

Propositional logic

Solution 1

(a)

$$\text{false}(\text{as}(\text{true} \Rightarrow \text{false}) \Leftrightarrow \text{false})$$

(b)

$$\text{true}(\text{as}(\text{false} \Rightarrow \text{false}) \Leftrightarrow \text{true})$$

(c)

$$\text{true}(\text{as}(\text{false} \Rightarrow \text{true}) \Leftrightarrow \text{true})$$

(d)

$$\text{true}(\text{as}(\text{false} \Rightarrow \text{false}) \Leftrightarrow \text{true})$$

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

Solution 2

(a)

p	q	$p \wedge q$	$(p \wedge q) \Rightarrow p$
t	t	t	t
t	f	f	t
f	t	f	t
f	f	f	t

(b)

p	q	$p \wedge q$	$\neg p$	$\neg p \Rightarrow (p \wedge q)$	$(\neg p \Rightarrow (p \wedge q)) \Leftrightarrow p$
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	$p \Rightarrow q$	$p \wedge (p \Rightarrow q)$	$(p \wedge (p \Rightarrow q)) \Rightarrow q$
t	t	t	t	t
t	f	f	f	t
f	t	t	f	t
f	f	t	f	t

Solution 3

(a)

$$\begin{aligned}
 p \Rightarrow \neg p & \\
 \Leftrightarrow \neg p \vee \neg p & \quad [\Rightarrow] \\
 \Leftrightarrow \neg p & \quad [\text{idempotence}]
 \end{aligned}$$

(b)

$$\begin{aligned}
 \neg p \Rightarrow p & \\
 \Leftrightarrow \neg \neg p \vee p & \quad [\Rightarrow] \\
 \Leftrightarrow p \vee p & \quad [\neg \neg] \\
 \Leftrightarrow p & \quad [\text{idempotence}]
 \end{aligned}$$

(c)

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

(d)

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

(e)

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

(f)

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

Solution 4

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

Solution 5

(a)

$$\exists d: Dog \bullet \text{gentle}(d) \wedge \text{well_rained}(d)$$

(b)

$$\forall d: Dog \bullet \text{neat}(d) \wedge \text{well_rained}(d) \Rightarrow \text{attractive}(d)$$

(c)

(Requires nested quantifier in implication - parser limitation)

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 $p \Leftrightarrow x \neq 1$.

(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 $q \Leftrightarrow x \neq 3$.

Solution 8

(a)

$$\forall x: N \bullet x \geq z$$

Equality**Solution 9**

(d)

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

Solution 10

As discussed, the quantifier exists₁ 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

$$\exists_1 n: N \bullet \forall m: N \bullet n \leq m$$

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

$$\mu n: N \bullet \forall m: N \bullet n \leq m$$

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

Solution 11

(a)

$(\mu a: N \bullet a = a) = 0$ is a provable statement, since 0 is the only natural number with the specified property.

(b)

$(\mu b: N \bullet b = b) = 1$ is not provable. The specified property is true of both 0 and 1, and thus the value of the mu-expression is undefined.

(c)

$(\mu c: N \bullet c > c) = (\mu c: N \bullet c > c)$ 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)

$(\mu d: N \bullet d = d) = 1$ 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)

$$\mu m: Mountain \mid \forall n: Mountain \bullet height(n) \leq height(m) \bullet height(m)$$

(b)

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

(c)

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

Deductive proofs

Solution 13

$$\frac{\frac{\frac{p \wedge (p \Rightarrow q)}{p} [\wedge\text{-elim}^{[1]}] \quad \frac{p \wedge (p \Rightarrow q)}{p} [\wedge\text{-elim}^{[1]}]}{q} [\Rightarrow\text{ elim}]}{p \wedge q} [\wedge\text{ intro}]$$

$$\frac{\neg(p \wedge (p \Rightarrow q))^{[1]}}{p \wedge (p \Rightarrow q) \Rightarrow p \wedge q} [\Rightarrow\text{-intro}^{[1]}]$$

Solution 14

In one direction:

$$\frac{\frac{\frac{\frac{\overline{p \wedge q}}{p \wedge q} [\text{derived}] \quad \frac{\overline{p \wedge q}}{p \wedge q} [\Rightarrow\text{ elim from } 1 \wedge 2]}{\frac{\neg p^{[2]} \quad \frac{\overline{p \wedge q}}{q} [\wedge\text{-elim}^{[3]}]}{p \Rightarrow q} [\Rightarrow\text{-intro}^{[2]}]}{(p \wedge q \Leftrightarrow p)^{[1]}} \frac{\neg p^{[2]}}{p \Rightarrow q} [\Rightarrow\text{-intro}^{[1]}]}{(p \wedge q \Leftrightarrow p) \Rightarrow (p \Rightarrow q)}$$

and the other:

$$\frac{\frac{\frac{\neg p \wedge q^{[2]} \quad \neg p^{[2]}}{p \wedge q \Rightarrow p} [\Rightarrow\text{-intro}^{[2]}] \quad \frac{\neg p^{[3]} \quad \neg p \wedge q^{[1]}}{p \Rightarrow p \wedge q} [\Rightarrow\text{-intro}^{[3]}]}{p \wedge q \Leftrightarrow p} [\Leftrightarrow\text{ intro}]}{(p \Rightarrow q) \Rightarrow (p \wedge q \Leftrightarrow p)} [\Rightarrow\text{-intro}^{[1]}]$$

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

Solution 15

$$\frac{\frac{\frac{\frac{\neg(p \Rightarrow q) \wedge \neg q^{[1]}}{\neg p} \quad \frac{\neg p^{[2]}}{\frac{\frac{q}{\neg \neg q^{[1]}} \quad \frac{\neg p^{[2]}}{\frac{\neg p}{\neg p}}}{\neg p}}}{\neg p} \quad \frac{\neg \neg q^{[1]}}{\text{false}}}{\text{false-elim}^{[2]}}}{\neg p} \quad \frac{\neg p}{\neg p}}{\neg p} \quad [\neg\text{-intro}^{[1]}]$$

Solution 16

In one direction:

$$\frac{\frac{\frac{\frac{\frac{\neg p^{[1]} \quad \neg r}{p \wedge r} \quad [\wedge \text{ intro}]}{\frac{\neg p \wedge \neg r}{p \wedge q \vee p \wedge r} \quad [\vee \text{ intro}]} \quad \frac{\frac{\neg p^{[1]} \quad \neg q}{p \wedge q} \quad [\wedge \text{ intro}]}{\frac{\neg p \wedge \neg q}{p \wedge q \vee p \wedge r} \quad [\vee \text{ intro}]} \quad \frac{\neg q \vee r^{[1]}}{\frac{\neg p \wedge (q \vee r)^{[1]}}{\frac{\neg p \wedge (q \vee r)}{p \wedge (q \vee r) \Rightarrow p \wedge q \vee p \wedge r}} \quad [\neg\text{-intro}^{[1]}]} \quad [\vee\text{-elim}^{[2]}]}{p \wedge (q \vee r) \Rightarrow p \wedge q \vee p \wedge r} \quad [\Rightarrow\text{-intro}^{[1]}]}{p \wedge (q \vee r) \Rightarrow p \wedge q \vee p \wedge r} \quad [\Rightarrow\text{-intro}^{[1]}]$$

In the other:

$\frac{}{p}$ [\wedge elim]	$\frac{}{q \vee r}$ [\vee intro]
$\frac{}{p}$ [\wedge elim]	$\frac{}{p \wedge (q \vee r)}$ [\wedge intro]
$\frac{}{q \vee r}$ [\vee intro]	
$\frac{}{p \wedge (q \vee r)}$ [\wedge intro]	
$\frac{\neg case1 \vee case2}{p \wedge (q \vee r)}$ [\neg -intro ^[3]]	$\frac{\neg case1 \vee case2}{p \wedge (q \vee r)}$ [\neg -intro ^[3]]
$\frac{}{p \wedge q \vee p \wedge r}$ [\neg -intro ^[3]]	$\frac{p \wedge q \vee p \wedge r}{p \wedge (q \vee r)}$ [\Rightarrow -intro ^[3]]
$\frac{}{p \wedge q \vee p \wedge r \Rightarrow p \wedge (q \vee r)}$	$\frac{}{p \wedge q \vee p \wedge r \Rightarrow p \wedge (q \vee r)}$ [\vee -elim ^[4]]

Solution 17

In one direction:

$$\frac{\vdash p \vee q \wedge r \neg [3] \quad \overline{(p \vee q) \wedge (p \vee r)}}{p \vee q \wedge r \Rightarrow (p \vee q) \wedge (p \vee r)} [\neg \text{intro}^{[3]}]$$

and the other:

$$\frac{\Gamma(p \vee q) \wedge (p \vee r) \neg [1] \quad \Gamma p \vee q \wedge r \neg [2]}{(p \vee q) \wedge (p \vee r) \Rightarrow p \vee q \wedge r} [\Rightarrow\text{-intro}^{[1]}]$$

Solution 18

In one direction:

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

and the other:

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

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)

$$\{z : Z \mid 0 \leq z \wedge z \leq 100\}$$

(b)

$$\{z : Z \mid z = 10\}$$

(c)

$$\{z : Z \mid z \bmod 2 = 0 \vee z \bmod 3 = 0 \vee z \bmod 5 = 0\}$$

Solution 22

(a)

$$\{n : N \mid n \leq 4 \bullet n^2\}$$

(b)

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

(c)

n : P 0, 1

(d)

n : P 0, 1 — true . (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 × Z

(b)

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

StrictlyPositivePairs == { m, n : Z | m > 0 ∧ n > 0 • (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 : 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:

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

(b)

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

(c)

$\{\emptyset\}$

(d)

$\{\emptyset\}$

Solution 28

(a)

$$\text{dom } R = \{0, 1, 2\}$$

(b)

$$\text{ran } R = \{1, 2, 3\}$$

(c)

$$\{1, 2\} \triangleleft R = \{1 \mapsto 2, 1 \mapsto 3, 2 \mapsto 3\}$$

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

$$| \quad childOf : Person \leftrightarrow Person$$

(a)

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

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

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

Or, via an axiomatic definition:

$$\frac{| \quad parentOf : Person \leftrightarrow Person}{| \quad parentOf = \text{childOf}^{-1}}$$

(b)

$$\text{siblingOf} == (\text{childOf} \circ \text{parentOf}) \setminus id$$

(c)

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

(d)

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

Solution 31

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

(a)

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

(b)

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

(c)

$$R+ == \{ a, b : N \mid b > a \}$$

(d)

$$R^* == \{ a, b : N \mid b \geq a \}$$

Solution 32

(a)

$$\begin{aligned} x \mapsto y \in A \triangleleft B \triangleleft R \\ \Leftrightarrow x \in A \wedge x \mapsto y \in (B \triangleleft R) \\ \Leftrightarrow x \in A \wedge x \in B \wedge x \mapsto y \in R \\ \Leftrightarrow x \in A \cap B \wedge x \mapsto y \in R \\ \Leftrightarrow x \mapsto y \in A \cap B \triangleleft R \end{aligned}$$

(b)

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

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)

axdef

children : Person - \hookrightarrow P Person

where

children = p : Person . p — \hookrightarrow parentOf(— p —)

end

(b)

axdef

number_of_random_children : Person —> N

where

number_of_random_children = p : Person . p | —> (parentOf o parentOf)(| p |)

end

Solution 36

(Requires power set, function types, and ran keyword)

axdef

number_{of}rivers : (Drivers < -> Cars) -> (Cars -> N)

where

forall r : Drivers $\vdash \ddot{c}$ Cars — number_{of}rivers(r) = c : ranr.c | -> { d : Drivers | d \mapsto c \in r }

end

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)

$$\frac{| f : Place \rightarrow \mathbb{P} \text{ Place}}{\forall p : Place \bullet f(p) = \{q : Place \mid p \mapsto q \in \text{ran } \textit{trains}\}}$$

(b)

$\{p : Place \mid \exists_1 x : \text{dom } \textit{trains} \bullet \textit{trains}(x).2 = p\}$

(c)

(mu p : Place — $\forall q : Place \bullet p \neq q$ — $\{x : \text{dom } \textit{trains} \mid \textit{trains}(x).2 = p\}$;
 $\{x : \text{dom } \textit{trains} \mid \textit{trains}(x).2 = q\}$)

(Blocked by: nested quantifiers in mu with multiple pipes - parser ambiguity)

Solution 39

(a)

$\text{large}_c oins : Collection \rightarrow N$

$\forall c : Collection \bullet \text{large}_c oins(c) = c(\text{large})$

(Blocked by: underscore in identifier for fuzz compatibility)

(b)

$\text{add}_{c\text{oin}} : \text{Collection} * \text{Coin} -> \text{Collection}$

$\forall c : \text{Collection} \bullet \forall d : \text{Coin} \bullet \text{add}_{c\text{oin}}(c, d) = c \cup \llbracket d \rrbracket$

(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)

$\text{hd} : \text{seq}(\text{Title} * \text{Length} * \text{Viewed})$

$\text{cumulative}_t \text{otalhd} <= 12000$

$\forall p : \text{ran } \text{hd} \bullet p.2 \leq 360$

Note that $\text{cumulative}_t \text{otal}$ is defined in $\text{part}(d)$.

(b)

$\{p : \text{ran } \text{hd} \mid p.2 > 120 \bullet p.1\}$

(c)

These can be defined recursively:

viewed $\text{if}_i = \text{if}_i$

viewed $\text{if}_i^s = \text{if}x.3 = y \text{esthen} < x >^v \text{iewedselsevieweds}$

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

(d)

$$\frac{\text{cumulative}_t otal : \text{seq } Title * Length * Viewed \rightarrow N}{\text{cumulative}_t otal(\langle \rangle) = 0 \quad \forall x: Title * Length * Viewed \bullet \forall s: \text{seq } Title * Length * Viewed \bullet \text{cumulative}_t otal(s) = \text{if}_i^s}$$

(e)

$(\mu u p : \text{ran } \text{hd} — \forall q: \text{ran } \text{hd} \bullet p \neq q — p.2 \downarrow q.2 — p.1)$

(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

$g : \text{seq}(Title * Length * Viewed) \dashv_i \text{seq}(Title * Length * Viewed)$

where

$\forall s: \text{seq } Title * Length * Viewed \bullet g(s) = s —_i \{x: \text{ran } s \mid x \neq \text{longest}_v \text{iewed}(s)\}$

end

Where $\text{longest}_{\text{viewed}}$ is defined as

axdef

$\text{longest}_{\text{viewed}} : \text{seq}(\text{Title} * \text{Length} * \text{Viewed}) + - > \text{Title} * \text{Length} * \text{Viewed}$

where

$\forall s : \text{seq } \text{Title} * \text{Length} * \text{Viewed} \bullet \text{longest}_{\text{viewed}}(s) = (\mu p : \text{ran } s \bullet p.3 = \text{yes} \text{ and } \forall q : \text{ran } s \bullet p \neq q \wedge q.3 = \text{yes} \rightarrow p.2 \geq q.2)$

end

(Blocked by: nested quantifiers in mu expressions - parser limitation)

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

(h)

$$\boxed{\begin{array}{l} s : \text{seq } \text{Title} * \text{Length} * \text{Viewed} \rightarrow \text{seq } \text{Title} * \text{Length} * \text{Viewed} \\ \forall x : \text{seq } \text{Title} * \text{Length} * \text{Viewed} \bullet \text{items}(s(x)) = \text{items}(x) \wedge \forall i, j : \text{dom } s(x) \bullet i < j \Rightarrow s(x)(i).2 \geq s(x)(j).2 \end{array}}$$

Solution 41

(a)

axdef

records : Year — ζ Table

where

$\text{dom}(\text{records}) = 1993.. \text{current}$

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

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

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

end

(Blocked by: nested quantifiers in predicates - parser limitation)

(b)

(i)

$\{e: \text{Entry} \mid \exists r: \text{ran } \text{records} \bullet e \in \text{ran } r \wedge e.3 = 479\}$

ii

$\{e: \text{Entry} \mid \exists r: \text{ran } \text{records} \bullet e \in \text{ran } r \wedge e.6 > e.5\}$

iii

$\{e: \text{Entry} \mid \exists r: \text{ran } \text{records} \bullet e \in \text{ran } r \wedge e.7 \geq 70\}$

iv

$\{c: \text{Course} \mid \forall r: \text{ran } \text{records} \bullet \forall e: \text{ran } r \bullet e.2 = c \Rightarrow e.7 \geq 70\}$

v

$y : \text{Year} — y \text{ in } \text{dom } \text{records} . y —\zeta l : \text{Lecturer} — c : \text{ran}(\text{records } y) — c.4 = l \zeta 6$

(c)

axdef

where

$$\forall x: \text{Entry} \bullet \forall s: \text{seq Entry} \bullet 479_c \text{ourses}(<>) = <> \text{and} 479_c \text{ourses}(< x >^s) \\) = \text{if} x.3 = 479 \text{then } < x >^4 79_c \text{ourses}(s) \text{else} 479_c \text{ourses}(s)$$

end

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

(d)

$$\boxed{\forall x: \text{Entry} \bullet \forall s: \text{seq Entry} \bullet \text{total}(\langle \rangle) = 0 \wedge \text{total}(\langle x \rangle \cap s) = x.5 + \text{total}(s)}$$

Solution 42

[Person]

axdef

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

where

forall s : State — $\forall i, j: \text{dom } s \bullet i \neq j \text{— ran}(s(i)) \text{ intersect ran}(s(j)) =$

end

(Blocked by: nested quantifiers with semicolon bindings - parser limitation)

(b)

axdef

$\text{add} : N * \text{Person} * \text{State} \multimap \text{State}$

where

$\forall n: N \bullet \forall p: \text{Person} \bullet \forall s: \text{State} \bullet n \in \text{dom } s \wedge p \notin \text{bigcup}(\text{ran ran } s) \multimap$

$\text{add}(n, p, s) = s ++ n \multimap s(n) < p >$

end

(Blocked by: \multimap_i operator not implemented)

Solution 43

(a)

(i) forall $i : \text{dom bookings} \multimap \forall x, y: \text{bookings}(i) \bullet x \neq y \multimap (x.2..x.3) \text{ intersect } (y.2..y.3) =$

(ii) forall $i : \text{dom bookings} \multimap \forall x: \text{bookings}(i) \bullet \{x.2, x.3\} \subseteq \{1..max(i.1)\}$

(iii) forall $i : \text{dom bookings} \multimap \forall b: \text{bookings}(i) \bullet b.2 \leq b.3$

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

(Blocked by: nested quantifiers - parser limitation)

(b)

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

(ii) $i : \text{dom } bookings \mid i.1 = \text{Banbury} \text{ and } \exists b : bookings(i) \bullet 50 \in b.2 \dots b.3$

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

(iv) $r : \text{Room} \mid \exists i : \text{dom } bookings \bullet i.1 = r \text{ — } (\text{bookings}(i))_{i=10}$

(Blocked by: semicolon bindings in set comprehensions and nested quantifiers)

Free types and induction

Solution 44

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

$$\text{reverse}(\text{j}\text{l}^t) = \text{reverset}[cat.1a] = (\text{reverset})^{<} > [cat.1b]$$

$$\text{where cat.1a is j}\text{l}^s = \text{sandcat.1biss}^{<} > = s$$

and the second by

$$\text{reverse } ((\text{!x}_i \ u)^t) = \text{reverse}(< x >^t u^t)[\text{cat.2}]$$

$$= \text{reverse } (\text{u }^t)^{< x >} [\text{reverse.2}]$$

$$= (\text{reverse } t \ ^r \text{everse} u)^{< x >} [\text{anti-distributive}]$$

$$= \text{reverse } t \ (^r \text{everse} u^{< x >})[\text{cat.2}]$$

$$= \text{reverse } t \ ^r \text{everse}(< x >^u)[\text{reverse.2}]$$

Solution 45

The base case:

$$\text{reverse } (\text{reverse } \text{!}i) = \text{reverse } \text{!}i [\text{reverse.1}] = \text{!}i [\text{reverse.1}]$$

The inductive step:

$$\text{reverse } (\text{reverse } (\text{!x}_i \ ^t))$$

$$= \text{reverse } ((\text{reverse } t)^{< x >})[\text{reverse.2}]$$

$$= \text{reverse } (\text{!x}_i \ ^r \text{everse}(\text{reverset}))[\text{anti-distributive}]$$

$$= \text{reverse } (\text{!x}_i \ ^{< >} \ ^r \text{everse}(\text{reverset}))[\text{cat.1}]$$

$$= ((\text{reverse } \text{!}i)^{< x >})^r \text{everse}(\text{reverset})[\text{reverse.2}]$$

$$= (\text{!}\zeta < x >)^r \text{everse}(\text{reverset})[\text{reverse.1}]$$

$$= \text{!}\mathbf{x}_\zeta^r \text{everse}(\text{reverset})[\text{cat.1}]$$

$$= \text{!}\mathbf{x}_\zeta^t [\text{reverse}(\text{reverset}) = t]$$

Solution 46

(a)

$$\text{count} : \text{Tree} \dashv \zeta \text{ N}$$

$$\text{count stalk} = 0$$

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

$$\forall t_1, t_2 : \text{Tree} \bullet \text{count}(\text{branch}(t_1, t_2)) = \text{count} t_1 + \text{count} t_2$$

(Blocked by: recursive free types and pattern matching)

(b)

$$\text{flatten} : \text{Tree} \dashv \zeta \text{ seq N}$$

$$\text{flatten stalk} = \text{!}\zeta$$

$$\forall n : N \bullet \text{flatten}(\text{leaf } n) = \text{!}\mathbf{n}_\zeta$$

$$\forall t_1, t_2 : \text{Tree} \bullet \text{flatten}(\text{branch}(t_1, t_2)) = \text{flatten} t_1 \text{ } f \text{ } \text{flatten} t_2$$

(Blocked by: recursive free types and pattern matching)

Solution 47

First, exhibit the induction principle for the free type:

P stalk and $(\forall n: N \bullet P(\text{leaf } n))$ and $(\forall t_1, t_2: Tree \bullet \mathbb{P} t_1 \wedge \mathbb{P} t_2 \Rightarrow \mathbb{P} \text{ branch}(t_1, t_2))$

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

This gives three cases for the proof:

(flatten stalk) = $\text{if } [\text{flatten}] = 0 \text{ then } [] \text{ else count stalk [count]}$

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

Supplementary material : assignment practice

Solution 48

songs : F SongId

users : F UserId

playlists : PlaylistId $\dashv\vdash$ Playlist

playlistOwner : PlaylistId $\dashv\vdash$ UserId

playlistSubscribers : PlaylistId $\dashv\vdash$ F1 UserId

$\forall i : \text{dom } playlists \bullet \text{ran } playlists(i) \text{subseteq songs}$

$\text{dom}(\text{playlistOwner}) \text{subseteq dom}(playlists)$

$\text{ran}(\text{playlistOwner}) \text{subseteq users}$

$\text{dom}(\text{playlistSubscribers}) \text{subseteq dom}(playlists)$

$\forall i : \text{dom } playlistSubscribers \bullet playlistSubscribers(i) \text{subseteq users}$

$\forall i : \text{dom } playlists \bullet \text{playlistOwner}(i) \in \text{playlistSubscribers}(i)$

(Blocked by: F and F1 finite set types not implemented)

Solution 49

$\text{hated} : \text{UserId} \dashv\vdash \text{F SongId}$

$\text{loved} : \text{UserId} \dashv\vdash \text{F SongId}$

$\text{dom}(\text{hated}) \text{subseteq users}$

$\forall i : \text{dom } \text{hated} \bullet \text{hated}(i) \text{subseteq songs}$

$\text{dom}(\text{loved}) \text{subseteq users}$

$\forall i : \text{dom } \text{loved} \bullet \text{loved}(i) \text{subseteq songs}$

forall $i : \text{dom}(\text{hated}) \cup \text{dom}(\text{loved})$ — $\text{hated}(i) \cap \text{loved}(i) = \text{emptyset}$

(Blocked by: F finite set type not implemented)

Solution 50

(a)

 $A == \text{users} \text{ bigcup}(\text{ran}(\text{playlistSubscribers}))$

(Blocked by: bigcup operator not implemented)

(b)

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

(c)

 $C == (\mu u \ u : \ \text{dom}(\text{loved}) \ — \ \forall v: \ \text{dom } \text{loved} \bullet u \neq v \ — \ (\text{loved}(u)) \ i \\ (\text{loved}(v)))$

(Blocked by: nested quantifiers in mu - parser ambiguity)

(d)

 $D == (\mu s \ s : \ \text{songs} \ — \ \forall t: \ \text{songs} \bullet s \neq t \ — \ \{u: \ \text{UserId} \mid s \in \text{loved}(u)\} \ i \\ \{u: \ \text{UserId} \mid t \in \text{loved}(u)\})$

(Blocked by: nested quantifiers in mu - parser ambiguity)

Solution 51

(a)

Let's first define two helper functions:

loveHateScore : SongId \rightarrow N

$$\begin{aligned} & \text{forall } i : \text{songs} — \{u: UserId \mid i \in loved(u)\} \not\subseteq \{u: UserId \mid i \in hated(u)\} \\ \Rightarrow \end{aligned}$$

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

and

$$\begin{aligned} & \text{forall } i : \text{songs} — \{u: UserId \mid i \in loved(u)\} \sqsubset \{u: UserId \mid i \in hated(u)\} \\ \Rightarrow \end{aligned}$$

$$\text{loveHateScore}(i) = 0$$

$$\boxed{\begin{array}{l} \text{playlistCount : SongId} \rightarrow N \\ \forall i: \text{songs} \bullet \text{playlistCount}(i) = \#\{p: \text{dom playlist} \mid i \in \text{ran playlist}(p)\} \end{array}}$$

We then have:

$$\boxed{\begin{array}{l} \text{length : SongId} \rightarrow N \\ \text{popularity : SongId} \rightarrow N \\ \text{dom length} \subseteq \text{songs} \text{ dom popularity} \subseteq \text{songs} \forall i: \text{songs} \bullet \text{popularity}(i) = \text{loveHateScore}(i) + p \end{array}}$$

(b)

mostPopular : SongId

$$(\exists \text{exists1 } i : \text{songs} — \forall j: \text{songs} \bullet i \neq j — \text{popularity}(i) \not\subseteq \text{popularity}(j)) \Rightarrow$$

$$\text{mostPopular} = (\mu i : \text{songs} — \forall j: \text{songs} \bullet i \neq j — \text{popularity}(i) \not\subseteq \text{popularity}(j))$$

and

$$\begin{aligned} & \text{not } (\exists \text{exists1 } i : \text{songs} — \forall j: \text{songs} \bullet i \neq j — \text{popularity}(i) \not\subseteq \text{popularity}(j)) \\ \Rightarrow \end{aligned}$$

mostPopular = nullSong

(Blocked by: nested quantifiers in mu - parser ambiguity)

(c)

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

Solution 52

(a)

$\text{premiumPlays} : \text{seq(Play)} \dashv \vdash \text{seq(Play)}$

$\text{premiumPlays}(\text{!}\ddot{\text{c}}) = \text{!}\ddot{\text{c}}$

forall x : Play; s : seq(Play) —

$\text{premiumPlays}(\text{!}\ddot{\text{x}}\text{!}\ddot{\text{c}}^s) = <\text{x}>^p \text{remiumPlays}(s) \text{if userStatus(x.2) = premium}$

$\text{premiumPlays}(s) \text{ if userStatus(x.2) = standard}$

(Note: Uses camelCase for fuzz compatibility)

(b)

$\text{standardPlays} : \text{seq(Play)} \dashv \vdash \text{seq(Play)}$

$\text{standardPlays}(\text{!}\ddot{\text{c}}) = \text{!}\ddot{\text{c}}$

forall x : Play; s : seq(Play) —

$\text{standardPlays}(\text{if } \text{userStatus}(x.2) = \text{standard}$

$\text{standardPlays}(s) \text{ if } \text{userStatus}(x.2) = \text{premium}$

(Note: Uses camelCase for fuzz compatibility)

(c)

$\text{cumulativeLength} : \text{seq(Play)} \rightarrow \mathbb{N}$

$\text{cumulativeLength}(\text{if } s) = 0$

$\text{forall } x : \text{Play}; s : \text{seq(Play)} \text{ — }$

$\text{cumulativeLength}(\text{if } x.1) = \text{length}(x.1) + \text{cumulativeLength}(s)$

(Note: Uses camelCase for fuzz compatibility)