times \{\{\}\} \ \ || \ \ loadnow := LORRY \times \{yes\}$

OPERATIONS

$next\_del(ll, dd) \ \widehat{=}$
PRE $ll : LORRY \wedge dd : ADDRESS \nrightarrow Order \wedge todeliver(ll) = \{\}$
THEN $todeliver(ll) := dd \ \ || \ \ loadnow(ll) := yes$
END;

$oo \leftarrow to\_load(ll) \ \widehat{=}$
PRE $ll : LORRY \wedge loadnow(ll) = yes$
THEN $oo := sumgoods(todeliver(ll)) \ \ || \ \ loadnow(ll) := no$
END;

$delivery(ll, aa) \ \widehat{=}$
PRE $ll : LORRY \wedge aa : ADDRESS$
THEN $todeliver(ll) := \{aa\} \lhd todeliver(ll)$
END;

END