Propositional logic

Solution 1

(a) $\mathit{false}(\mathit{as}(\mathit{true} \Rightarrow \mathit{false}) \Leftrightarrow \mathit{false})$

(b) $\mathit{true}(\mathit{as}(\mathit{false} \Rightarrow \mathit{false}) \Leftrightarrow \mathit{true})$

(c) $\mathit{true}(\mathit{as}(\mathit{false} \Rightarrow \mathit{true}) \Leftrightarrow \mathit{true})$

(d) $\mathit{true}(\mathit{as}(\mathit{false} \Rightarrow \mathit{false}) \Leftrightarrow \mathit{true})$

(Assuming that pigs can't fly . . .)

Solution 2

(a)

p	q	$p \wedge q$	$(\mathbf{p} \wedge \mathbf{q}) \Rightarrow \mathbf{p}$
t	t	t	**t**
t	f	f	**t**
f	t	f	**t**
f	f	f	**t**

(b)

p	q	$p \wedge q$	$\neg\, p$	$\neg\, p \Rightarrow (p \wedge q)$	$(\neg\, \mathbf{p} \Rightarrow (\mathbf{p} \wedge \mathbf{q})) \Leftrightarrow \mathbf{p}$
t	t	t	f	t	**t**
t	f	f	f	t	**t**
f	t	f	t	f	**t**
f	f	f	t	f	**t**

(c)

p	q	$p \Rightarrow q$	$p \wedge (p \Rightarrow q)$	$(\mathbf{p} \wedge (\mathbf{p} \Rightarrow \mathbf{q})) \Rightarrow \mathbf{q}$
t	t	t	t	**t**
t	f	f	f	**t**
f	t	t	f	**t**
f	f	t	f	**t**

Solution 3

(a)

$$
\begin{array}{ll}
p \Rightarrow \neg\, p & \\
\Leftrightarrow \neg\, p \vee \neg\, p & [\Rightarrow] \\
\Leftrightarrow \neg\, p & [\text{idempotence}]
\end{array}
$$

(b)

$$
\begin{array}{ll}
\neg\, p \Rightarrow p & \\
\Leftrightarrow \neg\, \neg\, p \vee p & [\Rightarrow] \\
\Leftrightarrow p \vee p & [\neg\ \neg\,] \\
\Leftrightarrow p & [\text{idempotence}]
\end{array}
$$

(c)

$$
\begin{aligned}
& p \Rightarrow (q \Rightarrow r) \\
& \Leftrightarrow \neg\, p \vee (q \Rightarrow r) && [\Rightarrow] \\
& \Leftrightarrow \neg\, p \vee \neg\, q \vee r && [\Rightarrow] \\
& \Leftrightarrow \neg\, p \vee \neg\, q \vee r && [\text{associativity}] \\
& \Leftrightarrow \neg\, (p \wedge q) \vee r && [\text{De Morgan}] \\
& \Leftrightarrow p \wedge q \Rightarrow r && [\Rightarrow]
\end{aligned}
$$

(d)

$$
\begin{aligned}
& q \Rightarrow (p \Rightarrow r) \\
& \Leftrightarrow \neg\, q \vee (p \Rightarrow r) && [\Rightarrow] \\
& \Leftrightarrow \neg\, q \vee \neg\, p \vee r && [\Rightarrow] \\
& \Leftrightarrow \neg\, p \vee \neg\, q \vee r && [\text{associativity} \wedge \text{commutativity}] \\
& \Leftrightarrow \neg\, p \vee (q \Rightarrow r) && [\Rightarrow] \\
& \Leftrightarrow p \Rightarrow (q \Rightarrow r) && [\Rightarrow]
\end{aligned}
$$

(e)

$$
\begin{aligned}
& p \wedge q \Leftrightarrow p \\
& \Leftrightarrow (p \wedge q \Rightarrow p) \wedge (p \Rightarrow p \wedge q) && [\Leftrightarrow] \\
& \Leftrightarrow (\neg\, (p \wedge q) \vee p) \wedge (\neg\, p \vee p \wedge q) && [\Rightarrow] \\
& \Leftrightarrow (\neg\, p \vee \neg\, q \vee p) \wedge (\neg\, p \vee p \wedge q) && [\text{De Morgan}] \\
& \Leftrightarrow (\neg\, q \vee \neg\, p \vee p) \wedge (\neg\, p \vee p \wedge q) && [\text{associativity} \wedge \text{comm .}] \\
& \Leftrightarrow (\neg\, q \vee \mathit{true}) \wedge (\neg\, p \vee p \wedge q) && [\text{excluded middle}] \\
& \Leftrightarrow \mathit{true} \wedge (\neg\, p \vee p \wedge q) && [\vee \wedge \text{true}] \\
& \Leftrightarrow \neg\, p \vee p \wedge q && [\wedge \wedge \text{true}] \\
& \Leftrightarrow (\neg\, p \vee p) \wedge (\neg\, p \vee q) && [\text{distribution}] \\
& \Leftrightarrow \mathit{true} \wedge (\neg\, p \vee q) && [\text{excluded middle}] \\
& \Leftrightarrow \neg\, p \vee q && [\wedge \wedge \text{true}] \\
& \Leftrightarrow p \Rightarrow q && [\Rightarrow]
\end{aligned}
$$

(f)

$$
\begin{aligned}
& p \vee q \Leftrightarrow p \\
& \Leftrightarrow (p \vee q \Rightarrow p) \wedge (p \Rightarrow p \vee q) && [\Leftrightarrow] \\
& \Leftrightarrow (\neg\, (p \vee q) \vee p) \wedge (\neg\, p \vee p \vee q) && [\Rightarrow] \\
& \Leftrightarrow (\neg\, p \wedge \neg\, q \vee p) \wedge (\neg\, p \vee p \vee q) && [\text{De Morgan}] \\
& \Leftrightarrow (\neg\, p \vee p) \wedge (\neg\, q \vee p) \wedge (\neg\, p \vee p \vee q) && [\text{distribution}] \\
& \Leftrightarrow \mathit{true} \wedge (\neg\, q \vee p) \wedge (\neg\, p \vee p \vee q) && [\text{excluded middle}] \\
& \Leftrightarrow (\neg\, q \vee p) \wedge (\neg\, p \vee p \vee q) && [\wedge \wedge \text{true}] \\
& \Leftrightarrow (\neg\, q \vee p) \wedge (\neg\, p \vee p \vee q) && [\text{associativity}] \\
& \Leftrightarrow (\neg\, q \vee p) \wedge (\mathit{true} \vee q) && [\text{excluded middle}] \\
& \Leftrightarrow (\neg\, q \vee p) \wedge \mathit{true} && [\vee \wedge \text{true}] \\
& \Leftrightarrow \neg\, q \vee p && [\wedge \wedge \text{true}] \\
& \Leftrightarrow q \Rightarrow p && [\Rightarrow]
\end{aligned}
$$

Solution 4

(a) (p or q) $\Leftrightarrow$ ((not p or not q) and q) is not a tautology. You might illustrate this via a truth table or via a chain of equivalences, showing that the proposition is not equivalent to true. Alternatively, you might try and find a combination of values for which the proposition is false. (In this case, the proposition is false when p and q are both true.)

(b) (p or q) $\Leftrightarrow$ ((not p and not q) or q) is not a tautology. In this case, the proposition is false when p is true and q is false.

Solution 5

(a) $\exists\, d : \mathit{Dog} \bullet \mathit{gentle}(d) \wedge \mathit{well\_trained}(d)$

(b) $\forall\, d : \mathit{Dog} \bullet \mathit{neat}(d) \wedge \mathit{well\_trained}(d) \Rightarrow \mathit{attractive}(d)$

(c) $\exists\, d : \mathit{Dog} \bullet \mathit{gentle}(d) \Rightarrow \forall\, t : \mathit{Trainer} \bullet \mathit{groomed}(d, t)$

Solution 6

(a) This is a true proposition: whatever the value of x, the expression x^2 - x + 1 denotes a natural number. If we choose y to be this natural number, we will find that p is true.

(b) This is a false proposition. We cannot choose a large enough value for y such that p will hold for any value of x.

(c) This is a false proposition. It is an implication whose antecedent part is true and whose consequent part is false.

(d) This is a true proposition. It is an implication whose antecedent part is false and whose consequent part is true.

Solution 7

(a) We must define a predicate p that is false for at least one value of x, and is true for at least one other value. A suitable solution would be $p \Leftrightarrow x > 1$.

(b) With the above choice of p, we require only that q is sometimes false when p is true (for else the universal quantification would hold). A suitable solution would be $q \Leftrightarrow x > 3$.

Solution 8

(a) $\forall x : \mathbb{N} \bullet x \geq z$

(b) $\forall z : \mathbb{N} \bullet z \geq x + y$

(c) $x + 3 > 0 \land \forall z : \mathbb{N} \bullet z \geq x + 3$

Equality

Solution 9

(a)

$$
\begin{aligned}
&\exists y : \mathbb{N} \bullet y \in \{0, 1\} \land y \neq 1 \land x \neq y \\
&\Leftrightarrow \exists y : \mathbb{N} \bullet y = 0 \land x \neq y && \text{[arithmetic]} \\
&\Leftrightarrow 0 \in \mathbb{N} \land x \neq 0 && \text{[one - point rule]} \\
&\Leftrightarrow x \neq 0
\end{aligned}
$$

(b)

$$
\begin{aligned}
&\exists x, y : \mathbb{N} \bullet x + y = 4 \land x < y \\
&\Leftrightarrow \exists x, y : \mathbb{N} \bullet y = 4 - x \land x < y \\
&\Leftrightarrow \exists x : \mathbb{N} \bullet 4 - x \in \mathbb{N} \land x < 4 - x \\
&\Leftrightarrow \mathit{true}
\end{aligned}
$$

The final equivalence holds because $0 \in N$, $4 - 0 \in N$, and $0 < 4$.

(c)

$$
\begin{aligned}
&\forall x : \mathbb{N} \bullet \exists y : \mathbb{N} \bullet x = y + 1 \\
&\Leftrightarrow \forall x : \mathbb{N} \bullet \exists y : \mathbb{N} \bullet y = x - 1 \\
&\Leftrightarrow \forall x : \mathbb{N} \bullet x - 1 \in \mathbb{N}
\end{aligned}
$$

The final equivalence holds because $0 \in N$ and yet $0 - 1 \notin N$. We may assume that the subtraction operator is defined for all integers.

(d)

$\exists x: \mathbb{N} \bullet x = 1 \land x > y \lor x = 2 \land x > z$
$\Leftrightarrow \exists x: \mathbb{N} \bullet x = 1 \land x > y \lor \exists x: \mathbb{N} \bullet x = 2 \land x > z$
$\Leftrightarrow 1 \in \mathbb{N} \land 1 > y \lor \exists x: \mathbb{N} \bullet x = 2 \land x > z$
$\Leftrightarrow 1 \in \mathbb{N} \land 1 > y \lor 2 \in \mathbb{N} \land 2 > z$
$\Leftrightarrow 1 > y \lor 2 > z$

Solution 10

As discussed, the quantifier $\exists_1$ can help give rise to a 'test' or 'precondition' to ensure that an application of mu will work.

So, as a simple example, as the proposition

$\exists_1 n: \mathbb{N} \bullet \forall m: \mathbb{N} \bullet n \leq m$

is equivalent to true, we can be certain that the statement

$\mu n: \mathbb{N} \bullet \forall m: \mathbb{N} \bullet n \leq m$

will return a result (which happens to be 0).

Solution 11

(a) $\mu a: \mathbb{N} \bullet a = a = 0$

is a provable statement, since 0 is the only natural number with the specified property.

(b) $\mu b: \mathbb{N} \bullet b = b = 1$

is not provable. The specified property is true of both 0 and 1, and thus the value of the mu-expression is undefined.

(c) $\mu c: \mathbb{N} \bullet c > c = \mu c: \mathbb{N} \bullet c > c$

is a provable statement. Neither expression is properly defined, but we may conclude that they are equal; there is little else that we can prove about them.

(d) $\mu d: \mathbb{N} \bullet d = d = 1$

is not a provable statement. We cannot confirm that 1 is the only natural number with the specified property; we do not know what value is taken by undefined operations.

Solution 12

(Requires mu-operator with expression part - not yet implemented)

(a) $\mu m: \mathit{Mountain} \mid \forall n: \mathit{Mountain} \bullet \mathit{height}(n) \leq \mathit{height}(m) \bullet \mathit{height}(m)$

(b) $\mu c: \mathit{Chapter} \mid \exists_1 d: \mathit{Chapter} \bullet \mathit{length}(d) > \mathit{length}(c) \bullet \mathit{length}(c)$

(c) Assuming the existence of a suitable function, max: ($\mu n: \mathbb{N} \bullet n = \mathit{max}(\{m: \mathbb{N} \mid 8 * m < 100.8 * m\})$. 100 - n)

Deductive proofs

Solution 13

$$\dfrac{\ulcorner p \wedge (p \Rightarrow q) \urcorner^{[1]} \qquad \dfrac{\dfrac{p \wedge (p \Rightarrow q)}{p}\,[\wedge\text{-elim}^{[1]}] \qquad \dfrac{\dfrac{p \wedge (p \Rightarrow q)}{p \Rightarrow q}\,[\wedge\text{-elim}^{[1]}] \quad \dfrac{p \wedge (p \Rightarrow q)}{p}\,[\wedge\text{-elim}^{[1]}]}{q}\,[\Rightarrow \text{elim}]}{p \wedge q}\,[\wedge \text{ intro}]}{p \wedge (p \Rightarrow q) \Rightarrow p \wedge q}\,[\Rightarrow\text{-intro}^{[1]}]$$

Solution 14

In one direction:

$$\dfrac{\ulcorner p \wedge q \Leftrightarrow p \urcorner^{[1]} \qquad \dfrac{\ulcorner p \urcorner^{[2]} \qquad \dfrac{\dfrac{\dfrac{}{p \wedge q}\,[\text{derived}]}{p \wedge q}\,[\Rightarrow \text{elim from } 1 \wedge 2]}{q}\,[\wedge\text{-elim}^{[3]}]}{p \Rightarrow q}\,[\Rightarrow\text{-intro}^{[2]}]}{(p \wedge q \Leftrightarrow p) \Rightarrow (p \Rightarrow q)}\,[\Rightarrow\text{-intro}^{[1]}]$$

and the other:

$$\dfrac{\ulcorner p \Rightarrow q \urcorner^{[1]} \qquad \dfrac{\dfrac{\ulcorner p \wedge q \urcorner^{[2]} \quad \ulcorner p \urcorner^{[2]}}{p \wedge q \Rightarrow p}\,[\Rightarrow\text{-intro}^{[2]}] \qquad \dfrac{\ulcorner p \urcorner^{[3]} \quad \ulcorner p \wedge q \urcorner^{[1]}}{p \Rightarrow p \wedge q}\,[\Rightarrow\text{-intro}^{[3]}]}{p \wedge q \Leftrightarrow p}\,[\Leftrightarrow \text{ intro}]}{(p \Rightarrow q) \Rightarrow (p \wedge q \Leftrightarrow p)}\,[\Rightarrow\text{-intro}^{[1]}]$$

We can then combine these two proofs *with* $\Leftrightarrow$ *intro*.

Solution 15

$$\dfrac{\ulcorner (p \Rightarrow q) \wedge \neg\, q \urcorner^{[1]} \qquad \dfrac{\ulcorner p \urcorner^{[2]} \qquad \dfrac{\dfrac{\ulcorner p \Rightarrow q \urcorner^{[1]} \quad \ulcorner p \urcorner^{[2]}}{q}\,[\Rightarrow \text{elim}] \qquad \ulcorner \neg\, q \urcorner^{[1]}}{\mathit{false}}\,[\text{false intro}]}{\neg\, p}\,[\text{false-elim}^{[2]}]}{(p \Rightarrow q) \wedge \neg\, q \Rightarrow \neg\, p}\,[\Rightarrow\text{-intro}^{[1]}]$$

Solution 16

In one direction:

$$\dfrac{\ulcorner p \wedge (q \vee r)\urcorner^{[1]} \qquad \dfrac{\ulcorner q \vee r\urcorner^{[1]} \qquad \dfrac{\dfrac{\ulcorner p\urcorner^{[1]} \quad \dfrac{}{q}\,[\text{case assumption}]}{p \wedge q}\,[\wedge\text{ intro}]}{p \wedge q \vee p \wedge r}\,[\vee\text{ intro}] \qquad \dfrac{\dfrac{\ulcorner p\urcorner^{[1]} \quad \dfrac{}{r}\,[\text{case assumption}]}{p \wedge r}\,[\wedge\text{ intro}]}{p \wedge q \vee p \wedge r}\,[\vee\text{ intro}]}{p \wedge q \vee p \wedge r}\,[\vee\text{-elim}^{[2]}]}{p \wedge (q \vee r) \Rightarrow p \wedge q \vee p \wedge r}\,[\Rightarrow\text{-intro}^{[1]}]$$

In the other:

$$\dfrac{\ulcorner p \wedge q \vee p \wedge r\urcorner^{[3]} \qquad \dfrac{\ulcorner \mathit{case}1 \vee \mathit{case}2\urcorner^{[3]} \qquad \dfrac{\dfrac{\dfrac{}{p}\,[\wedge\text{ elim}]}{q \vee r}\,[\vee\text{ intro}]}{p \wedge (q \vee r)}\,[\wedge\text{ intro}] \qquad \dfrac{\dfrac{\dfrac{}{p}\,[\wedge\text{ elim}]}{q \vee r}\,[\vee\text{ intro}]}{p \wedge (q \vee r)}\,[\wedge\text{ intro}]}{p \wedge (q \vee r)}\,[\vee\text{-elim}^{[4]}]}{p \wedge q \vee p \wedge r \Rightarrow p \wedge (q \vee r)}\,[\Rightarrow\text{-intro}^{[3]}]$$

Solution 17

In one direction:

$$\dfrac{\ulcorner p \vee q \wedge r\urcorner^{[3]} \qquad \dfrac{}{(p \vee q) \wedge (p \vee r)}\,[\vee\text{ elim } \wedge \wedge\text{ intro}]}{p \vee q \wedge r \Rightarrow (p \vee q) \wedge (p \vee r)}\,[\Rightarrow\text{-intro}^{[3]}]$$

and the other:

$$\dfrac{\ulcorner (p \vee q) \wedge (p \vee r)\urcorner^{[1]} \qquad \ulcorner p \vee q \wedge r\urcorner^{[2]}}{(p \vee q) \wedge (p \vee r) \Rightarrow p \vee q \wedge r}\,[\Rightarrow\text{-intro}^{[1]}]$$

Solution 18

In one direction:

$$\frac{\ulcorner p \Rightarrow q \urcorner^{[1]} \quad \neg\, p \vee q}{(p \Rightarrow q) \Rightarrow \neg\, p \vee q}\ [\Rightarrow\text{-intro}^{[1]}]$$

and the other:

$$\frac{\ulcorner \neg\, p \vee q \urcorner^{[3]} \quad \dfrac{\ulcorner p \urcorner^{[4]} \quad \ulcorner q \urcorner^{[3]}}{p \Rightarrow q}\ [\Rightarrow\text{-intro}^{[4]}]}{\neg\, p \vee q \Rightarrow (p \Rightarrow q)}\ [\Rightarrow\text{-intro}^{[3]}]$$

Sets and types

Solution 19

(a) 1 in $\{4, 3, 2, 1\}$ is true.

(b) $\{1\}$ in $\{1, 2, 3, 4\}$ is undefined.

(c) $\{1\}$ in $\{\{1\}, \{2\}, \{3\}, \{4\}\}$ is true.

(d) The empty set in $\{1, 2, 3, 4\}$ is undefined.

Solution 20

(a) $\{1\} \times \{2, 3\}$

is the set $\{(1, 2), (1, 3)\}$

(b) The empty set cross $\{2, 3\}$ is the empty set

(c) $\mathbb{P}\, \emptyset \times \{1\}$

is the set $\{(\emptyset, 1)\}$

(d) $\{(1, 2)\}$ cross $\{3, 4\}$ is the set $\{((1, 2), 3), ((1, 2), 4)\}$

Solution 21

There are various ways of describing these sets via set comprehensions. Examples are given below.

(a) $\{z : \mathbb{Z} \mid 0 \leq z \wedge z \leq 100\}$

(b) $\{z : \mathbb{Z} \mid z = 10\}$

(c) $\{z : \mathbb{Z} \mid z \bmod 2 = 0 \vee z \bmod 3 = 0 \vee z \bmod 5 = 0\}$

Solution 22

(a) $\{n : \mathbb{N} \mid n \leq 4 \bullet n^2\}$

(b) $\{n : \mathbb{N} \mid n \leq 4 \bullet (n, n^2)\}$

(c) $n : P0, 1$

(d) $\{n : \mathbb{P}\{0,1\} \mid true \bullet (n, \#n)\}$

Solution 23

(a)

$$
\begin{aligned}
& x \in a \cap a \\
& \Leftrightarrow x \in a \wedge x \in a \\
& \Leftrightarrow x \in a
\end{aligned}
$$

(b)

$$
\begin{aligned}
& x \in a \cup a \\
& \Leftrightarrow x \in a \vee x \in a \\
& \Leftrightarrow x \in a
\end{aligned}
$$

Solution 24

(a)

The set of all pairs of integers is Z cross Z. To give it a name, we could write:

Pairs == Z $\times\mathbb{Z}$

(b)

The set of all integer pairs in which each element is strictly greater than zero could be defined by:

StrictlyPositivePairs == { m, n : $\mathbb{Z} \mid m > 0 \wedge n > 0 \bullet (m, n)\}$

(c)

It is intuitive to use a singular noun for the name of a basic type; we define the set of all people by writing:

$[Person]$

(d)

The set of all couples could be defined by:

Couples == { s : $\mathbb{P}$ *Person* | $\#s = 2$}

Solution 25

(Requires generic set notation and Cartesian product)

Solution 26

(Requires generic parameters and relation type notation)

Relations

Solution 27

(a)

The power set of $\{(0,0),(0,1),(1,0),(1,1)\}$ is:

$\{\emptyset, \{(0,0)\}, \{(0,1)\}, \{(1,0)\}, \{(1,1)\}, \{(1,0),(1,1)\}, \{(0,0),(0,1)\}, \{(0,1),(1,1)\}, \{(0,1),(1,0)\}, \{(0,0),(1,1)\}, \{(0,0)$

(b) $\{\emptyset, \{(0,0)\}, \{(0,1)\}, \{(0,0),(0,1)\}\}$

(c) $\{\emptyset\}$

(d) $\{\emptyset\}$

Solution 28

(a) $\operatorname{dom} R = \{0,1,2\}$

(b) $\operatorname{ran} R = \{1,2,3\}$

(c) $\{1,2\} \lhd R = \{1 \mapsto 2, 1 \mapsto 3, 2 \mapsto 3\}$

Solution 29

(a) $\{2 \mapsto 4, 3 \mapsto 3, 3 \mapsto 4, 4 \mapsto 2\}$

(b) $\{1 \mapsto 3, 2 \mapsto 2, 2 \mapsto 3, 3 \mapsto 1\}$

(c) $\{1 \mapsto 1, 2 \mapsto 2, 2 \mapsto 3, 3 \mapsto 2, 3 \mapsto 3, 4 \mapsto 4\}$

(d) $\{1 \mapsto 4, 2 \mapsto 2, 2 \mapsto 3, 3 \mapsto 2, 3 \mapsto 3, 4 \mapsto 1\}$

Solution 30

| *childOf* : *Person* $\leftrightarrow$ *Person*

(a)

parentOf == childOf^{-1}

This is a good example of how there are many different ways of writing the same thing. An alternative abbreviation is:

parentOf == { x, y : $Person \mid x \mapsto y \in childOf \bullet y \mapsto x\}$

Or, via an axiomatic definition:

$$\begin{array}{|l} parentOf : Person \leftrightarrow Person \\ \hline parentOf = childOf^{-1} \end{array}$$

(b)

siblingOf == (childOf $\circ parentOf) \setminus id$

(c)

cousinOf == childOf $\circ siblingOf \circ parentOf$

(d)

ancestorOf == parentOf$^+$

Solution 31

(Requires compound identifiers with operators - R+, R*)

(a)

R == { a, b : $\mathbb{N} \mid b = a \vee b = a\}$

(b)

S == { a, b : $\mathbb{N} \mid b = a \vee b = a\}$

(c) R+ == $\{a, b \colon \mathbb{N} \mid b > a\}$

(d) R* == $\{a, b \colon \mathbb{N} \mid b \geq a\}$

Solution 32

(a)

$$\begin{array}{l} x \mapsto y \in A \lhd B \lhd R \\ \Leftrightarrow x \in A \wedge x \mapsto y \in (B \lhd R) \\ \Leftrightarrow x \in A \wedge x \in B \wedge x \mapsto y \in R \\ \Leftrightarrow x \in A \cap B \wedge x \mapsto y \in R \\ \Leftrightarrow x \mapsto y \in A \cap B \lhd R \end{array}$$

(b)

$$
\begin{aligned}
&x \mapsto y \in R \cup S \rhd C \\
&\Leftrightarrow x \mapsto y \in R \cup S \wedge y \in C \\
&\Leftrightarrow (x \mapsto y \in R \vee x \mapsto y \in S) \wedge y \in C \\
&\Leftrightarrow x \mapsto y \in R \wedge y \in C \vee x \mapsto y \in S \wedge y \in C \\
&\Leftrightarrow x \mapsto y \in R \rhd C \vee x \mapsto y \in S \rhd C \\
&\Leftrightarrow x \mapsto y \in (R \rhd C) \cup (S \rhd C)
\end{aligned}
$$

Functions

Solution 33

The set of 9 functions:

$\{\emptyset, \{(0,0)\}, \{(0,1)\}, \{(1,1)\}, \{(1,0)\}, \{(0,0),(1,1)\}, \{(0,1),(1,1)\}, \{(1,0),(0,0)\}, \{(0,1),(1,0)\}\}$

(a)

The set of total functions:

$\{\{(0,0),(1,1)\}, \{(0,1),(1,1)\}, \{(1,0),(0,0)\}, \{(0,1),(1,0)\}\}$

(b)

The set of functions which are neither injective nor surjective:

$\{\{(0,1),(1,1)\}, \{(0,0),(1,0)\}\}$

(c)

The set of functions which are injective but not surjective:

$\{\emptyset, \{(0,0)\}, \{(0,1)\}, \{(1,0)\}, \{(1,1)\}\}$

(d) There are no functions (of this type) which are surjective but not injective.

(e)

The set of bijective functions:

$\{\{(0,0),(1,1)\}, \{(0,1),(1,0)\}\}$

Solution 34

(a) $\{1 \mapsto a, 2 \mapsto b, 3 \mapsto c, 4 \mapsto b\}$

(b) $\{1 \mapsto c, 2 \mapsto b, 3 \mapsto c, 4 \mapsto d\}$

(c) $\{1 \mapsto c, 2 \mapsto b, 3 \mapsto c, 4 \mapsto b\}$

(d) $\{1 \mapsto c, 2 \mapsto b, 3 \mapsto c, 4 \mapsto b\}$

Solution 35

(Requires power set notation P and relational image)

(a)

$$\begin{array}{|l} children : Person \rightarrow \mathbb{P}\ Person \\ \hline children = \{p\colon Person \bullet p \mapsto parentOf(\!\lvert\ \{p\}\ \rvert\!)\} \end{array}$$

(b)

$$\begin{array}{|l} number\_of\_grandchildren : Person \rightarrow \mathbb{N} \\ \hline number\_of\_grandchildren = \{p\colon Person \bullet p \mapsto \#parentOf \circ parentOf(\!\lvert\ \{p\}\ \rvert\!)\} \end{array}$$

Solution 36

(Note: This solution demonstrates relation types in quantifier domains)

$$\begin{array}{|l} number\_of\_drivers : Drivers \leftrightarrow Cars \rightarrow (Cars \rightarrow \mathbb{N}) \\ \hline number\_of\_drivers = \lambda\, r : Drivers \leftrightarrow Cars \bullet \{c\colon \operatorname{ran} r \bullet c \mapsto \#\{d\colon Drivers \mid d \mapsto c \in r\}\} \end{array}$$

Sequences

Solution 37

(a) $\langle a \rangle$

(b) $\{1 \mapsto a, 2 \mapsto b, 2 \mapsto a, 3 \mapsto c, 3 \mapsto b, 4 \mapsto d\}$

(c) $\{2 \mapsto b, 3 \mapsto c, 4 \mapsto d\}$

(d) $\{1, 2, 3, 4\}$

(e) $\{a, b\}$

(f) $\{a \mapsto 1, b \mapsto 2, c \mapsto 3, d \mapsto 4\}$

(g) $\langle a, b \rangle$

(h) $\{3 \mapsto b\}$

(i) $\{a\}$

(j) c

Solution 38

(a)

$$\begin{array}{|l}
f : Place \to \mathbb{P}\ Place \\
\hline
\forall\, p\colon Place \bullet f(p) = \{q\colon Place \mid p \mapsto q \in \operatorname{ran} trains\}
\end{array}$$

(b) $\{p\colon Place \mid \exists_1\, x\colon \operatorname{dom} trains \bullet trains(x).2 = p\}$

(c) $\mu\, p\colon Place \bullet \forall\, q\colon Place \bullet p \neq q \land \#\{x\colon \operatorname{dom} trains \mid trains(x).2 = p\} > \#\{x\colon \operatorname{dom} trains \mid trains(x).2 = q\}$

Solution 39

(a)

$large\_coins : Collection \to N$

$\forall\, c\colon Collection \bullet large\_coins(c) = c(large)$

(Blocked by: underscore in identifier for fuzz compatibility)

(b)

$add\_coin : Collection * Coin \to Collection$

$\forall\, c\colon Collection \bullet \forall\, d\colon Coin \bullet add\_coin(c, d) = c \cup [\![d]\!]$

(Blocked by: underscore in identifier and bag union)

Modelling

Solutions 40-52 are work in progress - many require features not yet implemented

Solution 40

(Work in progress - requires semicolon-separated bindings in set comprehensions)

(a)

$hd : seq(Title * Length * Viewed)$

$cumulative\_total(hd) \leq 12000$

$\forall\, p\colon \operatorname{ran} hd \bullet p.2 \leq 360$

Note that cumulative$_{t}otalisdefinedinpart(d)$.

(b) $\{p : \operatorname{ran} hd \mid p.2 > 120 \bullet p.1\}$

(c)

These can be defined recursively:

$$\begin{array}{|l} viewed : \text{seq } Programme \rightarrow \text{seq } Programme \\ \hline viewed(\langle\rangle) = \langle\rangle \land \forall x : Programme \bullet \forall s : \text{seq } Programme \bullet viewed(\langle x \rangle \frown s) = (\text{if } x.3 = yes \text{ then } \langle x \rangle \frown viewe \end{array}$$

or otherwise (omitted - requires semicolon-separated bindings in set comprehension)

(d)

$$\begin{array}{|l} cumulative\_total : \text{seq } Title * Length * Viewed \rightarrow \mathbb{N} \\ \hline cumulative\_total(\langle\rangle) = 0 \, \forall x : Title * Length * Viewed \bullet \forall s : \text{seq } Title * Length * Viewed \bullet cumulative\_total \end{array}$$

(e)

(mu $p : ranhd \mid \forall q : \operatorname{ran} hd \bullet p \neq q \land p.2 > q.2 \mid$ p.1)

(This, of course, assumes that there is a unique element with this property.)

(f)

(f) Omitted - requires semicolon-separated bindings in nested set comprehension

(g)

axdef

$g : seq(Title * Length * Viewed) \rightarrow seq(Title * Length * Viewed)$

where

$\forall s : \text{seq } Title * Length * Viewed \bullet g(s) = s \rhd \{x : \operatorname{ran} s \mid x \neq longest\_viewed(s)\}$

end

Where longest$_v iewed is defined as$

axdef

$longest\_viewed : seq(Title * Length * Viewed)+ \rightarrow Title * Length * Viewed$

where

$\forall s :$ seq $Title * Length * Viewed \bullet longest_viewed(s) = (\mu p : \operatorname{ran} s \bullet p.3 = yes$and $\forall q : \operatorname{ran} s \bullet p \neq q \wedge q.3 = yes \wedge p.2 > q.2)$

end

This, of course, assumes that there is at least one viewed programme (and one of a unique maximum length).

(h)

$s :$ seq $Title * Length * Viewed \rightarrow$ seq $Title * Length * Viewed$

$\forall x :$ seq $Title * Length * Viewed \bullet items(s(x)) = items(x) \wedge \forall i, j : \operatorname{dom} s(x) \bullet i < j \Rightarrow s(x)(i).2 \geq s(x)(j).2$

Solution 41

(a)

axdef

$records : Year \rightarrowtail\!\!\!\!\rightarrow$ Table

where

dom(records) = 1993..current

$\forall y : \operatorname{dom} records \bullet \#records(y) \leq 50$

$\forall\ y : dom(records)\ |\forall e : \operatorname{ran} records(y) \bullet year(e.1) = y$

$\forall\ r : ran(records)\ |\forall i1, i2 : \operatorname{dom} r \bullet i1 \neq i2 \wedge r(i1).1 = r(i2).1 \Rightarrow r(i1).3 \neq r(i2).3$

end

(b)

(i) $\{e : Entry \mid \exists r : \operatorname{ran} records \bullet e \in \operatorname{ran} r \land e.3 = 479\}$
ii
$\{e : Entry \mid \exists r : \operatorname{ran} records \bullet e \in \operatorname{ran} r \land e.6 > e.5\}$
iii
$\{e : Entry \mid \exists r : \operatorname{ran} records \bullet e \in \operatorname{ran} r \land e.7 \geq 70\}$
iv
$\{c : Course \mid \forall r : \operatorname{ran} records \bullet \forall e : \operatorname{ran} r \bullet e.2 = c \Rightarrow e.7 \geq 70\}$
v

$\{y : Year \mid y \in \operatorname{dom} records \bullet y \mapsto \{l : Lecturer \mid \#\{c : \operatorname{ran} records(y) \mid c.4 = l\} > 6\}\}$

(c)

axdef

where

$\forall x : Entry \bullet \forall s : \operatorname{seq} Entry \bullet 479\_courses(\langle\rangle) = \langle\rangle$ and $479_courses(\langle x \rangle\ ^s) = ifx.3 = 479then\langle x \rangle\ ^479_courses(s)else479_courses(s)$

end

(Blocked by: underscore in identifier - use camelCase for fuzz compatibility)

(d)

$\forall x : Entry \bullet \forall s : \operatorname{seq} Entry \bullet total(\langle\rangle) = 0 \land total(\langle x \rangle \frown s) = x.5 + total(s)$

Solution 42

$[Person]$

axdef

$State : P(seq(iseq(Person)))$

where

$\forall\ s : \mathit{State}\ |\forall\, i, j :\ \mathrm{dom}\, s \bullet i \neq j \wedge \mathrm{ran}\, s(i) \cap \mathrm{ran}\, s(j) = \{\}$

end

(b)

axdef

$\mathit{add} : N * \mathit{Person} * \mathit{State} \rightarrowtail\!\!\!\!\!\!\mapsto$ State

where

$\forall\, n : \mathbb{N} \bullet \forall\, p : \mathit{Person} \bullet \forall\, s : \mathit{State} \bullet n \in \mathrm{dom}\, s \wedge p \notin \bigcup \mathrm{ran}\,\mathrm{ran}\, s|$

add(n, p, s) = s ++ n $\mapsto$ s(n) $\langle p \rangle$

end

(Blocked by: $\rightarrowtail\!\!\!\!\!\!\mapsto$ operator not implemented)

Solution 43

(a)

(i) $\forall\ i : \mathit{dombookings}\ |\forall\, x, y :\ \mathit{bookings}(i) \bullet x \neq y \wedge x.2 \mathinner{..} x.3 \cap y.2 \mathinner{..} y.3 = \{\}$

(ii) $\forall\ i : \mathit{dombookings}\ |\forall$ x : $\mathit{bookings}(i)|\ \{x.2, x.3\}$ subseteq 1..max(i.1)

(iii) $\forall\ i : \mathit{dombookings}\ |\forall\, b :\ \mathit{bookings}(i) \bullet b.2 \leq b.3$

(iv) This is enforced by the constraint for part (i).

(b)

(i) $\{i :\ \mathrm{dom}\, \mathit{bookings} \mid i.1 = \mathit{Banbury} \bullet i.2\}$

(ii) $\{i\colon \operatorname{dom} bookings \mid i.1 = Banbury \land \exists\, b\colon bookings(i) \bullet 50 \in b.2 \mathrel{..} b.3\}$

(iii) $r : Room;\ s : N \mid \exists\, i\colon \operatorname{dom} bookings \bullet i.1 = r \land i.2 = s.$ (r, s)

(iv) $r : Room \mid \exists\, i\colon \operatorname{dom} bookings \bullet i.1 = r \land \# bookings(i) \geq 10$

Free types and induction

$[N]$

$Tree ::= stalk \mid leaf \langle\!\langle \mathbb{N} \rangle\!\rangle \mid branch \langle\!\langle Tree \times Tree \rangle\!\rangle$

Solution 44

The two cases of the proof are established by equational reasoning: the first by

reverse $(\langle\rangle\ ^{t}) = reverset[cat.1a] = (reverset)\langle\rangle$ [cat.1b]

where cat.1a is $\langle\rangle$ s = s$andcat.1biss\langle\rangle$ = s

and the second by

reverse $((\langle x\rangle\ ^{u})^{t}) = reverse(\langle x\rangle\ ^{(}u^{t}))[cat.2]$

= reverse (u ^{t})$\langle x\rangle$ [reverse.2]

= (reverse t $^{r}everseu)\langle x\rangle$ [anti-distributive]

= reverse t $^{(}reverseu\langle x\rangle)$ [cat.2]

= reverse t $^{r}everse(\langle x \rangle\ ^{u})[reverse.2]$

Solution 45

The base case:

reverse (reverse $\langle\rangle$) = reverse $\langle\rangle$ [reverse.1] = $\langle\rangle$ [reverse.1]

The inductive step:

reverse (reverse $(\langle x \rangle\ ^{t})$)

= reverse ((reverse t) $\langle x \rangle$) [reverse.2]

= reverse $(\langle x \rangle)\ ^{r}everse(reverset)[anti - distributive]$

= reverse $(\langle x \rangle\ \langle\rangle)\ ^{r}everse(reverset)[cat.1]$

= ((reverse $\langle\rangle$) $\langle x \rangle$) $^{r}everse(reverset)[reverse.2]$

= $(\langle\rangle\ \langle x \rangle)\ ^{r}everse(reverset)[reverse.1]$

= $\langle x \rangle\ ^{r}everse(reverset)[cat.1]$

= $\langle x \rangle\ ^{t}[reverse(reverset) = t]$

Solution 46

(a)

$count : Tree \rightarrow N$

count $stalk = 0$

$\forall\, n : \mathbb{N} \bullet count(leaf(n)) = 1$

$\forall\, t1, t2 \colon Tree \bullet count(branch(t1, t2)) = count(t1) + count(t2)$

(Blocked by : $recursivefreetypesandpatternmatching$)

(b)

$flatten : Tree \rightarrow seqN$

flatten stalk $= \langle\rangle$

$\forall\, n \colon \mathbb{N} \bullet flatten(leaf(n)) = \langle n \rangle$

$\forall\, t1, t2 \colon Tree \bullet flatten(branch(t1, t2)) = flatten(t1^{flatten})(t2)$

(Blocked by : $recursivefreetypesandpatternmatching$)

Solution 47

First, exhibit the induction principle for the free type:

P stalk and $(\forall\, n \colon \mathbb{N} \bullet \mathbb{P}\ leaf(n))$ and $(\forall\, t1, t2 \colon Tree \bullet \mathbb{P}\ t1 \wedge \mathbb{P}\ t2 \Rightarrow \mathbb{P}\ branch(t1, t2))$

implies $\forall\, t \colon Tree \bullet \mathbb{P}\ t$

This gives three cases for the proof:

(flatten stalk) $=$ $\langle\rangle$ [flatten] $= 0$ [] $=$ count stalk [count]

(Remaining cases omitted - require equational reasoning with recursive functions)

Supplementary material : assignment practice

Solution 48

$[SongId, UserId, PlaylistId, Playlist]$

$songs : \mathbb{F}\ SongIdusers : \mathbb{F}\ UserIdplaylists : PlaylistId \pfun PlaylistplaylistOwner : PlaylistId \pfun UserIdplaylistS$

$\forall\, i\colon\ \mathrm{dom}\, playlists \bullet \mathrm{ran}\, playlists(i)(subseteq)(songs)\, \mathrm{dom}\, playlistOwner(subseteq)(\mathrm{dom}\, playlists)\, \mathrm{ran}\, playlistO$

Solution 49

$hated : UserId \pfun \mathbb{F}\ SongIdloved : UserId \pfun \mathbb{F}\ SongId$

$\mathrm{dom}\, hated(subseteq)(users)\, \forall\, i\colon\ \mathrm{dom}\, hated \bullet hated(i)(subseteq)(songs)\, \mathrm{dom}\, loved(subseteq)(users)\, \forall\, i\colon\ \mathrm{dom}$

Solution 50

(a)

A == users $\setminus \bigcup \mathrm{ran}\, playlistSubscribers$

(b)

B == { p : $\mathrm{dom}\, playlistSubscribers \mid \# playlistSubscribers(p) \geq 100\}$

(c)

C == $\mu\, u\colon\ \mathrm{dom}\, loved \bullet \forall\, v\colon\ \mathrm{dom}\, loved \bullet u \neq v \land \# loved(u) > \# loved(v)$

(d)

D == $\mu\, s\colon songs \bullet \forall\, t\colon songs \bullet s \neq t \land \#\{u\colon UserId \mid s \in loved(u)\} > \#\{u\colon UserId \mid t \in loved(u)\}$

Solution 51

(a)

Let's first define two helper functions:

$loveHateScore : SongId + \rightarrow N$

$\forall\ i : songs \mid \{u\colon UserId \mid i \in loved(u)\} \geq\ \{u\colon UserId \mid i \in hated(u)\} \Rightarrow$

loveHateScore(i) = $\{u\colon UserId \mid i \in loved(u)\}$ - $\{u\colon UserId \mid i \in hated(u)\}$

and

$\forall\ i : songs \mid \{u\colon\ UserId \mid i \in loved(u)\} <\ \{u\colon\ UserId \mid i \in hated(u)\} \Rightarrow$

loveHateScore(i) = 0

$$\begin{array}{|l} playlistCount : SongId \rightarrowtail \mathbb{N} \\ \hline \forall\, i\colon\ songs \bullet playlistCount(i) = \#\{p\colon\ \mathrm{dom}\, playlist \mid i \in \mathrm{ran}\, playlist(p)\} \end{array}$$

We then have:

$$\begin{array}{|l} length : SongId \rightarrowtail \mathbb{N} popularity : SongId \rightarrowtail \mathbb{N} \\ \hline \mathrm{dom}\, length(subseteq)(songs)\, \mathrm{dom}\, popularity(subseteq)(songs)\, \forall\, i\colon\ songs \bullet popularity(i) = loveHateScore(i) + \end{array}$$

(b)

$mostPopular : SongId$

$(\exists_1\ i : songs\ |\forall\, j\colon\ songs \bullet i \neq j \wedge popularity(i) > popularity(j)) \Rightarrow$

mostPopular = (mu $i : songs\ |\forall\, j\colon\ songs \bullet i \neq j \wedge popularity(i) > popularity(j))$

and

$\neg\ \exists_1\, i\colon\ songs \bullet \forall\, j\colon\ songs \bullet i \neq j \wedge popularity(i) > popularity(j) \Rightarrow mostPopular = nullSong$

(c) playlistsContainingMostPopularSong == $\{i\colon\ \mathrm{dom}\, playlists \mid mostPopular \in \mathrm{ran}\, playlists(i)\}$

Solution 52

(a)

$premiumPlays : seq(Play) \rightarrow seq(Play)$

premiumPlays($\langle\rangle$) = $\langle\rangle$

$\forall\ x : Play;\ s : seq(Play)\ |$

premiumPlays($\langle x\rangle\ ^s$) =$\langle x\rangle\ ^p remiumPlays(s) if userStatus(x.2) = premium$

premiumPlays(s) if userStatus(x.2) = standard

(Note: Uses camelCase for fuzz compatibility)

(b)

$standardPlays : seq(Play) \to seq(Play)$

standardPlays($\langle\rangle$) = $\langle\rangle$

$\forall\ x : Play;\ s : seq(Play)\ |$

standardPlays($\langle x\rangle\ ^s$) =$\langle x\rangle\ ^s tandardPlays(s) if userStatus(x.2) = standard$

standardPlays(s) if userStatus(x.2) = premium

(Note: Uses camelCase for fuzz compatibility)

(c)

$cumulativeLength : seq(Play) \to N$

cumulativeLength($\langle\rangle$) = 0

$\forall\ x : Play;\ s : seq(Play)\ |$

cumulativeLength($\langle x\rangle\ ^s$) = $length(x.1) + cumulativeLength(s)$

(Note: Uses camelCase for fuzz compatibility)