

Mi a típuselmélet?

Informatikai Logika munkacsoport

Diszciplináris Informatika és Számítástudomány műhely

A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg
(EFOP-3.6.3-VEKOP-16-2017-00002)

BIT-CON szakmai nap
2018. május 3.

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Motiváció

Mi a típuselmélet?

A munkacsoportról

Mellékletek

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Motiváció

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Mi a típuselmélet? (röviden)

A típuselmélet egy nyelv programozásra és bizonyításra.

Konkrét implementációi: Agda, Coq, ...

Agda IDE szolgáltatások:

- ▶ unikód karakterbevitel $\rightarrow \mapsto \mathcal{U} \ \mathbb{N} \ 1 \ 2 \ \perp \ \cup \ \times \ \equiv \ \leq \ \geq \ \dots$
- ▶ típusellenőrzés (bizonyítás ellenőrzés)
- ▶ automatikus és interaktív kód kiegészítés
- ▶ kódgenerálás (javascript, Haskell)

The screenshot shows the Emacs editor window titled 'emacs@grothendieck'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Minibuf', and 'Help'. The toolbar contains icons for opening files, saving, undo, and other standard editor functions. The code in the editor is as follows:

```

1
2 data N : Set where
3   Z : N
4   S : N → N
5
6   +_ : N → N → N
7   Z + m = m
8   S n + m = S (n + m)

```

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Mi a formális bizonyítás?

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei
Az alap típuselmélet kibővítése
Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások
Milyen formális nyelvet
használjunk?
Típuselméleti definíciók
Logika a típuselméletben
Megjegyzések
Egyéb

Tesztelés vs. bizonyítás

	matematikai példa	informatikai példa
állítás	derékszögű háromszögben $a^2+b^2=c^2$	a program nem ad ki érzékeny adatot
tesztelés	látjuk hogy konkrét háromszögekre igaz	látjuk hogy konkrét esetekben igaz
bizonyítás	látjuk hogy minden háromszögre igaz	látjuk hogy minden esetben igaz

Teszteléssel a hibák jelenléte, bizonyítással a hibák hiánya igazolható.

A matematika 2.300 éve áttért a bizonyításra.

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei
Az alap típuselmélet kibővítése
Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei
Az alap típuselmélet kibővítése
Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások
Milyen formális nyelvet
használjunk?
Típuselméleti definíciók
Logika a típuselméletben
Megjegyzések
Egyéb

“derékszögű háromszögben $a^2+b^2=c^2$ ” formálisabb állítás, mint
“a program nem ad ki érzékeny adatot”

- ▶ cél a formalizálás fokában:
géppel ellenőrizhető bizonyítások
- ▶ formális bizonyítás \equiv géppel ellenőrizhető bizonyítás
- ▶ szükséges egy nyelv, amin leírjuk az állítást és a bizonyítást
- ▶ ennek a nyelvnek le kell tudni írnia a matematikát

Mik készültek el típuselmélettel?

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

**Mik készültek el
típuselmélettel?**

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Top 100 matematikai állítás formalizálása¹

A matematikai bizonyítások formalizálása része a programhelyesség bizonyításnak.

rendszer	mire épül	formális bizonyítások
HOL Light	típuselmélet	86/100
Isabelle	típuselmélet	78/100
Coq	típuselmélet	69/100
Mizar	klasszikus logika	69/100
Metamath	szimbólum helyettesítés	67/100
ProofPower	típuselmélet	43/100

Konkrét példák formális bizonyításokra a mellékletben található.

¹<http://www.cs.ru.nl/~freek/100/>

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

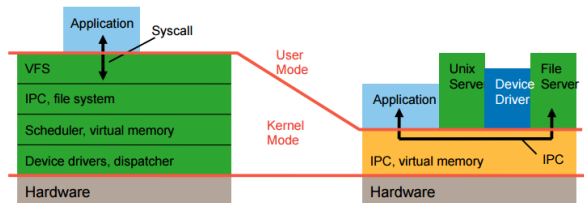
Logika a típuselméletben

Megjegyzések

Egyéb

seL4 mikrokernel specifikációja

mikrokernel: minimalizált védett módban futó szoftver



seL4: nagy hatékonyságú mikrokernel <10.000 sor C kóddal

- ▶ az seL4 megfelel az 5700 soros Haskell implementációnak²
- ▶ további formális bizonyítások az seL4-ről³
 - ▶ a válaszdő analízise
 - ▶ az elérési jogok betartása
 - ▶ érzékeny információ érzékeny is marad

²seL4: Formal Verification of an OS Kernel, 2009. (13 szerző)

³Comprehensive formal verification of an OS microkernel, 2014. (7 szerző)

Motiváció

Mi a formális bizonyítás?

Mik készültek el típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált bizonyítások

Milyen formális nyelvet használunk?

Típuselméleti definíciók

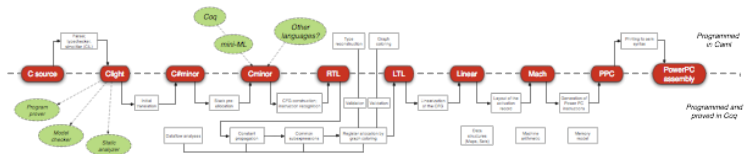
Logika a típuselméletben

Megjegyzések

Egyéb

Bizonyítottan helyes fordítás

- ▶ a C nyelv formális specifikációja⁴
- ▶ a SPARCV8 processzor utasításkészletének formális specifikációja⁵
- ▶ bizonyítottan helyes fordító C-ről optimalizált PowerPC, ARM, RISC-V és x86 kódra⁶



⁴The C standard formalized in Coq, PhD thesis of Robbert Jan Krebbers

⁵Formalizing SPARCV8 Instruction Set Architecture in Coq (Technical Report)

⁶<http://compcert.inria.fr/>

Motiváció

Mi a formális bizonyítás?

Mik készültek el típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált bizonyítások

Milyen formális nyelvet használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Mi a típuselmélet?

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

A típuselmélet alapelemei

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Konstruktív függvények

$a \mapsto a$:	$A \rightarrow A$
$a \mapsto b \mapsto a$:	$A \rightarrow B \rightarrow A$
$f \mapsto a \mapsto f a$:	$(A \rightarrow B) \rightarrow A \rightarrow B$
$f \mapsto a \mapsto f a a$:	$(A \rightarrow A \rightarrow B) \rightarrow A \rightarrow B$
$f \mapsto a \mapsto b \mapsto f b a$:	$(B \rightarrow A \rightarrow C) \rightarrow A \rightarrow B \rightarrow C$

Ugyanezek több zárójellel:

$a \mapsto a$:	$A \rightarrow A$
$a \mapsto (b \mapsto a)$:	$A \rightarrow (B \rightarrow A)$
$f \mapsto (a \mapsto f(a))$:	$(A \rightarrow B) \rightarrow (A \rightarrow B)$
$f \mapsto (a \mapsto (f(a))(a))$:	$(A \rightarrow (A \rightarrow B)) \rightarrow (A \rightarrow B)$
$f \mapsto (a \mapsto (b \mapsto (f(b))(a)))$:	$(B \rightarrow (A \rightarrow C)) \rightarrow (A \rightarrow (B \rightarrow C))$

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Függő típusú konstruktív függvények

Az értékészlet függhet a függvényparaméter értékétől.

Példa

$$n \mapsto (1, 2, 3, \dots, n) : (n : \mathbb{N}) \rightarrow \mathbb{N}^n$$

Csak függő típusú függvények vannak

$$n \mapsto (0, n+1, n) : (n : \mathbb{N}) \rightarrow \mathbb{N}^3$$

rövidítés:

$$n \mapsto (0, n+1, n) : \mathbb{N} \rightarrow \mathbb{N}^3$$

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

A normál forma fogalma

Tanár: Mennyi $7*9$?

Diák: $7*9$ az $9*7$.

Tanár: Nem, $7*9$ nem $9*7$.
 $7*9$ az 63 .

A beszélgetés formálissá tehető:

- ▶ Mennyi X ? \equiv Mi az X normál formája?
- ▶ X az Y . \equiv Az X normál formája Y .

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei
Az alap típuselmélet kibővítése
Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Normál forma a típuselméletben

Kifejezés átírási lépés (példák)

$$(x \mapsto x) y \Rightarrow y$$

$$(f \mapsto x \mapsto f (f x)) S 0 \Rightarrow (x \mapsto S (S x)) 0 \Rightarrow S (S 0)$$

Normál forma (példák)

$$(x \mapsto x) y \Downarrow y$$

$$(f \mapsto x \mapsto f (f x)) S 0 \Downarrow S (S 0)$$

$$1 + 1 \Downarrow 2$$

$$2 \Downarrow 2$$

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

A típuselmélet szintaxisa

A típuselmélet egy programozási nyelv.

azonosítók

$v ::= a \mid b \mid c \mid \dots \mid 0 \mid 1 \mid \dots \mid \mathbb{N} \mid \perp \mid _x_ \mid (_, _) \mid \dots$

kifejezések

$e, e' ::= v \mid v \mapsto e \mid e e' \mid (v: e) \rightarrow e' \mid \mathcal{U}$

deklarációk / ítéletek

$l ::= e : e' \mid e \Rightarrow e' \mid e \Downarrow e'$

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Az alap típuselmélet kibővítése

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Példa: \mathbb{N} , 0 és S definíciója

data

$\mathbb{N} : \mathcal{U}$ -- \mathcal{U} a típus univerzum

$0 : \mathbb{N}$

$S : \mathbb{N} \rightarrow \mathbb{N}$ -- szám rákövetkezője, pl. S (S 0) ugyanaz mint 2

A **data** egy nyelvi fűszer adatszerkezetek definiálására.

A **data** ellenőrzi a bevezetett adatszerkezet létezését.

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

További létező adatszerkezetek

data

$\perp : \mathcal{U}$ -- \perp létezik, nincs eleme

data

$_ \times _ : \mathcal{U} \rightarrow \mathcal{U} \rightarrow \mathcal{U}$ -- szorzat
 $(_, _) : A \rightarrow B \rightarrow A \times B$

data

$_ \uplus _ : \mathcal{U} \rightarrow \mathcal{U} \rightarrow \mathcal{U}$ -- diszjunkt unió
 $\text{inj}_1 : A \rightarrow A \uplus B$
 $\text{inj}_2 : B \rightarrow A \uplus B$

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Példa: az összeadás definíciója \mathbb{N} -en

rewrite

$$_ + _ : \mathbb{N} \rightarrow \mathbb{N} \rightarrow \mathbb{N}$$

$$0 + m \Rightarrow m$$

$$S n + m \Rightarrow S (n + m)$$

$$1 + 1 \Downarrow 2, \text{ mert } S 0 + S 0 \Rightarrow S (0 + S 0) \Rightarrow S (S 0)$$

A **rewrite** egy nyelvi fűszer átírási lépések megadására.

A **rewrite** ellenőrzi az átírási lépések helyességét (típushelyesség, totalitás, terminálás).

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

További létező függvények

rewrite

$\text{fst} : A \times B \rightarrow A$

$\text{fst} (a, b) \Rightarrow a$

rewrite

$\text{case} : (A \rightarrow C) \rightarrow (B \rightarrow C) \rightarrow A \uplus B \rightarrow C$

$\text{case } f \ g \ (\text{inj}_1 \ a) \Rightarrow f \ a$

$\text{case } f \ g \ (\text{inj}_2 \ b) \Rightarrow g \ b$

rewrite

$\text{absurd} : \perp \rightarrow A$ -- átírási lépés nélkül totális

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Programozás típuselmélettel

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Mellékhatásos programok leírása (példa)

data -- konzolos alkalmazásokat leíró adatszerkezet

$IO : \mathcal{U} \rightarrow \mathcal{U}$

`getChar` : IO Char

`putChar` : Char \rightarrow IO Unit

`return` : A \rightarrow IO A

`_>>=_` : IO A \rightarrow (A \rightarrow IO B) \rightarrow IO B

rewrite -- konkrét alkalmazás: egy beolvasott karakter kiírása

`main` : IO Unit

`main` \Rightarrow `getChar >>= putChar`

rewrite -- ugyanez az alkalmazás nyelvi fűszerekkel megadva

`main` \Rightarrow `do`

`c` \leftarrow `getChar`

`putChar c`

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Bizonyítottan helyes programok írása

1. a függvények elő- és utófeltételei beépíthetők a függvény típusába
2. a típusellenőrzés kizárja a hibás implementációt
3. a kódgenerálás során kitörölhető a bizonyítással kapcsolatos kód és adat

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Előfeltétel beépítése a függvény típusába

data

$$_ \leq _ : \mathbb{N} \rightarrow \mathbb{N} \rightarrow \mathcal{U}$$

$$\text{le0} : (n : \mathbb{N}) \rightarrow 0 \leq n$$

$$\text{leS} : (n : \mathbb{N}) \rightarrow (m : \mathbb{N}) \rightarrow n \leq m \rightarrow S n \leq S m$$

Akkor van $n \leq m$ típusú érték, ha n kisebb vagy egyenlő mint m .

- ▶ van $0 \leq S 0$ típusú érték, ez az $\text{le0} (S 0)$
- ▶ nincs $S 0 \leq 0$ típusú érték

Bármilyen állítás leírható típusokkal (lásd melléklet).

Előfeltétel beépítése típusba:

rewrite

$$f : (n : \mathbb{N}) \rightarrow (m : \mathbb{N}) \rightarrow (e : n \leq m) \rightarrow \mathbb{N}$$

...

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

A munkacsoportról

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Cél: a tökéletes típuselmélet kidolgozása és implementálása

Ehhez:

- ▶ jelenlegi típuselmélet implementációkban (Agda) formalizálunk állításokat és írunk programokat
- ▶ típuselmélet fordítóprogramokat írunk Haskellben, és ezekkel kísérletezünk
- ▶ a típuselméletet írjuk le a típuselméletben
Ez egy nagyon erős eszközt ad, “ha ezt meg tudjuk csinálni, akkor mindent meg tudunk csinálni”.

Visszajelzés adható az előadásról:

- ▶ [nyilvános Google fórum](#)

Bármilyen kapcsolódó kérdésre szívesen válaszolunk:

- ▶ [Kaposi Ambrus](#), Déli tömb 2.705
- ▶ [Diviánszky Péter](#), Déli tömb 2.606
- ▶ [Kovács András](#), Déli tömb 2.620

ELTE IK, Programozási Nyelvek és Fordítóprogramok tanszék

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Mellékletek

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Típuselmélettel formalizált bizonyítások

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

**Típuselmélettel formalizált
bizonyítások**

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

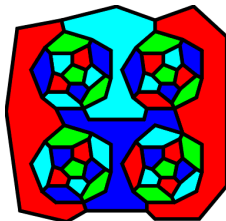
Logika a típuselméletben

Megjegyzések

Egyéb

Négyszíntétel bizonyítása

Minden síkba rajzolható gráf kiszínezhető négy színnel.



A formális bizonyítás a típuselméletben 60.000 sor.⁷

A formális bizonyítás klasszikus logikával több millió sor lenne.⁸

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mikészültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

⁷<https://github.com/tangentforks/FourColorTheorem>

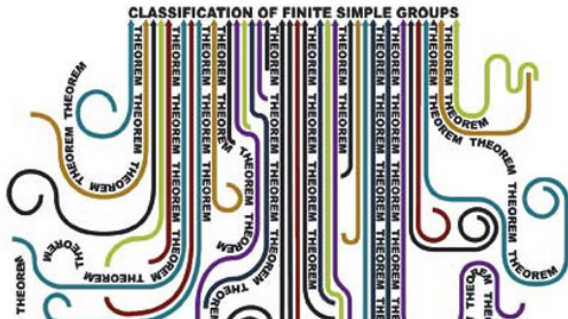
⁸An unavoidable set of D-reducible configurations, John P. Steinberger

Feit-Thompson tétel bizonyítása

Minden páratlan rendű véges csoport feloldható.

- ▶ (véges) csoportelmélet
- ▶ lineáris algebra
- ▶ Galois elmélet
- ▶ valós és komplex algebrai számok
- ▶ + 250 oldal bizonyítás (kb. 1000 oldal lett formálisan)
- ▶ + 15 matematikus, aki ezeket 150.000 sorban formalizálta

Forrás: [A Machine-Checked Proof of the Odd Order Theorem, 2013.](#)



Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Kepler tétel bizonyítása

Gömbökkel a legjobb térkitöltés hexagonális és lapközepes kockaráccsal érhető el.⁹



A jelenleg létező leghosszabb formális bizonyítás.
A formális bizonyítás klasszikus logikával túl hosszú lenne.

⁹ [A Formal Proof of the Kepler Conjecture, 2017. \(27 szerző\)](#)

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Milyen formális nyelvet használjunk?

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

**Milyen formális nyelvet
használjunk?**

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Kellenek absztrakt interfészek

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Boolean értékek definíciója

- ▶ $\text{False} = \emptyset$
- ▶ $\text{True} = \{\emptyset\}$
- ▶ $\text{Bool} = \{\text{False}, \text{True}\} = \{\emptyset, \{\emptyset\}\}$

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

**Milyen formális nyelvet
használunk?**

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Természetes számok definíciója

- ▶ $0 = \emptyset$
- ▶ $S = a \mapsto a \cup \{a\}$
- ▶ $1 \equiv S(0), 2 \equiv S(1), 3 \equiv S(2), \dots$
- ▶ $\mathbb{N} = \{0, 1, 2, \dots\} = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \dots\}$

Probléma

$\text{False} \in \mathbb{N}$ — False is egy természetes szám

Tudni kell számolni bizonyítás közben

$1 + 2 * 2 = 5$ formális bizonyítása (vázlat):

A) $2 * 2 = 4$ ezért $1 + 2 * 2 = 1 + 4$
másképp $1 + 4 = 5$

tehát az (=) tranzitivitása miatt $1 + 2 * 2 = 5$

B) rögtön látszik, mert mindkét oldalnak ugyanaz a normál formája

A négyzintétel és a Kepler tétel formális bizonyítása a B) módszerrel készült el, az A) módszerrel esélytelen lenne.

olyan formális nyelvre van szükség, amiben biztonságosan lehet számításokat végezni (akár bizonyítás közben is)

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Mi hiányzik az informatikából?

Minden típuselmélettől különböző szoftver verifikációs módszer ≥ 2 fázisú.

⇒ Nem fonódhat össze a számítás a bizonyítással, a *hogyan* a *miért*-tel.

Hol számít ez?

- ▶ bizonyítottan helyes komputer-algebrai rendszer, fordítóprogram, operációs rendszer
- ▶ a fejlesztői környezet programozás közben váltakozva ellenőriz és segít
- ▶ legyen minden szorosan együtt a még optimálisabb kódgeneráláshoz

Mi a típuselmélet?

Informatikai Logika munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált bizonyítások

Milyen formális nyelvet használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Típuselméleti definíciók

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Típusa vs. eleme

állítás / ítélet	jólformált?	további információ, ha igen
$\text{😊} \in \mathbb{N}$	✗	
$\emptyset \in \mathbb{N}$	✓	igaz / nincs ¹⁰
$\mathbb{N} \in \mathbb{N}$	✓	hamis
$0 \in \mathbb{N}$	✓	igaz
$\text{😊} : \mathbb{N}$	✗	
$\perp : \mathbb{N}$	✗	
$\mathbb{N} : \mathbb{N}$	✗	
$0 : \mathbb{N}$	✓	nincs

- ▶ a 'típusa' reláció elemibb, a jólformáltság elég
- ▶ a 'típusa' reláció korábban észreveszi a hibákat
- ▶ típusellenőrzés \equiv $e : e'$ jólformáltságának ellenőrzése

¹⁰ha absztrakt \mathbb{N} definíciója

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Függő párok

A második elem *típusa* függ az első elem *értékétől*.

$(3, (-4,5,6)) : (n : \mathbb{N}) \times \mathbb{Z}^n$.

data

$_ \times _ : (A : \mathcal{U}) \rightarrow (A \rightarrow \mathcal{U}) \rightarrow \mathcal{U}$

$_, _ : (a : A) \rightarrow B \ a \rightarrow A \times B$

Példa: végtelen sok prímszám létezésének a bizonyítása

$n \mapsto np \mapsto (\dots^{11}, (\dots^{12}, \dots^{13}))$

$: (n : \mathbb{N}) \rightarrow n \text{ prím} \rightarrow (p : \mathbb{N}) \times ((p > n) \times p \text{ prím})$

¹¹p az n-től kisebb vagy egyenlő prímekek szorzata + 1

¹²p nagyobb, mint n, mert n is prím

¹³p prímszám

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Logika a típuselméletben

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Mi a típuselmélet? Informatikai Logika munkacsoport

- ▶ Az állításokat típusokkal modellezzük.
- ▶ Jelölje $\llbracket P \rrbracket$ a P állításnak megfelelő típust.
- ▶ $\llbracket P \rrbracket$ tetszőleges, csak teljesüljön a következő feltétel:
 $\llbracket P \rrbracket$ -nek van eleme $\Leftrightarrow P$ igaz.

Szokásos megfeleltetés

indoklás

$$\llbracket P \wedge Q \rrbracket = \llbracket P \rrbracket \times \llbracket Q \rrbracket$$

$$A \times B \text{ \textasciitilde{üres} } \Leftrightarrow A \text{ \textasciitilde{üres} } \wedge B \text{ \textasciitilde{üres} }$$

$$\llbracket P \vee Q \rrbracket = \llbracket P \rrbracket \uplus \llbracket Q \rrbracket$$

$$A \uplus B \text{ \textasciitilde{üres} } \Leftrightarrow A \text{ \textasciitilde{üres} } \vee B \text{ \textasciitilde{üres} }$$

$$\llbracket P \Rightarrow Q \rrbracket = \llbracket P \rrbracket \rightarrow \llbracket Q \rrbracket$$

$$A \rightarrow B \text{ \textasciitilde{üres} } \Leftrightarrow A \text{ \textasciitilde{üres} } \vee B \text{ \textasciitilde{üres} }$$

$$\llbracket \neg P \rrbracket = \llbracket P \rrbracket \Rightarrow \perp$$

logikai azonosság

$$\llbracket \perp \rrbracket = \perp$$

\perp -nek nincs eleme

$$\llbracket \forall a \in A: P \rrbracket = (a : A) \rightarrow \llbracket P \rrbracket$$

$$(a:A) \rightarrow B \text{ \textasciitilde{üres} } \Leftrightarrow \forall a \in A: (B \text{ \textasciitilde{üres} })$$

$$\llbracket \exists a \in A: P \rrbracket = (a : A) \times \llbracket P \rrbracket$$

$$p:(a:A) \times B \Leftrightarrow \text{snd } p : B[a := \text{fst } p]$$

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Bizonyítás a típuselméletben

P-t úgy bizonyítjuk, hogy megadjuk $\llbracket P \rrbracket$ egy elemét:

állítás (logika)	állítás + bizonyítás (típuselmélet)
$P \Rightarrow P$	$a \mapsto a : A \rightarrow A$
$\forall P: P \Rightarrow P$	$A \mapsto a \mapsto a : (A : \mathcal{U}) \rightarrow A \rightarrow A$
$P \Rightarrow Q \Rightarrow Q \wedge P$	$a \mapsto b \mapsto (b, a) : A \rightarrow B \rightarrow B \times A$
$P \wedge Q \Rightarrow P$	$\text{fst} : A \times B \rightarrow A$
$P \Rightarrow \neg \neg P$	$a \mapsto f \mapsto f a : A \rightarrow (A \rightarrow \perp) \rightarrow \perp$
$\perp \Rightarrow P$	$\text{absurd} : \perp \rightarrow A$

- ▶ állítás kimondása \equiv típus megadása
- ▶ bizonyítás \equiv típus elemének megadása
- ▶ bizonyítás ellenőrzése \equiv típusellenőrzés (gép végzi)
- ▶ A bizonyításokkal lehet számolni, akár csak \mathbb{N} elemeivel.

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei
Az alap típuselmélet kibővítése
Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

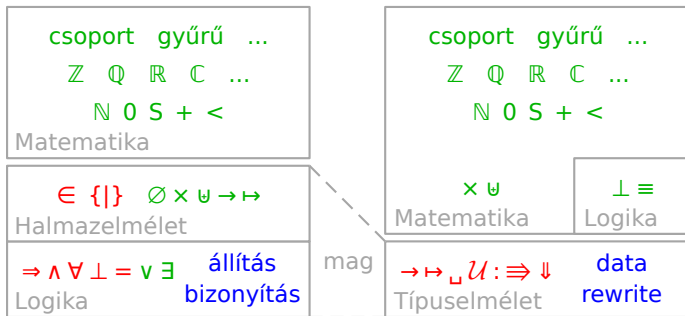
Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Klasszikus matematika vs. típuselmélet felépítése



alapelem nyevi fűszer definíció

Mi a típuselmélet?

Informatikai Logika munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei
Az alap típuselmélet kibővítése
Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált bizonyítások

Milyen formális nyelvet használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Példa: A teljes indukció bizonyítása

rewrite

$$\text{-- } \forall P: P(0) \Rightarrow (\forall m \in \mathbb{N}: P(m) \Rightarrow P(S(m))) \Rightarrow \forall n \in \mathbb{N}: P(n)$$

ind \mathbb{N} :

$$(P : \mathbb{N} \rightarrow \mathcal{U}) \rightarrow$$

$$P\ 0 \rightarrow$$

$$((m : \mathbb{N}) \rightarrow P\ m \rightarrow P\ (S\ m)) \rightarrow$$

$$(n : \mathbb{N}) \rightarrow P\ n$$

$$\text{ind}\mathbb{N}\ P\ z\ s\ 0 \Rightarrow z$$

$$\text{ind}\mathbb{N}\ P\ z\ s\ (S\ n) \Rightarrow s\ n\ (\text{ind}\mathbb{N}\ P\ z\ s\ n)$$

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Az egyenlőség definíciója

data

$$_ \equiv _ : A \rightarrow A \rightarrow \mathcal{U}$$
$$\text{refl} : (x : A) \rightarrow x \equiv x$$

Annak a bizonyítása, hogy $_ \equiv _$ ekvivalencia-reláció:

rewrite

$$\text{sym} : x \equiv y \rightarrow y \equiv x$$
$$\text{sym refl} \Rightarrow \text{refl}$$

rewrite

$$\text{trans} : x \equiv y \rightarrow y \equiv z \rightarrow x \equiv z$$
$$\text{trans refl refl} \Rightarrow \text{refl}$$

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei
Az alap típuselmélet kibővítése
Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Faktor típusok

Egész számok definiálása:

data

$\mathbb{Z} : \mathcal{U}$

$-, _ : \mathbb{N} \rightarrow \mathbb{N} \rightarrow \mathbb{Z}$

-- $\forall a, b, c, d \in \mathbb{N}: a + d = c + b \Rightarrow a - b = c - d$

$\text{EqZ} : (a : \mathbb{N}) \rightarrow (b : \mathbb{N}) \rightarrow (c : \mathbb{N}) \rightarrow (d : \mathbb{N}) \rightarrow a + d \equiv c + b \rightarrow a - b \equiv c - d$

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei
Az alap típuselmélet kibővítése
Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Megjegyzések

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Szokásostól eltérő típuselméleti jelölések a diákon

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

jelölés a dián	tipikus jelölés cikkekben	Agda jelölés
$v \mapsto e$	$\lambda v.e$	$\lambda v \rightarrow e$
$e \Rightarrow e'$	$e \rightsquigarrow e'$	$e = e'$

Az eltérések célja az hogy kevésbé legyen idegen a szintaxis.

Egyéb eltérések az irodalomban szokásostól

- ▶ a normál formát külön nyelvtannal definiálják
- ▶ hipotetikus ítéletek
- ▶ az \mathcal{U} típusú függvényparaméterek sok helyen nincsenek jelölve

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Egyéb

Mi a típuselmélet?

Informatikai Logika munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb

Máshonnan származó képek forrásai

Négyszíntétel:

https://www.vb-helper.com/tutorial_map_coloring.html

Feit-Thompson tétel:

<http://docmadhattan.fieldofscience.com/2012/03/classification-of-finite-simple-groups.html>

Kepler tétel: <https://www.mnn.com/lifestyle/arts-culture/stories/keplers-conjecture-why-it-took-400-years-prove-what-your-grocer-already-knows>

Mi a típuselmélet?

Informatikai Logika
munkacsoport

Motiváció

Mi a formális bizonyítás?

Mik készültek el
típuselmélettel?

Mi a típuselmélet?

A típuselmélet alapelemei

Az alap típuselmélet kibővítése

Programozás típuselmélettel

A munkacsoportról

Mellékletek

Típuselmélettel formalizált
bizonyítások

Milyen formális nyelvet
használjunk?

Típuselméleti definíciók

Logika a típuselméletben

Megjegyzések

Egyéb