Programming Languages and Computation

Week 10: Encoding first-order data

* 1. Construct a bijection between the set $E = \{0, 2, 4, \dots\}$ of all even numbers, and the set $O = \{1, 3, 5, \dots\}$ of all odd numbers, and show that it is one.

Solution

The function

$$h : E \to O$$
$$h(e) = e + 1$$

will do. It is injective because

$$h(e) = h(e') \quad \Leftrightarrow \quad e + 1 = e' + 1 \quad \Leftrightarrow \quad e = e'$$

It is surjective because every odd number is of the form $2n + 1$. We then have that $2n$ is even, and $f(2n) = 2n + 1$.

* 2. In the reference material there is a proof that β is a bijection. Verify that $\beta : \mathbb{Z} \xrightarrow{\cong} \mathbb{N}$ is also an isomorphism: show that the function $\beta^{-1} : \mathbb{N} \to \mathbb{Z}$ defined in the lecture has the property that $\beta^{-1} \circ \beta = id_{\mathbb{Z}}$ and $\beta \circ \beta^{-1} = \mathrm{id}_{\mathbb{N}}$.

Solution

By calculation: take all possible cases and show that β and β^{-1} do the right thing. For example, for $n \geq 0$ we have

$$(\beta^{-1} \circ \beta)(n) \stackrel{\text{def}}{=} \beta^{-1}(\beta(n)) = \beta^{-1}(2n) = n$$

by the definitions of β and β^{-1} respectively. Similarly, for $n < 0$ we have

$$(\beta^{-1} \circ \beta)(n) \stackrel{\text{def}}{=} \beta^{-1}(\beta(n)) = \beta^{-1}(-2n - 1) = -\frac{-2n - 1 + 1}{2} = -\frac{-2n}{2} = n$$

also by the definitions of β and β^{-1}. These two cases show that $\beta^{-1} \circ \beta = \mathrm{id}_{\mathbb{N}}$.

Conversely, for n even we have

$$(\beta \circ \beta^{-1})(n) \stackrel{\text{def}}{=} \beta(\beta^{-1}(n)) = \beta(n/2) = 2(n/2) = n$$

whereas for n odd we have

$$(\beta \circ \beta^{-1})(n) \stackrel{\text{def}}{=} \beta(\beta^{-1}(n)) = \beta\left(-\frac{n+1}{2}\right) = (-2)\left(-\frac{n+1}{2}\right) - 1 = n + 1 - 1 = n$$

by the definitions of β and β^{-1} respectively. These two cases show that $\beta \circ \beta^{-1} = \mathrm{id}_{\mathbb{Z}}$.

** 3. Argue that there cannot be a bijection $\mathbb{B} \xrightarrow{\cong} \mathbb{N}$.

Solution

A function $f : \mathbb{B} \to \mathbb{N}$ can never be surjective. Suppose $f(\bot) = n_0$ and $f(\top) = n_1$. Then for any n other than n_0 and n_1 there cannot be a $b \in \mathbb{B}$ such that $f(b) = n$.

Alternatively, suppose $f^{-1} : \mathbb{N} \to \mathbb{B}$ is an inverse to f. Then construct the elements

$$f^{-1}(0), f^{-1}(1), f^{-1}(2), \ldots \in \mathbb{B}$$

All of these are elements of $\mathbb{B}$, of which there are only 2 ($\bot$ and $\top$). Thus, by the pigeonhole principle, it must be that two elements of that list are the same, i.e. that $f^{-1}(i) = f^{-1}(j)$ for some $i, j \in \mathbb{N}$ with $i \neq j$. Thus f^{-1} cannot be injective. We can only conclude that there cannot be an inverse to f.

** 4. Construct a bijection $\phi_3 : \mathbb{N} \times \mathbb{N} \times \mathbb{N} \xrightarrow{\cong} \mathbb{N}$, and prove that it is a bijection.
[Hint: use the pairing function twice.]

Solution

We may construct a bijection by using the pairing function twice:

$$\begin{aligned} &\phi_3 : \mathbb{N} \times \mathbb{N} \times \mathbb{N} \to \mathbb{N} \\ &\phi_3(n_1, n_2, n_3) \stackrel{\text{def}}{=} \langle n_1, \langle n_2, n_3 \rangle \rangle \end{aligned}$$

The inverse is defined by unpacking the number twice:

$$\begin{aligned} &\phi_3^{-1} : \mathbb{N} \to \mathbb{N} \times \mathbb{N} \times \mathbb{N} \\ &\phi_3^{-1}(n) \stackrel{\text{def}}{=} (n_1', n_2', n_3') \text{ where } (n_1', x) \stackrel{\text{def}}{=} \mathsf{split}(n) \text{ and } (n_2', n_3') \stackrel{\text{def}}{=} \mathsf{split}(x) \end{aligned}$$

It is then possible to calculate that $\phi_3^{-1} \circ \phi = \mathrm{id}_{\mathbb{N} \times \mathbb{N} \times \mathbb{N}}$.

Given $(n_1, n_2, n_3) \in \mathbb{N} \times \mathbb{N} \times \mathbb{N}$ let

$$y \stackrel{\text{def}}{=} \phi_3(n_1, n_2, n_3) \stackrel{\text{def}}{=} \langle n_1, \langle n_2, n_3 \rangle \rangle$$

Calculate $\phi_3^{-1}(y) \stackrel{\text{def}}{=} (n_1', n_2', n_3')$ by first calculating $(n_1', x) \stackrel{\text{def}}{=} \mathsf{split}(y)$ and then $(n_2', n_3') \stackrel{\text{def}}{=} \mathsf{split}(x - 1)$.

First, we have

$$(n_1', x) \stackrel{\text{def}}{=} \mathsf{split}(y) = \mathsf{split}(\langle n_1, \langle n_2, n_3 \rangle \rangle) = (n_1, \langle n_2, n_3 \rangle)$$

by definition of y and then using the fact that $\langle -, - \rangle$ and $\mathsf{split}(-)$ are inverses. Thus $n_1' = n_1$.

Then, we calculate that

$$(n_2', n_3') \stackrel{\text{def}}{=} \mathsf{split}(x) = \mathsf{split}(\langle n_2, n_3 \rangle) = (n_2, n_3)$$

by definition and the fact $\langle -, - \rangle$ and $\mathsf{split}(-)$ are inverses. Thus $n_2' = n_2$ and $n_3' = n_3$.

In summary we have

$$\phi_3^{-1}(\phi_3(n_1, n_2, n_3)) = \phi_3^{-1}(y) = (n_1', n_2', n_3') = (n_1, n_2, n_3)$$

Thus $\phi_3^{-1} \circ \phi_3 = \mathrm{id}_{\mathbb{N} \times \mathbb{N} \times \mathbb{N}}$. A similar calculation shows that $\phi_3 \circ \phi_3^{-1} = \mathrm{id}_{\mathbb{N}}$.

** 5. Prove that if $f : A \xrightarrow{\cong} B$ is a bijection, then so is its inverse $f^{-1} : B \to A$.

Solution

There are two ways to prove this fact. The quick one is use the characterisation of a bijection as a function that has an inverse. It suffices to notice that if the inverse of f is f^{-1}, then f^{-1} is also the inverse of f: the definition of inverses is *self-dual*.

The longer way is to prove in detail that f^{-1} is an injection and a surjection. It is an injection because $f^{-1}(b_1) = f^{-1}(b_2)$ implies that $f(f^{-1}(b_1)) = f(f^{-1}(b_2))$, which by one of the defining equation of inverses implies that $b_1 = b_2$. It is a surjection because the equation $f^{-1}(f(a)) = a$ for all $a \in A$ implies that each $a \in A$ always has a preimage along f^{-1}, namely $f(a)$.

** 6. Let $f : A \to B$ and $g : B \to C$ be functions.

(a) Prove that if f and g are injections, then so is $g \circ f : A \to C$.

(b) Prove that if f and g are surjections, then so is $g \circ f : A \to C$.

(c) Prove that if $f : A \to B$ and $g : B \to C$ are bijections then so is $g \circ f : A \to C$.

Solution

(a) Suppose $(g \circ f)(a_1) = (g \circ f)(a_2)$. Unfolding the definition of $g \circ f$, this means that $g(f(a_1)) = g(f(a_2))$. As g is an injection, this implies that $f(a_1) = f(a_2)$. Given that f is an injection, this in turn implies that $a_1 = a_2$, which is what we wanted to prove.

(b) Let $c \in C$. As g is a surjection, we can find a $b \in B$ such that $g(b) = c$. Then, as f is a surjection, we can find a $a \in A$ such that $f(a) = b$. Then $(g \circ f)(a) = g(f(x)) = g(b) = c$.

(c) A bijection is a function that is injective and surjective. f and g are injective, so by the first part of the question so is $g \circ f$. Similarly for surjectivity. In conclusion, $g \circ f$ is a bijection.

** 7. Prove that if $f : A \to B$ is an isomorphism and $g : B \to C$ is an isomorphism then so is $g \circ f : A \to C$. [Hint: construct an inverse. It is possible to show this in a point-free style using the fact function composition is associative, i.e. $h \circ (g \circ f) = (h \circ g) \circ f$, and that the identity function is a unit for it, i.e. $\mathrm{id}_B \circ f = f = f \circ \mathrm{id}_A$.]

Solution

Suppose $f : A \to B$ has an inverse $f^{-1} : B \to A$ and $g : B \to C$ has an inverse $g^{-1} : C \to B$. Then we can show that $f^{-1} \circ g^{-1} : C \to A$ is an inverse $g \circ f : A \to C$. Moreover, we can do this in a point-free style:

$$(f^{-1} \circ g^{-1}) \circ (g \circ f) = f^{-1} \circ (g^{-1} \circ g) \circ f = f^{-1} \circ \mathrm{id}_{\mathbb{B}} \circ f = f^{-1} \circ f = \mathrm{id}_A$$

Similarly, $(g \circ f) \circ (f^{-1} \circ g^{-1}) = \mathrm{id}_C$.

*** 8. We define the set $\mathscr{T}$ of *binary trees* by the Backus-Naur form

$$t \in \mathscr{T} ::= \bullet \mid \mathsf{fork}(t_1, n, t_2)$$

where $n \in \mathbb{N}$ is a natural number. This is an inductive definition: a tree is either empty ($\bullet$), or is a fork, consisting of a left subtree t_1, a number $n \in \mathbb{N}$, and a right subtree t_2.

Construct a bijection $\mathscr{T} \xrightarrow{\cong} \mathbb{N}$.

[Hint: look at the way lists—also an inductively defined set!—are encoded as natural numbers in the reference material. Try to copy that. Also, use ϕ_3 from the previous exercise.]

Solution

A bijection is given by

$$\begin{aligned}
&\phi_T : \mathcal{N} \to \mathbb{T} \\
&\phi_T(\bullet) \stackrel{\text{def}}{=} 0 \\
&\phi_T(\mathsf{fork}(t_1, n, t_2)) \stackrel{\text{def}}{=} 1 + \langle n, \langle \phi_T(t_1), \phi_T(t_2) \rangle \rangle
\end{aligned}$$

Its inverse is given by

$$\begin{aligned}
&\phi_T^{-1} : \mathbb{N} \to \mathcal{T} \\
&\phi_T^{-1}(x) \stackrel{\text{def}}{=} \begin{cases} \bullet & \text{if } x = 0 \\ \mathsf{fork}(\phi_T^{-1}(n_1), n, \phi_T^{-1}(n_2)) & \text{if } x > 0, \text{where } (n, ns) \stackrel{\text{def}}{=} \mathsf{split}\,(x-1) \text{ and } (n_1, n_2) \stackrel{\text{def}}{=} \mathsf{split}\,(ns) \end{cases}
\end{aligned}$$

You need not prove that this is the inverse; the question did not ask you for that.

*** 9. Given a bijection $f : A \xrightarrow{\cong} \mathbb{N}$ and a bijection $g : B \xrightarrow{\cong} \mathbb{N}$, show how to construct a bijection $A \times B \xrightarrow{\cong} \mathbb{N}$. Prove that it is a bijection.

Solution

Define the function $h : A \times B \to \mathbb{N}$ by

$$h(a, b) \stackrel{\text{def}}{=} \langle f(a), g(b) \rangle$$

Then, its inverse $h^{-1} : \mathbb{N} \to A \times B$ is given by

$$h^{-1}(n) \stackrel{\text{def}}{=} (f^{-1}(n_1), g^{-1}(n_2)) \quad \text{where } (n_1, n_2) \stackrel{\text{def}}{=} \mathsf{split}\,(n)$$

A calculation like the ones given previously shows that these are inverses.

[BONUS] Here is a fun way to obtain the same result. Given any functions $h : A \to C$ and $k : B \to D$, define a function $h \times k : A \times B \to C \times D$ by

$$(h \times k)(a, c) \stackrel{\text{def}}{=} (h(a), k(c))$$

First prove that if h and k are bijections then so is $h \times k$. Hence we immediately obtain a bijection

$$f \times g : A \times B \xrightarrow{\cong} \mathbb{N} \times \mathbb{N}$$

We may then compose this with the pairing function. By previous exercises, the composition of two bijections is a bijection!

**** 10. Prove that bijections and isomorphisms are the same thing.

(a) (Easier.) Prove that every isomorphism is a bijection.

(b) (Harder.) Prove that every bijection is an isomorphism. [Hint: consider the preimage

> $f^{-1}(\{b\})$ of a bijection $f : A \to B$ at every possible $b \in B$. What does it look like?]

Solution

(a) First we prove that $f : A \to B$ is surjective. Given any particular $b \in B$ the equation $f \circ f^{-1} = \mathrm{id}_B$ gives us $f(f^{-1}(b)) = b$, so $f^{-1}(b) \in A$ is a preimage of f at $b \in B$.
Then, we prove that $f : A \to B$ is injective. Suppose $a_1, a_2 \in A$ have the property that $f(a_1) = f(a_2)$. Applying the inverse $f^{-1} : B \to A$ to both sides we have

$$a_1 = f^{-1}(f(a_1)) = f^{-1}(f(a_2)) = a_2$$

where we have used the fact $f^{-1} \circ f = \mathrm{id}_A$ in the first and last equations.

(b) Suppose $f : A \to B$ is a bijection. Given any $b \in B$, consider the preimage of f at b, i.e.

$$f^{-1}(\{b\}) \stackrel{\text{def}}{=} \{x \in A \mid f(x) = b\}$$

This is the set of all solutions of the equation $f(x) = b$ for $x \in A$.
First, notice that for any $b \in B$ the preimage $f^{-1}(\{b\})$ is non-empty. As f is a surjective function, we have that for every $b \in B$ there exists some $a_b \in A$ with $f(a_b) = b$. Thus this a_b is in the preimage $f^{-1}(\{b\})$.
Second, notice that for any $b \in B$ there is at most one element in the preimage $f^{-1}(\{b\})$. Suppose that $a_1, a_2 \in f^{-1}(\{b\})$. Then we have that $f(a_1) = b$ and $f(a_2) = b$, so $f(a_1) = f(a_2)$. As f is injective, it must be that $a_1 = a_2$.
We have thus shown that if f is a bijection then $f^{-1}(\{b\}) = \{a_b\}$ for a unique $a_b \in A$. We thus define the inverse by

$$\begin{aligned} f^{-1} &: B \to A \\ f^{-1}(b) &\stackrel{\text{def}}{=} a_b \end{aligned}$$

where the choice of a_b is now unique.
We must not forget to show that f^{-1} is an inverse! We clearly have $f(f^{-1}(b)) = f(a_b) = b$, hence $f \circ f^{-1} = \mathrm{id}_B$. Moreover, $f^{-1}(f(a)) = a$, as a is mapped to $f(a)$ by f, so it must be the unique preimage $a_{f(a)} \in f^{-1}(\{f(a)\})$ of f at $f(a) \in B$.

** 11. Is the predicate

> $$\mathsf{LUCKY}_{127} = \{ \ulcorner S \urcorner \mid \text{ running } S \text{ on input 1 runs for at least 127 computational steps } \}$$
>
> decidable? [Hint: if it is, describe a program that decides it. Think simply, write informally, and do not let the syntactic poverty of While confine you.]

Solution

It is decidable. It is decided by a program which, on input $\ulcorner S \urcorner$,

- Simulates a run of S on input 1.
- Counts the first 127 steps of that simulation.

If the simulation doesn't halt in a state of the form $\langle \texttt{skip}, \sigma \rangle$ before 127 steps, it returns true. Otherwise it returns false.

** 12. Prove that the set

$$\mathsf{Zero} = \{ \ulcorner S \urcorner \mid [\![\mathtt{S}]\!]_{\mathtt{x}}(0) \downarrow \}$$

is semi-decidable. [Hint: As above, think simply, write informally, and do not let the syntactic poverty of While confine you.]

Solution

The set Zero is semi-decided by a program which performs the following actions: on input m,

- Decode $m = \ulcorner S \urcorner$.
- Simulate the running of S on input 0.
- If and when that terminates, check if the memory is the form $[\mathtt{x} \mapsto m']$ for some $m' \in \mathbb{N}$. If it is, return 1. Otherwise, go into an infinite loop.

If $[\![\mathtt{S}]\!]_{\mathtt{x}}(0) \downarrow$ then the above simulation will terminate at some point, and our program will correctly return 1.

But if $[\![\mathtt{S}]\!]_{\mathtt{x}}(0) \uparrow$ then the simulation will either run forever or terminate in a 'rubbish' state (i.e. one with variables other than x set to a non-zero value). In the first case our program runs forever. In the second case, instruction 3 above forces our program to also run forever. So in either case our program runs forever.

Thus Zero is semi-decidable.

*** 13. Prove that if the predicates U and V are semi-decidable, then so is $U \cup V$. [Hint: use simulations.]

Solution

This was a trick question from previous week's sheets, which you now have the tools to solve.

Suppose we have a program A that computes the semi-characteristic function of U, and a program B that computes the semi-characteristic function of V. We want to build a program that computes the semi-characteristic function of $U \cup V$.

On input m,

- Set up a simulation of A on m, and of B on m.
- Alternate between running the first simulation for a finite number of computational steps (say, 42), and then running the second simulation for a finite number of steps.
- If either of the simulations ever halts and outputs 1, do the same.
- If both simulations halt in a 'rubbish' state, go into an infinite loop.

This program semi-decides $U \cup V$. If $m \in U$, then at some point the simulation of A on m will halt and output 1, and so will our program. Otherwise it will either halt in a 'rubbish' state, or run forever. Similarly if $m \in V$. But if m is in neither, then both simulations will either halt in a 'rubbish' state, or run forever. In the first case we loop forever, and in the second we are forced to keep simulating forever.