Típuselmélet

Alexy Marcell

2021. december 17.

Témavezető: Kaposi Ambrus

Típuselmélet és logika

Curry-Howard izomorfizmus
Állítások megfeleltethetők típusoknak.
Bizonyítás / tanú megfelel egy típus elemének.

$A : U$	állítások halmaza
$a : A$	az A állítás bizonyítása
$\perp$	hamis állítás
$\top$	igaz állítás
$A \uplus B$	A vagy B
$A \times B$	A és B
$A \to B$	A-ból következik B
$A \to \perp$	az A állítás tagadása
$B(x) : A \to U$	az A-n értelmezett állítások
$\Sigma_{a:A} B(a)$	$\exists x : A$ hogy $B(x)$ teljesüljön
$\Pi_{a:A} B(a)$	$\forall x : A$ teljesül $B(x)$

Egyszerűbb típusok

Üres típus: Nincs eleme, hamis állítás.

```
data ⊥ : Set where
exfalso : {C : Set} → ⊥ → C
exfalso ()
```

Egység típus: Egy eleme van, igaz állítás.

```
data ⊤ : Set where
  triv : ⊤
```

Természetes számok

Egész számok típusa.
Induktív definíció. "0" elem és a "rákövetkező" függvény.

```
data ℕ : Set where
  zero : ℕ
  suc : ℕ → ℕ

n0 : ℕ
n0 = zero
n1 : ℕ
n1 = suc n0
n2 : ℕ
n2 = suc n1
n3 : ℕ
n3 = suc n2
```

Függvények

Függvény: Az A típusú elemből B típusú elemet képez

$$\mathsf{to3} : \top \to \mathbb{N}$$
$$\mathsf{to3}\ \mathsf{triv} = \mathsf{suc}\ (\mathsf{suc}\ (\mathsf{suc}\ \mathsf{zero}))$$

Megfelel egy következtetésnek.
Tagadás: Következik a hamis állítás.

```
-- trivfalse : ⊤ → ⊥
-- trivfalse triv = {!!}
```

Direkt szorzat: A két típusból vett elempárok alkotják.

```
data _×_ (A : Set) (B : Set) : Set where
  _,_ : A → B → A × B

curry : {A B C : Set}
  → ((A × B) → C)
  → (A → B → C)
curry f = λ a → λ b → f (a , b)
```

Unió / Vagy

Diszjunkt unió: Vagy a jobboldali, vagy a baloldali típusból tartalmaz elemet.

$$
\begin{aligned}
&\textsf{data}\ \_\uplus\_\ (A : \textsf{Set})\ (B : \textsf{Set}) : \textsf{Set}\ \textsf{where} \\
&\quad \iota_1 : A \to A \uplus B \\
&\quad \iota_2 : B \to A \uplus B
\end{aligned}
$$

$$
\begin{aligned}
&\textsf{case} : \{A\ B\ C : \textsf{Set}\} \\
&\quad \to (A \to C) \to (B \to C) \to (A \uplus B) \to C \\
&\textsf{case}\ fa\ fb\ (\iota_1\ a) = fa\ a \\
&\textsf{case}\ fa\ fb\ (\iota_2\ b) = fb\ b
\end{aligned}
$$

$$
\begin{aligned}
&\textsf{applyEither} : \{A\ B\ C\ D : \textsf{Set}\} \\
&\quad \to (A \to C) \to (B \to D) \to (A \uplus B) \\
&\quad \to (C \uplus D) \\
&\textsf{applyEither}\ f\ g\ (\iota_1\ a) = \iota_1\ (f\ a) \\
&\textsf{applyEither}\ f\ g\ (\iota_2\ b) = \iota_2\ (g\ b)
\end{aligned}
$$

Kisebb vagy egyenlő

```
data _≤_ : (l : ℕ) (g : ℕ) → Set where
  z≤n : ∀ {n} → zero ≤ n
  s≤s : ∀ {n m} → n ≤ m → (suc n) ≤ (suc m)

1≤3 : n1 ≤ n3
1≤3 = s≤s z≤n

2≰1 : n2 ≤ n1 → ⊥
2≰1 (s≤s ())
3≰2 : n3 ≤ n2 → ⊥
3≰2 (s≤s x) = 2≰1 x
```

Páros - Páratlan

```
data Even : (n : ℕ) → Set where
  zeroEven : Even n0
  s2Even : ∀ {n} → Even n → Even (suc (suc n))

data Odd : (n : ℕ) → Set where
  oneOdd : Odd n1
  s2Odd : ∀ {n} → Odd n → Odd (suc (suc n))
```

Állítások

Állítás: Minden természetes szám páros vagy páratlan.

$$
\begin{aligned}
&\mathsf{evenOrOdd} : (n : \mathbb{N}) \to (\mathsf{Even}\ n) \uplus (\mathsf{Odd}\ n)\\
&\mathsf{evenOrOdd}\ \mathsf{zero} = \iota_1\ \mathsf{zeroEven}\\
&\mathsf{evenOrOdd}\ (\mathsf{suc}\ \mathsf{zero}) = \iota_2\ \mathsf{oneOdd}\\
&\mathsf{evenOrOdd}\ (\mathsf{suc}\ (\mathsf{suc}\ n)) =\\
&\quad \mathsf{applyEither}\ \mathsf{s2Even}\ \mathsf{s2Odd}\ (\mathsf{evenOrOdd}\ n)
\end{aligned}
$$

Állítás: Ha egy n páros természetes szám kisebb vagy egyenlő egy m páratlan számnál, akkor n rákövetkezőjénél is kisebb lesz.

$$
\begin{aligned}
&\mathsf{thm} : (n : \mathbb{N}) \to (m : \mathbb{N}) \to \mathsf{Even}\ n \to \mathsf{Odd}\ m \to (n \leq m)\\
&\quad \to ((\mathsf{suc}\ n) \leq m)\\
&\mathsf{thm}\ \mathsf{zero}\ (\mathsf{suc}\ m)\ \mathit{En}\ \mathit{Om}\ \mathit{ineq} = \mathsf{s{\leq}s}\ \mathsf{z{\leq}n}\\
&\mathsf{thm}\ (\mathsf{suc}\ (\mathsf{suc}\ n))\ (\mathsf{suc}\ (\mathsf{suc}\ m))\\
&\quad (\mathsf{s2Even}\ \mathit{En})\ (\mathsf{s2Odd}\ \mathit{Om})\ (\mathsf{s{\leq}s}\ (\mathsf{s{\leq}s}\ \mathit{ineq})) =\\
&\qquad \mathsf{s{\leq}s}\ (\mathsf{s{\leq}s}\ (\mathsf{thm}\ n\ m\ \mathit{En}\ \mathit{Om}\ \mathit{ineq}))
\end{aligned}
$$

Források

The Univalent Foundations Program, *Homotopy Type Theory: Univalent Foundations of Mathematics*, első fejezet.
Kaposi Ambrus, *Bevezetés a homotópia-típuselméletbe*
Agda wiki (https://wiki.portal.chalmers.se/agda/pmwiki.php?n=Main.HomePage)
Diviánszky Péter, *Agda Tutorial*
Jan Malakhovski, *Brutal [Meta]Introduction to Dependent Types in Agda*

Vége

Köszönöm a figyelmet.