Übung zur Vorlesung

Automatisierte Logik und Programmierung

Prof. Chr. Kreitz

Universität Potsdam, Theoretische Informatik — Wintersemester 2013/14

Blatt 1 — Abgabetermin: 25.10.2013

Das erste Übungsblatt soll dazu dienen, Erfahrungen mit der Entwicklung von Evidenztermen zu sammeln. Es ist in gewissen Grenzen wie das Programmieren einfachster Algorithmen, bei denen nur die Datentypen als Rahmenbedingungen vorgegeben sind. Ansonsten haben Sie völlige Freiheit.

Wir werden mögliche Lösungen zu Beginn der Veranstaltung am 25.10.2013 besprechen

---

Aufgabe 1.1 (Evidenz)

Bestimmen Sie mögliche Evidenzterme für die folgenden aussagenlogischen Formeln. Wenn es nicht möglich ist, Evidenz zu konstruieren, geben Sie eine kurze Begründung oder ein Gegenbeispiel für die Gültigkeit der Formel an.

1.1–a $(P \Rightarrow Q) \Rightarrow (\neg Q \Rightarrow \neg P)$:

1.1–b $(P \Rightarrow Q) \Rightarrow (\neg P \Rightarrow \neg Q)$:

1.1–c $\neg(P \vee Q) \Rightarrow (\neg P \wedge \neg Q)$:

1.1–d $(\neg P \wedge \neg Q) \Rightarrow \neg(P \vee Q)$:

1.1–e $\neg(\neg P \wedge \neg Q) \Rightarrow (P \vee Q)$:

1.1–f $\neg\neg P \Rightarrow P$:

1.1–g $\neg P \vee P$:

1.1–h $\neg(P \vee \neg P)$:

1.1–i $\neg\neg(P \vee \neg P)$:

1.1–j $(P \vee (Q \wedge R)) \Rightarrow (P \vee Q) \wedge (P \vee R)$:

Lösung 1.1

1.1–a $(P \Rightarrow Q) \Rightarrow (\neg Q \Rightarrow \neg P)$:

The evidence is $\lambda f.\,(\lambda g.\,(\lambda p.\,g(f(p))))$

1.1–b $(P \Rightarrow Q) \Rightarrow (\neg P \Rightarrow \neg Q)$:

There cannot be any evidence. A possible counterexample is Q being `0=0` and P being `0=1`

Again there is an alternative argument.
Any evidence for this formula must have the form $\lambda f.\lambda g.\,(\lambda q.\,x)$ for some $x : \{\}$, where q is a variable of type $[Q]$. The only way to construct an element of $\{\}$ would be using g but for that we need an element of $[P]$ as input. There is no way to construct that element.

1.1–c $\neg(P \vee Q) \Rightarrow (\neg P \wedge \neg Q)$:

The evidence is $\lambda f.\,(\lambda p.\,f(\mathtt{inl}(p)),\ \lambda q.\,f(\mathtt{inr}(q)))$

1.1–d $(\neg P \wedge \neg Q) \Rightarrow \neg(P \vee Q)$:

The evidence is $\lambda x.\,(\lambda y.\,((\text{case } y \text{ of inl}(p) \rightarrow x_1(p) \mid \text{inr}(q) \rightarrow x_2(q))))$

1.1–e $\neg(\neg P \wedge \neg Q) \Rightarrow (P \vee Q)$:

Any evidence for this formula must have the form $\lambda f.\,\mathtt{inl}(p)$ for some $p : [P]$ or $\lambda f.\,\mathtt{inr}(q)$ for some $q : [Q]$ where f is a function in $([\neg P]\times[\neg Q])\rightarrow\{\}$. There is no way to construct either p or q from that function, since f can only construct elements of $\{\}$.

1.1–f $\neg\neg P \Rightarrow P$:

Any evidence for this formula must have the form $\lambda f.\,p$ for some $p : [P]$ where f is a function in $[\neg P]\rightarrow\{\}$. There is no way to construct p.

1.1–g $\neg P \vee P$:

Any evidence for this formula must have the form $\mathtt{inl}(\lambda p.\,x)$ for some $x : \{\}$ or $\mathtt{inr}(p)$ for some $p : [P]$. We have nothing in our hands to construct either of the two.

1.1–h $\neg(P \vee \neg P)$:

Any evidence for this formula must have the form $\lambda x.\,z_x$ where $z_h : \{\}$ whenever $x : [P] + ([P]\rightarrow\{\})$. x can be either $\mathtt{inl}(p)$ for some $p : [P]$ or $\mathtt{inr}(h)$ for some $h : [P]\rightarrow\{\}$. In both cases we have nothing in our hands to construct an element of $\{\}$.

1.1–i $\neg\neg(P \vee \neg P)$:

Evidence for this formula must have the form $\lambda h.\,z_h$ where $z_h : \{\}$ whenever $h : ([P] + ([P]\rightarrow\{\}))\rightarrow\{\}$. At a first glance it seems almost impossible to construct z_h solely out of h. But it is actually possible to construct a valid input $x : [P] + ([P]\rightarrow\{\})$ for the function h.

x can be either $\mathtt{inl}(p_0)$ for some $p_0 : [P]$ or $\mathtt{inr}(g)$ for some $g : [P]\rightarrow\{\}$. We can't construct p_0 from scratch. To construct g we need to take an input $p : [P]$ and construct an element $x_p : \{\}$. The only known way to do that is to apply h to $\mathtt{inl}(p)$. Thus $g = \lambda p.h(\mathtt{inl}(p)$ and $z_h = h(\mathtt{inr}(g) = h(\mathtt{inr}(\lambda p.h(\mathtt{inl}(p))$

The evidence is $\lambda h.\,h(\mathtt{inr}(\lambda p.h(\mathtt{inl}(p))$

1.1–j $(P \vee (Q \wedge R)) \Rightarrow (P \vee Q) \wedge (P \vee R)$:

The evidence is $\lambda x.(\mathtt{case}\ x\ \mathtt{of}\ \ \mathtt{inl}(p) \rightarrow (\mathtt{inl}(p), \mathtt{inl}(p))\ \mid\ \mathtt{inr}(y) \rightarrow (\mathtt{inr}(y_1), \mathtt{inr}(y_2)))$