Propositional logic

Solution 1

(a) $false(as(true \Rightarrow false) \Leftrightarrow false)$

(b) $true(as(false \Rightarrow false) \Leftrightarrow true)$

(c) $true(as(false \Rightarrow true) \Leftrightarrow true)$

(d) $true(as(false \Rightarrow false) \Leftrightarrow true)$

(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)

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

(d)

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

(e)

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

(f)

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

Solution 4

(a) $p \vee q \Leftrightarrow (\neg\, p \vee \neg\, q) \wedge q$ is $\neg\, a$ 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) $p \vee q \Leftrightarrow \neg\, p \wedge \neg\, q \vee q$ is $\neg\, a$ tautology. In this case, the proposition is false when p is true and q is false.

Solution 5

(a) $\exists\, d\colon Dog \bullet gentle(d) \wedge well\_trained(d)$

(b) $\forall\, d\colon Dog \bullet neat(d) \wedge well\_trained(d) \Rightarrow attractive(d)$

(c) $\exists\, d\colon Dog \bullet gentle(d) \Rightarrow \forall\, t\colon Trainer \bullet groomed(d, t)$

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) $\forall\, x\colon \mathbb{N} \bullet x \geq z$

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

(c) $x + 3 > 0 \wedge \forall\, z\colon \mathbb{N} \bullet z \geq x + 3$

Equality

Solution 9

(a)

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

(b)

$$\begin{array}{l}
\exists\, x, y\colon \mathbb{N} \bullet x + y = 4 \wedge x < y \\
\Leftrightarrow \exists\, x, y\colon \mathbb{N} \bullet y = 4 - x \wedge x < y \\
\Leftrightarrow \exists\, x\colon \mathbb{N} \bullet 4 - x \in \mathbb{N} \wedge x < 4 - x \\
\Leftrightarrow true
\end{array}$$

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)

$$\begin{aligned}
&\exists x : \mathbb{N} \bullet x = 1 \wedge x > y \vee x = 2 \wedge x > z \\
&\Leftrightarrow \exists x : \mathbb{N} \bullet x = 1 \wedge x > y \vee \exists x : \mathbb{N} \bullet x = 2 \wedge x > z \\
&\Leftrightarrow 1 \in \mathbb{N} \wedge 1 > y \vee \exists x : \mathbb{N} \bullet x = 2 \wedge x > z \\
&\Leftrightarrow 1 \in \mathbb{N} \wedge 1 > y \vee 2 \in \mathbb{N} \wedge 2 > z \\
&\Leftrightarrow 1 > y \vee 2 > z
\end{aligned}$$

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 $\neg$ *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

(a) $\mu\, m\colon Mountain \bullet \forall\, n\colon Mountain \bullet height(n) \leq height(m).height(m)$

(b) $\mu\, c\colon Chapter \bullet \exists_1\, d\colon Chapter \bullet length(d) > length(c).length(c)$

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

Deductive proofs

Solution 13

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

Solution 14

In one direction:

$$\dfrac{\ulcorner p\land q \Leftrightarrow p\urcorner^{[1]} \quad \dfrac{\ulcorner p \urcorner^{[2]} \quad \dfrac{\dfrac{\dfrac{}{p\land q}\,[\text{derived}]}{p\land q}\,[\Rightarrow \text{elimfrom1} \land 2]}{q}\,[\land\text{-elim}^{[3]}]}{p\Rightarrow q}\,[\Rightarrow\text{-intro}^{[2]}]}{(p\land q\Leftrightarrow p)\Rightarrow(p\Rightarrow q)}\,[\Rightarrow\text{-intro}^{[1]}]$$

and the other:

$$\dfrac{\ulcorner p\Rightarrow q\urcorner^{[1]} \quad \dfrac{\dfrac{\ulcorner p\land q\urcorner^{[2]} \quad \dfrac{}{p}\,[\land\text{-elim}^{[2]}]}{p\land q\Rightarrow p}\,[\Rightarrow\text{-intro}^{[2]}] \quad \dfrac{\ulcorner p\urcorner^{[3]} \quad \dfrac{}{p\land q}\,[\land \text{ introfrom1}\land 3]}{p\Rightarrow p\land q}\,[\Rightarrow\text{-intro}^{[3]}]}{p\land q\Leftrightarrow p}\,[\Leftrightarrow\text{ intro}]}{(p\Rightarrow q)\Rightarrow(p\land q\Leftrightarrow p)}\,[\Rightarrow\text{-intro}^{[1]}]$$

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

Solution 15

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

Solution 16

In one direction:

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

In the other:

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

Solution 17

In one direction:

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

and the other:

$$\cfrac{\ulcorner (p \vee q) \wedge (p \vee r) \urcorner^{[1]} \qquad \cfrac{}{p \vee q \wedge r}\,[\vee\text{ elimfrom2} \wedge 3]}{(p \vee q) \wedge (p \vee r) \Rightarrow p \vee q \wedge r}\,[\Rightarrow\text{-intro}^{[1]}]$$

Solution 18

In one direction:

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

and the other:

$$\dfrac{\ulcorner \neg\, p \lor q \urcorner^{[3]} \qquad \dfrac{\ulcorner p \urcorner^{[4]} \quad \dfrac{}{q}\,[\lor \text{ elim} \land \text{false-elim}^{[3]}]}{p \Rightarrow q}\,[\Rightarrow \text{-intro}^{[4]}]}{\neg\, p \lor q \Rightarrow (p \Rightarrow q)}\,[\Rightarrow \text{-intro}^{[3]}]$$

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) $\{z : \mathbb{Z} \mid 0 \leq z \land z \leq 100\}$

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

(c) $\{z : \mathbb{Z} \mid z \mathsf{\ mod\ } 2 = 0 \lor z \mathsf{\ mod\ } 3 = 0 \lor z \mathsf{\ mod\ } 5 = 0\}$

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) $\{n : \mathbb{P}\{0, 1\}\}$

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

Solution 23

(a)

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

(b)

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

Solution 24

(a)

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

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

(b)

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

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

(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:

$$Couples == \{s : \mathbb{P}\, Person \mid \# s = 2\}$$

Solution 25

(a) $\emptyset[\mathbb{N}] \in \emptyset[\mathbb{P}\,\mathbb{N}]$

(b) $\emptyset[\mathbb{N} \times \mathbb{N}] \subseteq \emptyset[\mathbb{N}] \times \emptyset[\mathbb{N}]$

(c) $\emptyset[\mathbb{N}] \times \{\emptyset[\mathbb{N}]\} \subseteq \emptyset[\mathbb{N} \times \mathbb{P}\,\mathbb{N}]$

Solution 26

We may define notin using our built-in operator (notin is already implemented as a binary operator mapping to $\notin$)

Relations

Solution 27

(a)

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

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

(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

$$\mid \; \mathit{childOf} : \mathit{Person} \leftrightarrow \mathit{Person}$$

(a)

$$\mathit{parentOf} == \mathit{childOf}^{-1}$$

This is a good example of how there are many different ways of writing the same thing. An alternative abbreviation would *be* : $\mathit{parentOf} == \{x, y : \mathit{Person} \mid x \mapsto y \in \mathit{childOf} \bullet y \mapsto x\}$. Or via an axiomatic definition with $\mathit{parentOf} : \mathit{Person} <\!\!\rightarrow \mathit{Personandwhereclauseparent Of} =$ childOf .

(b)

$$\mathit{siblingOf} == (\mathit{childOf} \circ \mathit{parentOf}) \setminus \mathrm{id}[\mathit{Person}]$$

(c)

$$\mathit{cousinOf} == \mathit{childOf} \circ \mathit{siblingOf} \circ \mathit{parentOf}$$

(d)

$$\mathit{ancestorOf} == \mathit{parentOf}^{+}$$

Solution 31

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

(a)

$$R == \{a, b \colon \mathbb{N} \mid b = a \lor b = a\}$$

(b)

$$S == \{a, b \colon \mathbb{N} \mid b = a \lor b = a\}$$

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

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

Solution 32

(a)

$$\begin{aligned}
& x \mapsto y \in A \lhd B \lhd R \\
& \Leftrightarrow x \in A \land x \mapsto y \in (B \lhd R) \\
& \Leftrightarrow x \in A \land x \in B \land x \mapsto y \in R \\
& \Leftrightarrow x \in A \cap B \land x \mapsto y \in R \\
& \Leftrightarrow x \mapsto y \in A \cap B \lhd R
\end{aligned}$$

(b)

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

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

(a)

$$\begin{array}{|l} children : Person \to \mathbb{P}\, Person \\ \hline children = \{p : Person \bullet p \mapsto (parentOf \,(\!|\, \{p\} \,|\!))\} \end{array}$$

(b)

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

Solution 36

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

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

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)

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

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

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

Solution 39

(a)

$$
\begin{array}{|l}
largeCoins : Collection \rightarrow \mathbb{N} \\
\hline
\forall\, c : Collection \bullet largeCoins(c) = c(large)
\end{array}
$$

(b)

$$
\begin{array}{|l}
addCoin : Collection \times Coin \rightarrow Collection \\
\hline
\forall\, c : Collection \bullet \forall\, d : Coin \bullet addCoin(c, d) = c \cup [\![d]\!]
\end{array}
$$

Modelling

Solution 40

Note: Refactored to use schemas with named fields instead of tuples for fuzz compatibility.

Changed underscore identifiers to camelCase for fuzz compatibility.

$$[\mathit{Title}, \mathit{Length}, \mathit{Viewed}]$$

$$\begin{array}{|l}
\mathit{Programme} \\ \hline
\mathit{title} : \mathit{Title} \\
\mathit{length} : \mathit{Length} \\
\mathit{viewed} : \mathit{Viewed} \\ \hline
\end{array}$$

(a)

$$\begin{array}{|l}
\mathit{hd} : \mathrm{seq}\, \mathit{Programme} \\ \hline
\mathit{cumulativeTotal}(\mathit{hd}) \leq 12000 \\
\forall\, p : \mathrm{ran}\, \mathit{hd} \bullet p.\mathit{length} \leq 360
\end{array}$$

Note that cumulativeTotal is defined in part (d).

(b)

Assuming hd is defined as in part (a):

$p : \mathit{ranhd} \mid p.\mathit{length} > 120$. p.title

(c)

These can be defined recursively:

$$\begin{array}{|l}
\mathit{viewedProgrammes} : \mathrm{seq}\, \mathit{Programme} \rightarrow \mathrm{seq}\, \mathit{Programme} \\ \hline
\mathit{viewedProgrammes}(\langle\rangle) = \langle\rangle \\
\forall\, x : \mathit{Programme} \bullet \forall\, s : \mathrm{seq}\, \mathit{Programme} \bullet \mathit{viewedProgrammes}(\langle x\rangle \frown s) = (\mathrm{if}\ x.\mathit{viewed} = \mathit{yes}\ \mathrm{then}\ \langle x\rangle \frown \mathit{viewedProgra}
\end{array}$$

(d)

$$\begin{array}{|l}
\mathit{cumulativeTotal} : \mathrm{seq}\, \mathit{Programme} \rightarrow \mathbb{N} \\ \hline
\mathit{cumulativeTotal}(\langle\rangle) = 0 \\
\forall\, x : \mathit{Programme} \bullet \forall\, s : \mathrm{seq}\, \mathit{Programme} \bullet \mathit{cumulativeTotal}(\langle x\rangle \frown s) = x.\mathit{length} + \mathit{cumulativeTotal}(s)
\end{array}$$

(e)

Assuming hd is defined as in part (a), the title of the longest programme:

(mu p : $ranhd$ $|\forall\, q\colon \mathrm{ran}\, hd \bullet p \neq q \land p.length > q.length|$ p.title)

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

(f)

axdef

$totalsByTitle$: $seq(Programme) \rightarrow (Title{+} \rightarrow Length)$

where

$\forall$ s : $seq(Programme)$ |

totalsByTitle(s) = t : $Title$ | $(\exists\, p\colon \mathrm{ran}\, s \bullet p.title = t)$.

t $\mapsto$ cumulativeTotal(s $\rhd$ l : $Length$; v : $Viewed$. (t, l, v))

end

(Note: Complex nested set comprehensions - may require simplification for implementation)

(g)

$$\begin{array}{|l} removeTheLongestViewed : \mathrm{seq}\, Programme \rightarrow \mathrm{seq}\, Programme \\ \hline \forall\, s\colon \mathrm{seq}\, Programme \bullet removeTheLongestViewed(s) = s \rhd \{x\colon \mathrm{ran}\, s \mid x \neq longestViewed(s)\} \end{array}$$

Where longestViewed is defined as:

$$\begin{array}{|l} longestViewed : \mathrm{seq}\, Programme \rightarrowtail Programme \\ \hline \forall\, s\colon \mathrm{seq}\, Programme \bullet longestViewed(s) = \mu\, p\colon \mathrm{ran}\, s \bullet p.viewed = yes \land \forall\, q\colon \mathrm{ran}\, s \bullet p \neq q \land q.viewed = yes \land p.leng \end{array}$$

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

(h)

$$\begin{array}{|l} sortByLength : \mathrm{seq}\, Programme \rightarrow \mathrm{seq}\, Programme \\ \hline \forall\, x\colon \mathrm{seq}\, Programme \bullet items(sortByLength(x)) = items(x) \land \forall\, i, j\colon \mathrm{dom}\, sortByLength(x) \bullet i < j \Rightarrow sortByLength(x \end{array}$$

Solution 41

$[Year, Course, Lecturer]$

$$\begin{array}{|l}
\hline
\textbf{Entry} \\
year : Year \\
course : Course \\
code : \mathbb{N} \\
lecturer : Lecturer \\
enrolled : \mathbb{N} \\
completed : \mathbb{N} \\
grade : \mathbb{N} \\
\hline
\end{array}$$

$Table == \mathbb{N} \rightarrowtail Entry$

(a)

$$\begin{array}{|l}
records : Year \rightarrowtail Table \\
\hline
\mathrm{dom}\, records = 1993 \mathrel{..} current \\
\forall\, y : \mathrm{dom}\, records \bullet \#records(y) \leq 50 \\
\forall\, y : \mathrm{dom}\, records \bullet \forall\, e : \mathrm{ran}\, records(y) \bullet e.year = y \\
\forall\, r : \mathrm{ran}\, records \bullet \forall\, i1, i2 : \mathrm{dom}\, r \bullet i1 \neq i2 \land r(i1).year = r(i2).year \Rightarrow r(i1).code \neq r(i2).code
\end{array}$$

(b) $i(\{e : Entry \mid \exists\, r : \mathrm{ran}\, records \bullet e \in \mathrm{ran}\, r \land e.code = 479\})$
$ii(\{e : Entry \mid \exists\, r : \mathrm{ran}\, records \bullet e \in \mathrm{ran}\, r \land e.completed > e.enrolled\})$
$iii(\{e : Entry \mid \exists\, r : \mathrm{ran}\, records \bullet e \in \mathrm{ran}\, r \land e.grade \geq 70\})$
$iv(\{c : Course \mid \forall\, r : \mathrm{ran}\, records \bullet \forall\, e : \mathrm{ran}\, r \bullet e.course = c \Rightarrow e.grade \geq 70\})$
$v(\{y : Year \mid y \in \mathrm{dom}\, records \bullet y \mapsto \{l : Lecturer \mid \#\{e : \mathrm{ran}\, records(y) \mid e.lecturer = l\} > 6\}\})$

(c)

$$\begin{array}{|l}
\hline
\forall\, x : Entry \bullet \forall\, s : \mathrm{seq}\, Entry \bullet courses479(\langle\rangle) = \langle\rangle \land courses479(\langle x \rangle \frown s) = (\text{if } x.code = 479 \text{ then } \langle x \rangle \frown courses479(
\end{array}$$

(d)

$$\begin{array}{|l}
\hline
\forall\, x : Entry \bullet \forall\, s : \mathrm{seq}\, Entry \bullet total(\langle\rangle) = 0 \land total(\langle x \rangle \frown s) = x.enrolled + total(s)
\end{array}$$

Solution 42

(a)

$[Person]$

$$\begin{array}{|l}
State : \mathbb{P}\,\mathrm{seq}(\mathrm{iseq}\, Person) \\
\hline
\forall\, s : State \bullet \forall\, i, j : \mathrm{dom}\, s \bullet i \neq j \land \mathrm{ran}\, s(i) \cap \mathrm{ran}\, s(j) = \{\}
\end{array}$$

(b)

$$\begin{array}{|l}
add : \mathbb{N} \times Person \times State \rightarrowtail State \\
\hline
\forall\, n : \mathbb{N} \bullet \forall\, p : Person \bullet \forall\, s : State \bullet n \in \mathrm{dom}\, s \land p \notin \bigcup \mathrm{ran}\,\mathrm{ran}\, s . add(n, p, s) = s \oplus \{n \mapsto s(n)^{\langle p \rangle}\}
\end{array}$$

Solution 43

Note: Assuming given types Cinema, Film, Date, Booking and a bookings relation.

The problem statement *defines* : *bookings*: (Cinema cross Film) $\nrightarrow$ seq Booking

where Booking is a triple of (bookingRef, startDay, endDay) : (N cross N cross N)

$[Cinema, Film, Date]$

Booking
$ref : \mathbb{N}$
$startDay : \mathbb{N}$
$endDay : \mathbb{N}$

Assuming: bookings : (Cinema cross Film) $\nrightarrow$ seq Booking

(a)

(i) $\forall\ i : dombookings\ |\forall\, x, y\colon bookings(i) \bullet x \neq y|$ (x.startDay..x.endDay) intersect (y.startDay..y.endDay) $= \emptyset$

(ii) $\forall\ i : dombookings\ |\forall$ x : $bookings(i)|\ \{x.startDay, x.endDay\}$ subseteq 1..max(i.1)

(Note: Assuming max is a function on Cinema that returns the maximum day number)

(iii) $\forall\ i : dombookings\ |\forall\, b\colon bookings(i) \bullet b.startDay \leq b.endDay$

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

(b)

Assuming Banbury : Cinema and bookings is defined

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

(ii) $\{i\colon \mathrm{dom}\, bookings \mid i.1 = Banbury \wedge \exists\, b\colon bookings(i) \bullet 50 \in b \bullet startDay \mathrel{..} b.endDay\}$

(iii) Assuming Room is Cinema, returning unique cinema/film pairs:

$c : Cinema;\ f : Film\ |\exists\, i\colon \mathrm{dom}\, bookings \bullet i.1 = c \wedge i.2 = f.$ (c, f)

(iv) $\{c\colon Cinema \mid \exists\, i\colon \mathrm{dom}\, bookings \bullet i.1 = c \wedge \#bookings(i) \geq 10\}$

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)

$$
\begin{array}{|l}
count : Tree \to \mathbb{N} \\
\hline
count(stalk) = 0 \\
\forall\, n{:}\, \mathbb{N} \bullet count(leaf(n)) = 1 \\
\forall\, t1, t2{:}\ Tree \bullet count(branch(t1, t2)) = count(t1) + count(t2)
\end{array}
$$

(b)

$$
\begin{array}{|l}
flatten : Tree \to \operatorname{seq} \mathbb{N} \\
\hline
flatten(stalk) = \langle\rangle \\
\forall\, n{:}\, \mathbb{N} \bullet flatten(leaf(n)) = \langle n \rangle \\
\forall\, t1, t2{:}\ Tree \bullet flatten(branch(t1, t2)) = flatten(t1)^{flatten}(t2)
\end{array}
$$

Solution 47

First, exhibit the induction principle for the free type:

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

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

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

$$[SongId, UserId, PlaylistId, Playlist]$$

$$
\begin{array}{|l}
songs : \mathbb{F}\ SongId \\
users : \mathbb{F}\ UserId \\
playlists : PlaylistId \pfun Playlist \\
playlistOwner : PlaylistId \pfun UserId \\
playlistSubscribers : PlaylistId \pfun \mathbb{F}_1\ UserId \\
\hline
\forall i : \mathrm{dom}\, playlists \bullet \mathrm{ran}\, playlists(i) \subseteq songs \\
\mathrm{dom}\, playlistOwner \subseteq \mathrm{dom}\, playlists \\
\mathrm{ran}\, playlistOwner \subseteq users \\
\mathrm{dom}\, playlistSubscribers \subseteq \mathrm{dom}\, playlists \\
\forall i : \mathrm{dom}\, playlistSubscribers \bullet playlistSubscribers(i) \subseteq users \\
\forall i : \mathrm{dom}\, playlists \bullet playlistOwner(i) \in playlistSubscribers(i)
\end{array}
$$

Solution 49

$$
\begin{array}{|l}
hated : UserId \pfun \mathbb{F}\ SongId \\
loved : UserId \pfun \mathbb{F}\ SongId \\
\hline
\mathrm{dom}\, hated \subseteq users \\
\forall i : \mathrm{dom}\, hated \bullet hated(i) \subseteq songs \\
\mathrm{dom}\, loved \subseteq users \\
\forall i : \mathrm{dom}\, loved \bullet loved(i) \subseteq songs \\
\forall i : \mathrm{dom}\, hated \cup \mathrm{dom}\, loved \bullet hated(i) \cap loved(i) = \emptyset
\end{array}
$$

Solution 50

(a)

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

(b)

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

(c)

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

(d)

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

Solution 51

(a)

$$\begin{array}{|l} loveHateScore : SongId \pfun \mathbb{N} \\ \hline \forall i : songs \bullet loveHateScore(i) = (\text{if } \#\{u : UserId \mid i \in loved(u)\} \geq \#\{u : UserId \mid i \in hated(u)\} \text{ then } \#(\{u : UserId \end{array}$$

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

$$\begin{array}{|l} length : SongId \pfun \mathbb{N} \\ popularity : SongId \pfun \mathbb{N} \\ \hline \operatorname{dom} length \subseteq songs \\ \operatorname{dom} popularity \subseteq songs \\ \forall i : songs \bullet popularity(i) = loveHateScore(i) + playlistCount(i) \end{array}$$

(b)

$$\begin{array}{|l} mostPopular : SongId \\ \hline mostPopular = (\text{if } \exists_1 i : songs \bullet \forall j : songs \bullet i \neq j \wedge popularity(i) > popularity(j) \text{ then } \mu i : songs \bullet \forall j : songs \bullet i \neq \end{array}$$

(c)

$$playlistsContainingMostPopularSong == \{i : \operatorname{dom} playlists \mid mostPopular \in \operatorname{ran} playlists(i)\}$$

Solution 52

(a)

$$\begin{array}{|l} premiumPlays : \operatorname{seq} Play \rightarrow \operatorname{seq} Play \\ \hline premiumPlays(\langle\rangle) = \langle\rangle \\ \forall x : Play \bullet \forall s : \operatorname{seq} Play \bullet premiumPlays(\langle x \rangle \frown s) = (\text{if } userStatus(x.2) = premium \text{ then } \langle x \rangle \frown premiumPlays(s) \text{ e} \end{array}$$

(b)

$$\begin{array}{|l} standardPlays : \operatorname{seq} Play \rightarrow \operatorname{seq} Play \\ \hline standardPlays(\langle\rangle) = \langle\rangle \\ \forall x : Play \bullet \forall s : \operatorname{seq} Play \bullet standardPlays(\langle x \rangle \frown s) = (\text{if } userStatus(x.2) = standard \text{ then } \langle x \rangle \frown standardPlays(s) \text{ el} \end{array}$$

(c)

$$\begin{array}{|l} cumulativeLength : \operatorname{seq} Play \rightarrow \mathbb{N} \\ \hline cumulativeLength(\langle\rangle) = 0 \\ \forall x : Play \bullet \forall s : \operatorname{seq} Play \bullet cumulativeLength(\langle x \rangle \frown s) = length(x.1) + cumulativeLength(s) \end{array}$$