

Sedmá přednáška

NAIL062 Výroková a predikátová logika

Jakub Bulín (KTIML MFF UK)

Zimní semestr 2025

Program

- extenze teorií, extenze o definice
- definovatelnost a databázové dotazy
- vztah výrokové a predikátové logiky
- tablo metoda v predikátové logice, jazyky s rovností

Materiály

Zápisky z přednášky, Sekce 6.7-6.9 z Kapitoly 6, Sekce 7.1-7.3 z Kapitoly 7

6.7 Estenze teorií

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$
- **jednoduchá:** $L' = L$

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$
- **jednoduchá:** $L' = L$
- **konzervativní:** $\text{Csq}_L(T) = \text{Csq}_L(T') = \text{Csq}_{L'}(T') \cap \text{Fm}_L$

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$
- **jednoduchá:** $L' = L$
- **konzervativní:** $\text{Csq}_L(T) = \text{Csq}_L(T') = \text{Csq}_{L'}(T') \cap \text{Fm}_L$
- **ekvivalentní:** T' extenzí T a T extenzí T' (obě v témž jazyce)

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$
- **jednoduchá:** $L' = L$
- **konzervativní:** $\text{Csq}_L(T) = \text{Csq}_L(T') = \text{Csq}_{L'}(T') \cap \text{Fm}_L$
- **ekvivalentní:** T' extenzí T a T extenzí T' (obě v témž jazyce)

Jsou-li T, T' ve stejném jazyce L :

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$
- **jednoduchá:** $L' = L$
- **konzervativní:** $\text{Csq}_L(T) = \text{Csq}_L(T') = \text{Csq}_{L'}(T') \cap \text{Fm}_L$
- **ekvivalentní:** T' extenzí T a T extenzí T' (obě v témž jazyce)

Jsou-li T, T' ve stejném jazyce L :

- T' je extenze T , právě když $M_L(T') \subseteq M_L(T)$

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$
- **jednoduchá:** $L' = L$
- **konzervativní:** $\text{Csq}_L(T) = \text{Csq}_L(T') = \text{Csq}_{L'}(T') \cap \text{Fm}_L$
- **ekvivalentní:** T' extenzí T a T extenzí T' (obě v témž jazyce)

Jsou-li T, T' ve stejném jazyce L :

- T' je extenze T , právě když $M_L(T') \subseteq M_L(T)$
- T' je ekvivalentní s T , právě když $M_L(T') = M_L(T)$

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$
- **jednoduchá:** $L' = L$
- **konzervativní:** $\text{Csq}_L(T) = \text{Csq}_L(T') = \text{Csq}_{L'}(T') \cap \text{Fm}_L$
- **ekvivalentní:** T' extenzí T a T extenzí T' (obě v témž jazyce)

Jsou-li T, T' ve stejném jazyce L :

- T' je extenze T , právě když $M_L(T') \subseteq M_L(T)$
- T' je ekvivalentní s T , právě když $M_L(T') = M_L(T)$

Zvětšíme-li jazyk:

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$
- **jednoduchá:** $L' = L$
- **konzervativní:** $\text{Csq}_L(T) = \text{Csq}_L(T') = \text{Csq}_{L'}(T') \cap \text{Fm}_L$
- **ekvivalentní:** T' extenzí T a T extenzí T' (obě v témž jazyce)

Jsou-li T, T' ve stejném jazyce L :

- T' je extenze T , právě když $M_L(T') \subseteq M_L(T)$
- T' je ekvivalentní s T , právě když $M_L(T') = M_L(T)$

Zvětšíme-li jazyk:

- **ve výrokové logice:** přidáváme/zapomínáme hodnoty pro nové prvovýroky

Stejně jako ve výrokové logice, je-li T teorie v jazyce L :

- **extenze:** T' v jazyce $L' \supseteq L$ splňující $\text{Csq}_L(T) \subseteq \text{Csq}_{L'}(T')$
- **jednoduchá:** $L' = L$
- **konzervativní:** $\text{Csq}_L(T) = \text{Csq}_L(T') = \text{Csq}_{L'}(T') \cap \text{Fm}_L$
- **ekvivalentní:** T' extenzí T a T extenzí T' (obě v témž jazyce)

Jsou-li T, T' ve stejném jazyce L :

- T' je extenze T , právě když $M_L(T') \subseteq M_L(T)$
- T' je ekvivalentní s T , právě když $M_L(T') = M_L(T)$

Zvětšíme-li jazyk:

- **ve výrokové logice:** přidáváme/zapomínáme hodnoty pro nové prvovýroky
- **v predikátové logice:** expandujeme/redukujeme modely (přidáváme/zapomínáme nové relace, funkce, konstanty)

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

(i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

- (i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T
- (ii) T' je **konzervativní extenzí** $T \Leftrightarrow T'$ je extenzí T , a každý model T lze expandovat do L' na nějaký model T'

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

- (i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T
- (ii) T' je **konzervativní extenzí** $T \Leftrightarrow T'$ je extenzí T , a každý model T lze expandovat do L' na nějaký model T'

Poznámka: Důkaz (ii) \Rightarrow vynecháme (technický problém: model, který nelze expandovat \rightsquigarrow L -sentence platná v T ale ne v T')

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

- (i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T
- (ii) T' je **konzervativní extenzí** $T \Leftrightarrow T'$ je extenzí T , a každý model T lze expandovat do L' na nějaký model T'

Poznámka: Důkaz (ii) \Rightarrow vynecháme (technický problém: model, který nelze expandovat \rightsquigarrow L -sentence platná v T ale ne v T')

Důkaz:

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

- (i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T
- (ii) T' je **konzervativní extenzí** $T \Leftrightarrow T'$ je extenzí T , a každý model T lze expandovat do L' na nějaký model T'

Poznámka: Důkaz (ii) \Rightarrow vynecháme (technický problém: model, který nelze expandovat \rightsquigarrow L -sentence platná v T ale ne v T')

Důkaz: (i) \Rightarrow Buď \mathcal{A}' model T' , \mathcal{A} jeho L -redukt.

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

- (i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T
- (ii) T' je **konzervativní extenzí** $T \Leftrightarrow T'$ je extenzí T , a každý model T lze expandovat do L' na nějaký model T'

Poznámka: Důkaz (ii) \Rightarrow vynecháme (technický problém: model, který nelze expandovat $\rightsquigarrow L$ -sentence platná v T ale ne v T')

Důkaz: (i) \Rightarrow Buď \mathcal{A}' model T' , \mathcal{A} jeho L -redukt. Protože T' je extenzí, platí v ní, tedy i v \mathcal{A}' , každý axiom $\varphi \in T$. Ten ale obsahuje jen symboly z L , tedy platí i v \mathcal{A} .

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

- (i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T
- (ii) T' je **konzervativní extenzí** $T \Leftrightarrow T'$ je extenzí T , a každý model T lze expandovat do L' na nějaký model T'

Poznámka: Důkaz (ii) \Rightarrow vynecháme (technický problém: model, který nelze expandovat $\rightsquigarrow L$ -sentence platná v T ale ne v T')

Důkaz: (i) \Rightarrow Buď \mathcal{A}' model T' , \mathcal{A} jeho L -redukt. Protože T' je extenzí, platí v ní, tedy i v \mathcal{A}' , každý axiom $\varphi \in T$. Ten ale obsahuje jen symboly z L , tedy platí i v \mathcal{A} .

(i) \Leftarrow **Mějme:** L -sentenci φ , $T \models \varphi$. **Chceme:** $T' \models \varphi$.

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

- (i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T
- (ii) T' je **konzervativní extenzí** $T \Leftrightarrow T'$ je extenzí T , a každý model T lze expandovat do L' na nějaký model T'

Poznámka: Důkaz (ii) \Rightarrow vynecháme (technický problém: model, který nelze expandovat $\rightsquigarrow L$ -sentence platná v T ale ne v T')

Důkaz: (i) \Rightarrow Buď \mathcal{A}' model T' , \mathcal{A} jeho L -redukt. Protože T' je extenzí, platí v ní, tedy i v \mathcal{A}' , každý axiom $\varphi \in T$. Ten ale obsahuje jen symboly z L , tedy platí i v \mathcal{A} .

(i) \Leftarrow **Mějme:** L -sentenci φ , $T \models \varphi$. **Chceme:** $T' \models \varphi$. Pro lib. model $\mathcal{A}' \in M_{L'}(T')$ víme, že jeho L -redukt \mathcal{A} je modelem T , tedy $\mathcal{A} \models \varphi$. Z toho plyne i $\mathcal{A}' \models \varphi$ (opět φ je v L).

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

- (i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T
- (ii) T' je **konzervativní extenzí** $T \Leftrightarrow T'$ je extenzí T , a každý model T lze expandovat do L' na nějaký model T'

Poznámka: Důkaz (ii) \Rightarrow vynecháme (technický problém: model, který nelze expandovat $\rightsquigarrow L$ -sentence platná v T ale ne v T')

Důkaz:(i) \Rightarrow Buď \mathcal{A}' model T' , \mathcal{A} jeho L -redukt. Protože T' je extenzí, platí v ní, tedy i v \mathcal{A}' , každý axiom $\varphi \in T$. Ten ale obsahuje jen symboly z L , tedy platí i v \mathcal{A} .

(i) \Leftarrow **Mějme:** L -sentenci φ , $T \models \varphi$. **Chceme:** $T' \models \varphi$. Pro lib. model $\mathcal{A}' \in M_{L'}(T')$ víme, že jeho L -redukt \mathcal{A} je modelem T , tedy $\mathcal{A} \models \varphi$. Z toho plyne i $\mathcal{A}' \models \varphi$ (opět φ je v L).

(ii) \Leftarrow **Mějme:** L -sentenci φ , $T' \models \varphi$. **Chceme:** $T \models \varphi$.

Extenze teorie: sémantický popis

Mějme jazyky $L \subseteq L'$, L -teorii T a L' -teorii T' :

- (i) T' je **extenzí** $T \Leftrightarrow L$ -redukt každého modelu T' je model T
- (ii) T' je **konzervativní extenzí** $T \Leftrightarrow T'$ je extenzí T , a každý model T lze expandovat do L' na nějaký model T'

Poznámka: Důkaz (ii) \Rightarrow vynecháme (technický problém: model, který nelze expandovat $\rightsquigarrow L$ -sentence platná v T ale ne v T')

Důkaz: (i) \Rightarrow Buď \mathcal{A}' model T' , \mathcal{A} jeho L -redukt. Protože T' je extenzí, platí v ní, tedy i v \mathcal{A}' , každý axiom $\varphi \in T$. Ten ale obsahuje jen symboly z L , tedy platí i v \mathcal{A} .

(i) \Leftarrow **Mějme:** L -sentenci φ , $T \models \varphi$. **Chceme:** $T' \models \varphi$. Pro lib. model $\mathcal{A}' \in M_{L'}(T')$ víme, že jeho L -redukt \mathcal{A} je modelem T , tedy $\mathcal{A} \models \varphi$. Z toho plyne i $\mathcal{A}' \models \varphi$ (opět φ je v L).

(ii) \Leftarrow **Mějme:** L -sentenci φ , $T' \models \varphi$. **Chceme:** $T \models \varphi$. Každý $\mathcal{A} \in M_L(T)$ lze expandovat na nějaký $\mathcal{A}' \in M_{L'}(T')$. Víme, že $\mathcal{A}' \models \varphi$, takže i $\mathcal{A} \models \varphi$. Tím jsme dokázali $T \models \varphi$. □

Extenze o definice (neformálně)

Extenze o definice (neformálně)

- přidáme nový symbol, jehož význam je jednoznačně daný **definující formulí** (jako procedura/funkce v programování)

Extenze o definice (neformálně)

- přidáme nový symbol, jehož význam je jednoznačně daný **definující formulí** (jako procedura/funkce v programování)
- pro relační symboly jednoduché, pro funkční symboly musíme navíc zaručit **existenci** a **jednoznačnost** funkční hodnoty

Extenze o definice (neformálně)

- přidáme nový symbol, jehož význam je jednoznačně daný **definující formulí** (jako procedura/funkce v programování)
- pro relační symboly jednoduché, pro funkční symboly musíme navíc zaručit **existenci** a **jednoznačnost** funkční hodnoty

Ukážeme:

Extenze o definice (neformálně)

- přidáme nový symbol, jehož význam je jednoznačně daný **definující formulí** (jako procedura/funkce v programování)
- pro relační symboly jednoduché, pro funkční symboly musíme navíc zaručit **existenci** a **jednoznačnost** funkční hodnoty

Ukážeme:

- je to konzervativní extenze, dokonce každý model původní teorie lze **jednoznačně** expandovat na model nové teorie

Extenze o definice (neformálně)

- přidáme nový symbol, jehož význam je jednoznačně daný **definující formulí** (jako procedura/funkce v programování)
- pro relační symboly jednoduché, pro funkční symboly musíme navíc zaručit **existenci** a **jednoznačnost** funkční hodnoty

Ukážeme:

- je to konzervativní extenze, dokonce každý model původní teorie lze **jednoznačně** expandovat na model nové teorie
- každou formuli používající nové symboly lze přepsat na formuli v původním jazyce (tak, že jsou v extenzi ekvivalentní)

Definice relačního symbolu

Definice relačního symbolu

nový n -ární relační symbol R lze definovat lib. formulí $\psi(x_1, \dots, x_n)$

Definice relačního symbolu

nový n -ární relační symbol R lze definovat lib. formulí $\psi(x_1, \dots, x_n)$

- teorii v jazyce s rovností lze rozšířit o symbol \neq definovaný formulí $\neg x_1 = x_2$; tj. požadujeme, aby: $x_1 \neq x_2 \leftrightarrow \neg x_1 = x_2$

Definice relačního symbolu

nový n -ární relační symbol R lze definovat lib. formulí $\psi(x_1, \dots, x_n)$

- teorii v jazyce s rovností lze rozšířit o symbol \neq definovaný formulí $\neg x_1 = x_2$; tj. požadujeme, aby: $x_1 \neq x_2 \leftrightarrow \neg x_1 = x_2$
- teorii uspořádání lze rozšířit o $<$ definovaný formulí $x_1 \leq x_2 \wedge \neg x_1 = x_2$; tj. platí: $x_1 < x_2 \leftrightarrow x_1 \leq x_2 \wedge \neg x_1 = x_2$

Definice relačního symbolu

nový n -ární relační symbol R lze definovat lib. formulí $\psi(x_1, \dots, x_n)$

- teorii v jazyce s rovností lze rozšířit o symbol \neq definovaný formulí $\neg x_1 = x_2$; tj. požadujeme, aby: $x_1 \neq x_2 \leftrightarrow \neg x_1 = x_2$
- teorii uspořádání lze rozšířit o $<$ definovaný formulí $x_1 \leq x_2 \wedge \neg x_1 = x_2$; tj. platí: $x_1 < x_2 \leftrightarrow x_1 \leq x_2 \wedge \neg x_1 = x_2$
- v aritmetice lze zavést \leq takto: $x_1 \leq x_2 \leftrightarrow (\exists y)(x_1 + y = x_2)$

Definice relačního symbolu

nový n -ární relační symbol R lze definovat lib. formulí $\psi(x_1, \dots, x_n)$

- teorii v jazyce s rovností lze rozšířit o symbol \neq definovaný formulí $\neg x_1 = x_2$; tj. požadujeme, aby: $x_1 \neq x_2 \leftrightarrow \neg x_1 = x_2$
- teorii uspořádání lze rozšířit o $<$ definovaný formulí $x_1 \leq x_2 \wedge \neg x_1 = x_2$; tj. platí: $x_1 < x_2 \leftrightarrow x_1 \leq x_2 \wedge \neg x_1 = x_2$
- v aritmetice lze zavést \leq takto: $x_1 \leq x_2 \leftrightarrow (\exists y)(x_1 + y = x_2)$
- v uspořádaném stromu lze zavést unární predikát $\text{Leaf}(x)$:
 $\text{Leaf}(x) \leftrightarrow \neg(\exists y)(x <_T y)$

Definice relačního symbolu

nový n -ární relační symbol R lze definovat lib. formulí $\psi(x_1, \dots, x_n)$

- teorii v jazyce s rovností lze rozšířit o symbol \neq definovaný formulí $\neg x_1 = x_2$; tj. požadujeme, aby: $x_1 \neq x_2 \leftrightarrow \neg x_1 = x_2$
- teorii uspořádání lze rozšířit o $<$ definovaný formulí $x_1 \leq x_2 \wedge \neg x_1 = x_2$; tj. platí: $x_1 < x_2 \leftrightarrow x_1 \leq x_2 \wedge \neg x_1 = x_2$
- v aritmetice lze zavést \leq takto: $x_1 \leq x_2 \leftrightarrow (\exists y)(x_1 + y = x_2)$
- v uspořádaném stromu lze zavést unární predikát $\text{Leaf}(x)$:
 $\text{Leaf}(x) \leftrightarrow \neg(\exists y)(x <_T y)$

Mějme teorii T a formuli $\psi(x_1, \dots, x_n)$ v jazyce L . Označme jako L' rozšíření jazyka L o nový n -ární relační symbol R .

Definice relačního symbolu

nový n -ární relační symbol R lze definovat lib. formulí $\psi(x_1, \dots, x_n)$

- teorii v jazyce s rovností lze rozšířit o symbol \neq definovaný formulí $\neg x_1 = x_2$; tj. požadujeme, aby: $x_1 \neq x_2 \leftrightarrow \neg x_1 = x_2$
- teorii uspořádání lze rozšířit o $<$ definovaný formulí $x_1 \leq x_2 \wedge \neg x_1 = x_2$; tj. platí: $x_1 < x_2 \leftrightarrow x_1 \leq x_2 \wedge \neg x_1 = x_2$
- v aritmetice lze zavést \leq takto: $x_1 \leq x_2 \leftrightarrow (\exists y)(x_1 + y = x_2)$
- v uspořádaném stromu lze zavést unární predikát $\text{Leaf}(x)$:
 $\text{Leaf}(x) \leftrightarrow \neg(\exists y)(x <_T y)$

Mějme teorii T a formuli $\psi(x_1, \dots, x_n)$ v jazyce L . Označme jako L' rozšíření jazyka L o nový n -ární relační symbol R . **Extenze teorie T o definici R formulí ψ** je L' -teorie:

Definice relačního symbolu

nový n -ární relační symbol R lze definovat lib. formulí $\psi(x_1, \dots, x_n)$

- teorii v jazyce s rovností lze rozšířit o symbol \neq definovaný formulí $\neg x_1 = x_2$; tj. požadujeme, aby: $x_1 \neq x_2 \leftrightarrow \neg x_1 = x_2$
- teorii uspořádání lze rozšířit o $<$ definovaný formulí $x_1 \leq x_2 \wedge \neg x_1 = x_2$; tj. platí: $x_1 < x_2 \leftrightarrow x_1 \leq x_2 \wedge \neg x_1 = x_2$
- v aritmetice lze zavést \leq takto: $x_1 \leq x_2 \leftrightarrow (\exists y)(x_1 + y = x_2)$
- v uspořádaném stromu lze zavést unární predikát $\text{Leaf}(x)$:
 $\text{Leaf}(x) \leftrightarrow \neg(\exists y)(x <_T y)$

Mějme teorii T a formuli $\psi(x_1, \dots, x_n)$ v jazyce L . Označme jako L' rozšíření jazyka L o nový n -ární relační symbol R . **Extenze teorie T o definici R formulí ψ** je L' -teorie:

$$T' = T \cup \{R(x_1, \dots, x_n) \leftrightarrow \psi(x_1, \dots, x_n)\}$$

Definice relačního symbolu: vlastnosti

Definice relačního symbolu: vlastnosti

Tvrzení:

- (i) T' je konzervativní extenze T .

Definice relačního symbolu: vlastnosti

Tvrzení:

- (i) T' je konzervativní extenze T .
- (ii) Pro každou L' -formuli φ' existuje L -formule φ taková, že $T' \models \varphi' \leftrightarrow \varphi$.

Definice relačního symbolu: vlastnosti

Tvrzení:

- (i) T' je konzervativní extenze T .
- (ii) Pro každou L' -formuli φ' existuje L -formule φ taková, že
$$T' \models \varphi' \leftrightarrow \varphi.$$

Důkaz:

Definice relačního symbolu: vlastnosti

Tvrzení:

- (i) T' je konzervativní extenze T .
- (ii) Pro každou L' -formuli φ' existuje L -formule φ taková, že $T' \models \varphi' \leftrightarrow \varphi$.

Důkaz: (i) ihned ze sémantického popisu extenzí, neboť zřejmě každý model T lze **jednoznačně** expandovat na model T'

Definice relačního symbolu: vlastnosti

Tvrzení:

- (i) T' je konzervativní extenze T .
- (ii) Pro každou L' -formuli φ' existuje L -formule φ taková, že $T' \models \varphi' \leftrightarrow \varphi$.

Důkaz: (i) ihned ze sémantického popisu extenzí, neboť zřejmě každý model T lze **jednoznačně** expandovat na model T'

(ii) atomickou podformulí s novým symbolem R , tj. tvaru $R(t_1, \dots, t_n)$, nahradíme formulí

$$\psi'(x_1/t_1, \dots, x_n/t_n)$$

Definice relačního symbolu: vlastnosti

Tvrzení:

- (i) T' je konzervativní extenze T .
- (ii) Pro každou L' -formuli φ' existuje L -formule φ taková, že $T' \models \varphi' \leftrightarrow \varphi$.

Důkaz: (i) ihned ze sémantického popisu extenzí, neboť zřejmě každý model T lze **jednoznačně** expandovat na model T'

(ii) atomickou podformulí s novým symbolem R , tj. tvaru $R(t_1, \dots, t_n)$, nahradíme formulí

$$\psi'(x_1/t_1, \dots, x_n/t_n)$$

kde ψ' je **varianta ψ zaručující substituovatelnost** všech termů (např. přejmenujeme všechny vázané proměnné ψ na zcela nové) \square

Definice funkčního symbolu: příklady

Definice funkčního symbolu: příklady

vztah $f(x_1, \dots, x_n) = y$ definujeme formulí $\psi(x_1, \dots, x_n, y)$; pro každý vstup (x_1, \dots, x_n) musí **existovat jednoznačný** výstup y

Definice funkčního symbolu: příklady

vztah $f(x_1, \dots, x_n) = y$ definujeme formulí $\psi(x_1, \dots, x_n, y)$; pro každý vstup (x_1, \dots, x_n) musí **existovat jednoznačný** výstup y

1. **Teorie grup:** binární funkční symbol $-_b$ pomocí $+$ a unárního $-$

$$x_1 -_b x_2 = y \leftrightarrow x_1 + (-x_2) = y$$

Definice funkčního symbolu: příklady

vztah $f(x_1, \dots, x_n) = y$ definujeme formulí $\psi(x_1, \dots, x_n, y)$; pro každý vstup (x_1, \dots, x_n) musí **existovat jednoznačný** výstup y

1. **Teorie grup**: binární funkční symbol $-_b$ pomocí $+$ a unárního $-$

$$x_1 -_b x_2 = y \leftrightarrow x_1 + (-x_2) = y$$

- zřejmě pro každá x, y **existuje jednoznačné** z splňující definici

Definice funkčního symbolu: příklady

vztah $f(x_1, \dots, x_n) = y$ definujeme formulí $\psi(x_1, \dots, x_n, y)$; pro každý vstup (x_1, \dots, x_n) musí **existovat jednoznačný** výstup y

1. **Teorie grup**: binární funkční symbol $-_b$ pomocí $+$ a unárního $-$

$$x_1 -_b x_2 = y \leftrightarrow x_1 + (-x_2) = y$$

- zřejmě pro každá x, y **existuje jednoznačné** z splňující definici

2. **Teorie lineárních uspořádání**: binární funkční symbol **min**

$$\min(x_1, x_2) = y \leftrightarrow y \leq x_1 \wedge y \leq x_2 \wedge (\forall z)(z \leq x_1 \wedge z \leq x_2 \rightarrow z \leq y)$$

Definice funkčního symbolu: příklady

vztah $f(x_1, \dots, x_n) = y$ definujeme formulí $\psi(x_1, \dots, x_n, y)$; pro každý vstup (x_1, \dots, x_n) musí **existovat jednoznačný** výstup y

1. **Teorie grup**: binární funkční symbol $-_b$ pomocí $+$ a unárního $-$

$$x_1 -_b x_2 = y \leftrightarrow x_1 + (-x_2) = y$$

- zřejmě pro každá x, y **existuje jednoznačné** z splňující definici

2. **Teorie lineárních uspořádání**: binární funkční symbol **min**

$$\min(x_1, x_2) = y \leftrightarrow y \leq x_1 \wedge y \leq x_2 \wedge (\forall z)(z \leq x_1 \wedge z \leq x_2 \rightarrow z \leq y)$$

- existence a jednoznačnost platí díky linearitě $(x \leq y \vee y \leq x)$

Definice funkčního symbolu: příklady

vztah $f(x_1, \dots, x_n) = y$ definujeme formulí $\psi(x_1, \dots, x_n, y)$; pro každý vstup (x_1, \dots, x_n) musí **existovat jednoznačný** výstup y

1. **Teorie grup**: binární funkční symbol $-_b$ pomocí $+$ a unárního $-$

$$x_1 -_b x_2 = y \leftrightarrow x_1 + (-x_2) = y$$

- zřejmě pro každá x, y **existuje jednoznačné** z splňující definici

2. **Teorie lineárních uspořádání**: binární funkční symbol **min**

$$\min(x_1, x_2) = y \leftrightarrow y \leq x_1 \wedge y \leq x_2 \wedge (\forall z)(z \leq x_1 \wedge z \leq x_2 \rightarrow z \leq y)$$

- existence a jednoznačnost platí díky linearitě ($x \leq y \vee y \leq x$)
- pouze v teorii uspořádání by nešlo o dobrou definici:
 $\min^A(a_1, a_2)$ nemusí existovat

Definice funkčního symbolu: definice

Definice funkčního symbolu: definice

Mějme teorii T a formuli $\psi(x_1, \dots, x_n, y)$ v jazyce L . Označme L' rozšíření L o nový n -ární funkční symbol f . Necht' platí:

Definice funkčního symbolu: definice

Mějme teorii T a formuli $\psi(x_1, \dots, x_n, y)$ v jazyce L . Označme L' rozšíření L o nový n -ární funkční symbol f . Nechť platí:

- $T \models (\exists y)\psi(x_1, \dots, x_n, y)$ (existence)

Definice funkčního symbolu: definice

Mějme teorii T a formuli $\psi(x_1, \dots, x_n, y)$ v jazyce L . Označme L' rozšíření L o nový n -ární funkční symbol f . Nechť platí:

- $T \models (\exists y)\psi(x_1, \dots, x_n, y)$ (existence)
- $T \models \psi(x_1, \dots, x_n, y) \wedge \psi(x_1, \dots, x_n, z) \rightarrow y = z$ (jednoznačnost)

Definice funkčního symbolu: definice

Mějme teorii T a formuli $\psi(x_1, \dots, x_n, y)$ v jazyce L . Označme L' rozšíření L o nový n -ární funkční symbol f . Necht' platí:

- $T \models (\exists y)\psi(x_1, \dots, x_n, y)$ (existence)
- $T \models \psi(x_1, \dots, x_n, y) \wedge \psi(x_1, \dots, x_n, z) \rightarrow y = z$ (jednoznačnost)

Potom **extenze teorie T o definici f formulí ψ** je L' -teorie:

$$T' = T \cup \{f(x_1, \dots, x_n) = y \leftrightarrow \psi(x_1, \dots, x_n, y)\}$$

Definice funkčního symbolu: definice

Mějme teorii T a formuli $\psi(x_1, \dots, x_n, y)$ v jazyce L . Označme L' rozšíření L o nový n -ární funkční symbol f . Nechť platí:

- $T \models (\exists y)\psi(x_1, \dots, x_n, y)$ (existence)
- $T \models \psi(x_1, \dots, x_n, y) \wedge \psi(x_1, \dots, x_n, z) \rightarrow y = z$ (jednoznačnost)

Potom **extenze teorie T o definici f formulí ψ** je L' -teorie:

$$T' = T \cup \{f(x_1, \dots, x_n) = y \leftrightarrow \psi(x_1, \dots, x_n, y)\}$$

- ψ definuje v modelu $(n+1)$ -ární relaci, ta **musí být funkcí**

Definice funkčního symbolu: definice

Mějme teorii T a formuli $\psi(x_1, \dots, x_n, y)$ v jazyce L . Označme L' rozšíření L o nový n -ární funkční symbol f . Necht' platí:

- $T \models (\exists y)\psi(x_1, \dots, x_n, y)$ (existence)
- $T \models \psi(x_1, \dots, x_n, y) \wedge \psi(x_1, \dots, x_n, z) \rightarrow y = z$ (jednoznačnost)

Potom **extenze teorie T o definici f formulí ψ** je L' -teorie:

$$T' = T \cup \{f(x_1, \dots, x_n) = y \leftrightarrow \psi(x_1, \dots, x_n, y)\}$$

- ψ definuje v modelu $(n+1)$ -ární relaci, ta **musí být funkcí**
- je-li ψ tvaru $t(x_1, \dots, x_n) = y$ pro term t , vždy to platí

Definice funkčního symbolu: definice

Mějme teorii T a formuli $\psi(x_1, \dots, x_n, y)$ v jazyce L . Označme L' rozšíření L o nový n -ární funkční symbol f . Necht' platí:

- $T \models (\exists y)\psi(x_1, \dots, x_n, y)$ (existence)
- $T \models \psi(x_1, \dots, x_n, y) \wedge \psi(x_1, \dots, x_n, z) \rightarrow y = z$ (jednoznačnost)

Potom **extenze teorie T o definici f formulí ψ** je L' -teorie:

$$T' = T \cup \{f(x_1, \dots, x_n) = y \leftrightarrow \psi(x_1, \dots, x_n, y)\}$$

- ψ definuje v modelu $(n+1)$ -ární relaci, ta **musí být funkcí**
- je-li ψ tvaru $t(x_1, \dots, x_n) = y$ pro term t , vždy to platí

Tvrzení:

- (i) T' je konzervativní extenze T .

Definice funkčního symbolu: definice

Mějme teorii T a formuli $\psi(x_1, \dots, x_n, y)$ v jazyce L . Označme L' rozšíření L o nový n -ární funkční symbol f . Necht' platí:

- $T \models (\exists y)\psi(x_1, \dots, x_n, y)$ (existence)
- $T \models \psi(x_1, \dots, x_n, y) \wedge \psi(x_1, \dots, x_n, z) \rightarrow y = z$ (jednoznačnost)

Potom **extenze teorie T o definici f formulí ψ** je L' -teorie:

$$T' = T \cup \{f(x_1, \dots, x_n) = y \leftrightarrow \psi(x_1, \dots, x_n, y)\}$$

- ψ definuje v modelu $(n+1)$ -ární relaci, ta **musí být funkcí**
- je-li ψ tvaru $t(x_1, \dots, x_n) = y$ pro term t , vždy to platí

Tvrzení:

- (i) T' je konzervativní extenze T .
- (ii) Pro každou L' -formuli φ' existuje L -formule φ taková, že $T' \models \varphi' \leftrightarrow \varphi$.

Definice funkčního symbolu: definice

Mějme teorii T a formuli $\psi(x_1, \dots, x_n, y)$ v jazyce L . Označme L' rozšíření L o nový n -ární funkční symbol f . Necht' platí:

- $T \models (\exists y)\psi(x_1, \dots, x_n, y)$ (existence)
- $T \models \psi(x_1, \dots, x_n, y) \wedge \psi(x_1, \dots, x_n, z) \rightarrow y = z$ (jednoznačnost)

Potom **extenze teorie T o definici f formulí ψ** je L' -teorie:

$$T' = T \cup \{f(x_1, \dots, x_n) = y \leftrightarrow \psi(x_1, \dots, x_n, y)\}$$

- ψ definuje v modelu $(n+1)$ -ární relaci, ta **musí být funkcí**
- je-li ψ tvaru $t(x_1, \dots, x_n) = y$ pro term t , vždy to platí

Tvrzení:

- (i) T' je konzervativní extenze T .
- (ii) Pro každou L' -formuli φ' existuje L -formule φ taková, že $T' \models \varphi' \leftrightarrow \varphi$.

Důkaz: (i) modely T lze **jednoznačně** expandovat na modely T'

(ii) stačí pro jediný výskyt symbolu f , jinak induktivně (je-li více vnořených výskytů $f(\dots f(\dots) \dots)$, potom od vnitřních k vnějším)

(ii) stačí pro jediný výskyt symbolu f , jinak induktivně (je-li více vnořených výskytů $f(\dots f(\dots) \dots)$, potom od vnitřních k vnějším)

1. nahradíme term $f(t_1, \dots, t_n)$ novou proměnnou z : výsledek φ^*

(ii) stačí pro jediný výskyt symbolu f , jinak induktivně (je-li více vnořených výskytů $f(\dots f(\dots) \dots)$, potom od vnitřních k vnějším)

1. nahradíme term $f(t_1, \dots, t_n)$ novou proměnnou z : výsledek φ^*
2. φ zkonstruujeme takto: $(\exists z)(\varphi^* \wedge \psi'(x_1/t_1, \dots, x_n/t_n, y/z))$
(kde ψ' je varianta ψ zaručující substituovatelnost)

Pokračování důkazu

(ii) stačí pro jediný výskyt symbolu f , jinak induktivně (je-li více vnořených výskytů $f(\dots f(\dots) \dots)$, potom od vnitřních k vnějším)

1. nahradíme term $f(t_1, \dots, t_n)$ **novou** proměnnou z : **výsledek** φ^*
2. φ zkonstruujeme takto: $(\exists z)(\varphi^* \wedge \psi'(x_1/t_1, \dots, x_n/t_n, y/z))$
(kde ψ' je varianta ψ zaručující substituovatelnost)

Ukážeme, že pro libovolný model $\mathcal{A} \models T'$ a ohodnocení e platí:

$$\mathcal{A} \models \varphi'[e] \quad \text{právě když} \quad \mathcal{A} \models \varphi[e]$$

(ii) stačí pro jediný výskyt symbolu f , jinak induktivně (je-li více vnořených výskytů $f(\dots f(\dots) \dots)$, potom od vnitřních k vnějším)

1. nahradíme term $f(t_1, \dots, t_n)$ novou proměnnou z : **výsledek** φ^*
2. φ zkonstruujeme takto: $(\exists z)(\varphi^* \wedge \psi'(x_1/t_1, \dots, x_n/t_n, y/z))$
(kde ψ' je varianta ψ zaručující substituovatelnost)

Ukážeme, že pro libovolný model $\mathcal{A} \models T'$ a ohodnocení e platí:

$$\mathcal{A} \models \varphi'[e] \quad \text{právě když} \quad \mathcal{A} \models \varphi[e]$$

Označme $a = (f(t_1, \dots, t_n))^{\mathcal{A}}[e]$. Díky existenci a jednoznačnosti:

$$\mathcal{A} \models \psi'(x_1/t_1, \dots, x_n/t_n, y/z)[e] \quad \text{právě když} \quad e(z) = a$$

(ii) stačí pro jediný výskyt symbolu f , jinak induktivně (je-li více vnořených výskytů $f(\dots f(\dots) \dots)$, potom od vnitřních k vnějším)

1. nahradíme term $f(t_1, \dots, t_n)$ **novou** proměnnou z : **výsledek** φ^*
2. φ zkonstruujeme takto: $(\exists z)(\varphi^* \wedge \psi'(x_1/t_1, \dots, x_n/t_n, y/z))$
(kde ψ' je varianta ψ zaručující substituovatelnost)

Ukážeme, že pro libovolný model $\mathcal{A} \models T'$ a ohodnocení e platí:

$$\mathcal{A} \models \varphi'[e] \quad \text{právě když} \quad \mathcal{A} \models \varphi[e]$$

Označme $a = (f(t_1, \dots, t_n))^{\mathcal{A}}[e]$. Díky existenci a jednoznačnosti:

$$\mathcal{A} \models \psi'(x_1/t_1, \dots, x_n/t_n, y/z)[e] \quad \text{právě když} \quad e(z) = a$$

Máme tedy: $\mathcal{A} \models \varphi'[e] \Leftrightarrow \mathcal{A} \models \varphi^*[e(z/a)] \Leftrightarrow \mathcal{A} \models \varphi[e]$ □

Definice konstantního symbolu

- speciální případ: funkční symbol arity 0

Definice konstantního symbolu

- **speciální případ:** funkční symbol arity 0
- extenze o definici konstantního symbolu c formulí $\psi(y)$:

Definice konstantního symbolu

- speciální případ: funkční symbol arity 0
- extenze o definici konstantního symbolu c formulí $\psi(y)$:

$$T' = T \cup \{c = y \leftrightarrow \psi(y)\}$$

Definice konstantního symbolu

- speciální případ: funkční symbol arity 0
- extenze o definici konstantního symbolu c formulí $\psi(y)$:

$$T' = T \cup \{c = y \leftrightarrow \psi(y)\}$$

- musí platit $T \models (\exists y)\psi(y)$ a $T \models \psi(y) \wedge \psi(z) \rightarrow y = z$

Definice konstantního symbolu

- **speciální případ:** funkční symbol arity 0
- extenze o definici konstantního symbolu c formulí $\psi(y)$:

$$T' = T \cup \{c = y \leftrightarrow \psi(y)\}$$

- musí platit $T \models (\exists y)\psi(y)$ a $T \models \psi(y) \wedge \psi(z) \rightarrow y = z$
- platí stejná tvrzení

Definice konstantního symbolu

- **speciální případ:** funkční symbol arity 0
- extenze o definici konstantního symbolu c formulí $\psi(y)$:

$$T' = T \cup \{c = y \leftrightarrow \psi(y)\}$$

- musí platit $T \models (\exists y)\psi(y)$ a $T \models \psi(y) \wedge \psi(z) \rightarrow y = z$
- platí stejná tvrzení

1. teorie v jazyce aritmetiky, rozšíříme o definici symbolu 1 formulí $\psi(y)$ tvaru $y = S(0)$, přidáme tedy axiom $1 = y \leftrightarrow y = S(0)$

Definice konstantního symbolu

- **speciální případ:** funkční symbol arity 0
- extenze o definici konstantního symbolu c formulí $\psi(y)$:

$$T' = T \cup \{c = y \leftrightarrow \psi(y)\}$$

- musí platit $T \models (\exists y)\psi(y)$ a $T \models \psi(y) \wedge \psi(z) \rightarrow y = z$
- platí stejná tvrzení

1. teorie v jazyce aritmetiky, rozšíříme o definici symbolu 1 formulí $\psi(y)$ tvaru $y = S(0)$, přidáme tedy axiom $1 = y \leftrightarrow y = S(0)$

2. teorie těles, nový symbol $\frac{1}{2}$, definice formulí $y \cdot (1 + 1) = 1$, tj. přidáním $\frac{1}{2} = y \leftrightarrow y \cdot (1 + 1) = 1$?

Definice konstantního symbolu

- **speciální případ:** funkční symbol arity 0
- extenze o definici konstantního symbolu c formulí $\psi(y)$:

$$T' = T \cup \{c = y \leftrightarrow \psi(y)\}$$

- musí platit $T \models (\exists y)\psi(y)$ a $T \models \psi(y) \wedge \psi(z) \rightarrow y = z$
- platí stejná tvrzení

1. teorie v jazyce aritmetiky, rozšíříme o definici symbolu 1 formulí $\psi(y)$ tvaru $y = S(0)$, přidáme tedy axiom $1 = y \leftrightarrow y = S(0)$

2. teorie těles, nový symbol $\frac{1}{2}$, definice formulí $y \cdot (1 + 1) = 1$, tj. přidáním $\frac{1}{2} = y \leftrightarrow y \cdot (1 + 1) = 1$?

- není extenze o definici! neplatí existence: v tělese charakteristiky 2, např. \mathbb{Z}_2 , nemá rovnice $y \cdot (1 + 1) = 1$ řešení

Definice konstantního symbolu

- **speciální případ:** funkční symbol arity 0
- extenze o definici konstantního symbolu c formulí $\psi(y)$:

$$T' = T \cup \{c = y \leftrightarrow \psi(y)\}$$

- musí platit $T \models (\exists y)\psi(y)$ a $T \models \psi(y) \wedge \psi(z) \rightarrow y = z$
- platí stejná tvrzení

1. teorie v jazyce aritmetiky, rozšíříme o definici symbolu 1 formulí $\psi(y)$ tvaru $y = S(0)$, přidáme tedy axiom $1 = y \leftrightarrow y = S(0)$

2. teorie těles, nový symbol $\frac{1}{2}$, definice formulí $y \cdot (1 + 1) = 1$, tj. přidáním $\frac{1}{2} = y \leftrightarrow y \cdot (1 + 1) = 1$?

- není extenze o definici! neplatí existence: v tělese charakteristiky 2, např. \mathbb{Z}_2 , nemá rovnice $y \cdot (1 + 1) = 1$ řešení
- ale v teorii těles charakteristiky různé od 2, tj. přidáme-li axiom $\neg(1 + 1 = 0)$, už ano; např. v \mathbb{Z}_3 máme $\frac{1}{2}^{\mathbb{Z}_3} = 2$

Estenze o definice

L' -teorie T' je **extenzí** L -teorie T o **definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Extenze o definice

L' -teorie T' je **extenzí** L -teorie T o **definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Tvrzení: (snadno indukcí)

Extenze o definice

L' -teorie T' je **extenzí** L -teorie T o **definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Tvrzení: (snadno indukcí)

- Každý model T lze jednoznačně expandovat na model T' .

Extenze o definice

L' -teorie T' je **extenzí** L -teorie T o **definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Tvrzení: (snadno indukcí)

- Každý model T lze jednoznačně expandovat na model T' .
- T' je konzervativní extenze T .

Extenze o definice

L' -teorie T' je **extenzí** L -teorie T o **definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Tvrzení: (snadno indukcí)

- Každý model T lze jednoznačně expandovat na model T' .
- T' je konzervativní extenze T .
- Pro L' -formuli φ' existuje L -formule φ , že $T' \models \varphi' \leftrightarrow \varphi$.

Extenze o definice

L' -teorie T' je **extenzí** L -teorie T o **definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Tvrzení: (snadno indukcí)

- Každý model T lze jednoznačně expandovat na model T' .
- T' je konzervativní extenze T .
- Pro L' -formuli φ' existuje L -formule φ , že $T' \models \varphi' \leftrightarrow \varphi$.

Příklad: $T = \{(\exists y)(x + y = 0), (x + y = 0) \wedge (x + z = 0) \rightarrow y = z\}$

Extenze o definice

L' -teorie T' je **extenzí** L -teorie T o **definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Tvrzení: (snadno indukcí)

- Každý model T lze jednoznačně expandovat na model T' .
- T' je konzervativní extenze T .
- Pro L' -formuli φ' existuje L -formule φ , že $T' \models \varphi' \leftrightarrow \varphi$.

Příklad: $T = \{(\exists y)(x + y = 0), (x + y = 0) \wedge (x + z = 0) \rightarrow y = z\}$

$L = \langle +, 0, \leq \rangle$ s rovností, zavedeme $<$ a unární $-$ přidáním axiomů:

Extenze o definice

L' -teorie T' je **extenzí** L -teorie T o **definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Tvrzení: (snadno indukcí)

- Každý model T lze jednoznačně expandovat na model T' .
- T' je konzervativní extenze T .
- Pro L' -formuli φ' existuje L -formule φ , že $T' \models \varphi' \leftrightarrow \varphi$.

Příklad: $T = \{(\exists y)(x + y = 0), (x + y = 0) \wedge (x + z = 0) \rightarrow y = z\}$

$L = \langle +, 0, \leq \rangle$ s rovností, zavedeme $<$ a unární $-$ přidáním axiomů:

$$\begin{aligned} T' = T \cup \{ & -x = y \leftrightarrow x + y = 0, \\ & x < y \leftrightarrow x \leq y \wedge \neg(x = y) \} \end{aligned}$$

Extenze o definice

L' -teorie T' je **extenzí** L -teorie T **o definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Tvrzení: (snadno indukcí)

- Každý model T lze jednoznačně expandovat na model T' .
- T' je konzervativní extenze T .
- Pro L' -formuli φ' existuje L -formule φ , že $T' \models \varphi' \leftrightarrow \varphi$.

Příklad: $T = \{(\exists y)(x + y = 0), (x + y = 0) \wedge (x + z = 0) \rightarrow y = z\}$

$L = \langle +, 0, \leq \rangle$ s rovností, zavedeme $<$ a unární $-$ přidáním axiomů:

$$\begin{aligned} T' = T \cup \{ & -x = y \leftrightarrow x + y = 0, \\ & x < y \leftrightarrow x \leq y \wedge \neg(x = y) \} \end{aligned}$$

Formule $-x < y$ v jazyce $L' = \langle +, -, 0, \leq, < \rangle$ s rovností je v T' ekvivalentní formuli:

Extenze o definice

L' -teorie T' je **extenzí** L -teorie T **o definice**, pokud vznikla postupnou extenzí o definice relačních a funkčních (vč. konstantních) symbolů.

Tvrzení: (snadno indukcí)

- Každý model T lze jednoznačně expandovat na model T' .
- T' je konzervativní extenze T .
- Pro L' -formuli φ' existuje L -formule φ , že $T' \models \varphi' \leftrightarrow \varphi$.

Příklad: $T = \{(\exists y)(x + y = 0), (x + y = 0) \wedge (x + z = 0) \rightarrow y = z\}$

$L = \langle +, 0, \leq \rangle$ s rovností, zavedeme $<$ a unární $-$ přidáním axiomů:

$$\begin{aligned} T' = T \cup \{ & -x = y \leftrightarrow x + y = 0, \\ & x < y \leftrightarrow x \leq y \wedge \neg(x = y) \} \end{aligned}$$

Formule $-x < y$ v jazyce $L' = \langle +, -, 0, \leq, < \rangle$ s rovností je v T' ekvivalentní formuli: $(\exists z)((z \leq y \wedge \neg(z = y)) \wedge x + z = 0)$

6.8 Definovatelnost ve struktuře

- formule φ s jednou volnou proměnnou $x \dots$ “vlastnost” prvků

Definovatelné množiny

- formule φ s jednou volnou proměnnou $x \dots$ “vlastnost” prvků
- ve struktuře **definuje** množinu prvků, které vlastnost splňují (tj. prvků a takových, že φ platí při ohodnocení kde $e(x) = a$)

Definovatelné množiny

- formule φ s jednou volnou proměnnou x ... “vlastnost” prvků
- ve struktuře **definuje** množinu prvků, které vlastnost splňují (tj. prvků a takových, že φ platí při ohodnocení kde $e(x) = a$)
- $\varphi(x, y)$ definuje binární relaci, atp.

Definovatelné množiny

- formule φ s jednou volnou proměnnou $x \dots$ “vlastnost” prvků
- ve struktuře **definuje** množinu prvků, které vlastnost splňují (tj. prvků a takových, že φ platí při ohodnocení kde $e(x) = a$)
- $\varphi(x, y)$ definuje binární relaci, atp.

Množina **definovaná** $\varphi(x_1, \dots, x_n)$ **ve struktuře** \mathcal{A} (v témž jazyce):

$$\varphi^{\mathcal{A}}(x_1, \dots, x_n) = \{(a_1, \dots, a_n) \in A^n \mid \mathcal{A} \models \varphi[e(x_1/a_1, \dots, x_n/a_n)]\}$$

Definovatelné množiny

- formule φ s jednou volnou proměnnou $x \dots$ “vlastnost” prvků
- ve struktuře **definuje** množinu prvků, které vlastnost splňují (tj. prvků a takových, že φ platí při ohodnocení kde $e(x) = a$)
- $\varphi(x, y)$ definuje binární relaci, atp.

Množina **definovaná** $\varphi(x_1, \dots, x_n)$ **ve struktuře** \mathcal{A} (v témž jazyce):

$$\varphi^{\mathcal{A}}(x_1, \dots, x_n) = \{(a_1, \dots, a_n) \in A^n \mid \mathcal{A} \models \varphi[e(x_1/a_1, \dots, x_n/a_n)]\}$$

Zkráceně píšeme: $\varphi^{\mathcal{A}}(\bar{x}) = \{\bar{a} \in A^n \mid \mathcal{A} \models \varphi[e(\bar{x}/\bar{a})]\}$

Definovatelné množiny

- formule φ s jednou volnou proměnnou $x \dots$ “vlastnost” prvků
- ve struktuře **definuje** množinu prvků, které vlastnost splňují (tj. prvků a takových, že φ platí při ohodnocení kde $e(x) = a$)
- $\varphi(x, y)$ definuje binární relaci, atp.

Množina **definovaná** $\varphi(x_1, \dots, x_n)$ **ve struktuře** \mathcal{A} (v témž jazyce):

$$\varphi^{\mathcal{A}}(x_1, \dots, x_n) = \{(a_1, \dots, a_n) \in A^n \mid \mathcal{A} \models \varphi[e(x_1/a_1, \dots, x_n/a_n)]\}$$

Zkráceně píšeme: $\varphi^{\mathcal{A}}(\bar{x}) = \{\bar{a} \in A^n \mid \mathcal{A} \models \varphi[e(\bar{x}/\bar{a})]\}$

- formule $\neg(\exists y)E(x, y)$ definuje **v daném grafu** množinu všech **izolovaných** vrcholů

Definovatelné množiny

- formule φ s jednou volnou proměnnou $x \dots$ “vlastnost” prvků
- ve struktuře **definuje** množinu prvků, které vlastnost splňují (tj. prvků a takových, že φ platí při ohodnocení kde $e(x) = a$)
- $\varphi(x, y)$ definuje binární relaci, atp.

Množina **definovaná** $\varphi(x_1, \dots, x_n)$ **ve struktuře** \mathcal{A} (v témž jazyce):

$$\varphi^{\mathcal{A}}(x_1, \dots, x_n) = \{(a_1, \dots, a_n) \in A^n \mid \mathcal{A} \models \varphi[e(x_1/a_1, \dots, x_n/a_n)]\}$$

Zkráceně píšeme: $\varphi^{\mathcal{A}}(\bar{x}) = \{\bar{a} \in A^n \mid \mathcal{A} \models \varphi[e(\bar{x}/\bar{a})]\}$

- formule $\neg(\exists y)E(x, y)$ definuje **v daném grafu** množinu všech **izolovaných** vrcholů
- $(\exists y)(y \cdot y = x) \wedge \neg(x = 0)$ definuje **v tělese \mathbb{R}** množinu všech kladných reálných čísel

Definovatelné množiny

- formule φ s jednou volnou proměnnou $x \dots$ “vlastnost” prvků
- ve struktuře **definuje** množinu prvků, které vlastnost splňují (tj. prvků a takových, že φ platí při ohodnocení kde $e(x) = a$)
- $\varphi(x, y)$ definuje binární relaci, atp.

Množina **definovaná** $\varphi(x_1, \dots, x_n)$ **ve struktuře** \mathcal{A} (v témž jazyce):

$$\varphi^{\mathcal{A}}(x_1, \dots, x_n) = \{(a_1, \dots, a_n) \in A^n \mid \mathcal{A} \models \varphi[e(x_1/a_1, \dots, x_n/a_n)]\}$$

Zkráceně píšeme: $\varphi^{\mathcal{A}}(\bar{x}) = \{\bar{a} \in A^n \mid \mathcal{A} \models \varphi[e(\bar{x}/\bar{a})]\}$

- formule $\neg(\exists y)E(x, y)$ definuje **v daném grafu** množinu všech **izolovaných** vrcholů
- $(\exists y)(y \cdot y = x) \wedge \neg(x = 0)$ definuje **v tělese \mathbb{R}** množinu všech kladných reálných čísel
- $x \leq y \wedge \neg(x = y)$ definuje v **uspořádané množině $\langle S, \leq^S \rangle$** relaci **ostrého uspořádání** $<^S$

Definovatelnost s parametry

- vlastnosti prvků relativně k jiným prvkům? nelze čistě syntakticky, ale můžeme dosadit prvky jako **parametry**

Definovatelnost s parametry

- vlastnosti prvků relativně k jiným prvkům? nelze čistě syntakticky, ale můžeme dosadit prvky jako **parametry**
- zápis $\varphi(\bar{x}, \bar{y})$: volné proměnné $x_1, \dots, x_n, y_1, \dots, y_k$

Definovatelnost s parametry

- vlastnosti prvků relativně k jiným prvkům? nelze čistě syntakticky, ale můžeme dosadit prvky jako **parametry**
- zápis $\varphi(\bar{x}, \bar{y})$: volné proměnné $x_1, \dots, x_n, y_1, \dots, y_k$

Mějme $\varphi(\bar{x}, \bar{y})$ (kde $|\bar{x}| = n$, $|\bar{y}| = k$), strukturu \mathcal{A} (v témž jazyce), $\bar{b} \in A^k$. Množina **definovaná** $\varphi(\bar{x}, \bar{y})$ **s parametry** \bar{b} **ve struktuře** \mathcal{A} :

$$\varphi^{\mathcal{A}, \bar{b}}(\bar{x}, \bar{y}) = \{\bar{a} \in A^n \mid \mathcal{A} \models \varphi[e(\bar{x}/\bar{a}, \bar{y}/\bar{b})]\}$$

Definovatelnost s parametry

- vlastnosti prvků relativně k jiným prvkům? nelze čistě syntakticky, ale můžeme dosadit prvky jako **parametry**
- zápis $\varphi(\bar{x}, \bar{y})$: volné proměnné $x_1, \dots, x_n, y_1, \dots, y_k$

Mějme $\varphi(\bar{x}, \bar{y})$ (kde $|\bar{x}| = n$, $|\bar{y}| = k$), strukturu \mathcal{A} (v témž jazyce), $\bar{b} \in A^k$. Množina **definovaná** $\varphi(\bar{x}, \bar{y})$ **s parametry** \bar{b} **ve struktuře** \mathcal{A} :

$$\varphi^{\mathcal{A}, \bar{b}}(\bar{x}, \bar{y}) = \{\bar{a} \in A^n \mid \mathcal{A} \models \varphi[e(\bar{x}/\bar{a}, \bar{y}/\bar{b})]\}$$

Pro $B \subseteq A$ označíme $\text{Df}^n(\mathcal{A}, B)$ množinu všech množin definovatelných v \mathcal{A} s parametry pocházejícími z B .

Definovatelnost s parametry

- vlastnosti prvků relativně k jiným prvkům? nelze čistě syntakticky, ale můžeme dosadit prvky jako **parametry**
- zápis $\varphi(\bar{x}, \bar{y})$: volné proměnné $x_1, \dots, x_n, y_1, \dots, y_k$

Mějme $\varphi(\bar{x}, \bar{y})$ (kde $|\bar{x}| = n$, $|\bar{y}| = k$), strukturu \mathcal{A} (v témž jazyce), $\bar{b} \in A^k$. Množina **definovaná** $\varphi(\bar{x}, \bar{y})$ **s parametry** \bar{b} **ve struktuře** \mathcal{A} :

$$\varphi^{\mathcal{A}, \bar{b}}(\bar{x}, \bar{y}) = \{\bar{a} \in A^n \mid \mathcal{A} \models \varphi[e(\bar{x}/\bar{a}, \bar{y}/\bar{b})]\}$$

Pro $B \subseteq A$ označíme $\text{Df}^n(\mathcal{A}, B)$ množinu všech množin definovatelných v \mathcal{A} s parametry pocházejícími z B .

Pozorování: $\text{Df}^n(\mathcal{A}, B)$ je uzavřená na doplněk, průnik, sjednocení, a obsahuje \emptyset a A^n : je to **podalgebra potenční algebry** $\mathcal{P}(A^n)$.

Definovatelnost s parametry

- vlastnosti prvků relativně k jiným prvkům? nelze čistě syntakticky, ale můžeme dosadit prvky jako **parametry**
- zápis $\varphi(\bar{x}, \bar{y})$: volné proměnné $x_1, \dots, x_n, y_1, \dots, y_k$

Mějme $\varphi(\bar{x}, \bar{y})$ (kde $|\bar{x}| = n$, $|\bar{y}| = k$), strukturu \mathcal{A} (v témž jazyce), $\bar{b} \in A^k$. Množina **definovaná** $\varphi(\bar{x}, \bar{y})$ **s parametry** \bar{b} **ve struktuře** \mathcal{A} :

$$\varphi^{\mathcal{A}, \bar{b}}(\bar{x}, \bar{y}) = \{\bar{a} \in A^n \mid \mathcal{A} \models \varphi[e(\bar{x}/\bar{a}, \bar{y}/\bar{b})]\}$$

Pro $B \subseteq A$ označíme $\text{Df}^n(\mathcal{A}, B)$ množinu všech množin definovatelných v \mathcal{A} s parametry pocházejícími z B .

Pozorování: $\text{Df}^n(\mathcal{A}, B)$ je uzavřená na doplněk, průnik, sjednocení, a obsahuje \emptyset a A^n : je to **podalgebra potenční algebry** $\mathcal{P}(A^n)$.

Např. pro $\varphi(x, y) = E(x, y)$ a vrchol $v \in V(\mathcal{G})$ je $\varphi^{\mathcal{G}, v}(x, y)$ množina všech sousedů vrcholu v .

- relační databáze: jedna nebo více tabulek, také relace

Aplikace: databázové dotazy

- relační databáze: jedna nebo více tabulek, také relace
- řádky tabulky jsou záznamy (records), také tice (tuples)

Aplikace: databázové dotazy

- relační databáze: jedna nebo více tabulek, také relace
- řádky tabulky jsou záznamy (records), také tice (tuples)
- struktura v čistě relačním jazyce

Aplikace: databázové dotazy

- **relační databáze**: jedna nebo více **tabulek**, také **relace**
- řádky tabulky jsou **záznamy (records)**, také **tice (tuples)**
- struktura v čistě relačním jazyce

Movies

title	director	actor
Forrest Gump	R. Zemeckis	T. Hanks
Philadelphia	J. Demme	T. Hanks
Batman Returns	T. Burton	M. Keaton
⋮	⋮	⋮

Aplikace: databázové dotazy

- **relační databáze**: jedna nebo více **tabulek**, také **relace**
- řádky tabulky jsou **záznamy (records)**, také **tice (tuples)**
- struktura v čistě relačním jazyce

Movies

title	director	actor
Forrest Gump	R. Zemeckis	T. Hanks
Philadelphia	J. Demme	T. Hanks
Batman Returns	T. Burton	M. Keaton
⋮	⋮	⋮

Program

cinema	title	time
Atlas	Forrest Gump	20:00
Lucerna	Forrest Gump	21:00
Lucerna	Philadelphia	18:30
⋮	⋮	⋮

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)
- výsledek je množina definovaná touto formulí (s parametry)

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)
- výsledek je množina definovaná touto formulí (s parametry)

“Kdy a kde můžeme vidět film s Tomem Hanksem?”

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)
- výsledek je množina definovaná touto formulí (s parametry)

“Kdy a kde můžeme vidět film s Tomem Hanksem?”

```
select Program.cinema, Program.time from Program, Movies where  
Program.title = Movies.title and Movies.actor = 'T. Hanks'
```

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)
- výsledek je množina definovaná touto formulí (s parametry)

“Kdy a kde můžeme vidět film s Tomem Hanksem?”

```
select Program.cinema, Program.time from Program, Movies where  
Program.title = Movies.title and Movies.actor = 'T. Hanks'
```

- výsledek je množina $\varphi^{\text{Database, 'T. Hanks'}}(x_{\text{cinema}}, x_{\text{time}}, y_{\text{actor}})$

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)
- výsledek je množina definovaná touto formulí (s parametry)

“Kdy a kde můžeme vidět film s Tomem Hanksem?”

```
select Program.cinema, Program.time from Program, Movies where  
Program.title = Movies.title and Movies.actor = 'T. Hanks'
```

- výsledek je množina $\varphi^{\text{Database}, 'T. Hanks'}(x_{\text{cinema}}, x_{\text{time}}, y_{\text{actor}})$
- definovaná ve struktuře **Database** = $\langle D, \text{Program}, \text{Movies} \rangle$

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)
- výsledek je množina definovaná touto formulí (s parametry)

“Kdy a kde můžeme vidět film s Tomem Hanksem?”

```
select Program.cinema, Program.time from Program, Movies where  
Program.title = Movies.title and Movies.actor = 'T. Hanks'
```

- výsledek je množina $\varphi^{\text{Database}, 'T. Hanks'}(x_{\text{cinema}}, x_{\text{time}}, y_{\text{actor}})$
- definovaná ve struktuře **Database** = $\langle D, \text{Program}, \text{Movies} \rangle$
- jejíž doména je $D = \{ \text{'Atlas'}, \text{'Lucerna'}, \dots, \text{'M. Keaton'} \}$

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)
- výsledek je množina definovaná touto formulí (s parametry)

“Kdy a kde můžeme vidět film s Tomem Hanksem?”

```
select Program.cinema, Program.time from Program, Movies where  
Program.title = Movies.title and Movies.actor = 'T. Hanks'
```

- výsledek je množina $\varphi^{\text{Database}, 'T. Hanks'}(x_{\text{cinema}}, x_{\text{time}}, y_{\text{actor}})$
- definovaná ve struktuře **Database** = $\langle D, \text{Program}, \text{Movies} \rangle$
- jejíž doména je $D = \{ \text{'Atlas'}, \text{'Lucerna'}, \dots, \text{'M. Keaton'} \}$
- s parametrem **'T. Hanks'**,

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)
- výsledek je množina definovaná touto formulí (s parametry)

“Kdy a kde můžeme vidět film s Tomem Hanksem?”

```
select Program.cinema, Program.time from Program, Movies where  
Program.title = Movies.title and Movies.actor = 'T. Hanks'
```

- výsledek je množina $\varphi^{\text{Database}, 'T. Hanks'}(x_{\text{cinema}}, x_{\text{time}}, y_{\text{actor}})$
- definovaná ve struktuře **Database** = $\langle D, \text{Program}, \text{Movies} \rangle$
- jejíž doména je $D = \{ \text{'Atlas'}, \text{'Lucerna'}, \dots, \text{'M. Keaton'} \}$
- s parametrem **'T. Hanks'**,
- definující formule $\varphi(x_{\text{cinema}}, x_{\text{time}}, y_{\text{actor}})$:

Příklad SQL dotazu

- SQL dotaz v nejjednodušší formě je formule (pomineme např. **agregační funkce**)
- výsledek je množina definovaná touto formulí (s parametry)

“Kdy a kde můžeme vidět film s Tomem Hanksem?”

```
select Program.cinema, Program.time from Program, Movies where  
Program.title = Movies.title and Movies.actor = 'T. Hanks'
```

- výsledek je množina $\varphi^{\text{Database}, 'T. Hanks'}(x_{\text{cinema}}, x_{\text{time}}, y_{\text{actor}})$
- definovaná ve struktuře **Database** = $\langle D, \text{Program}, \text{Movies} \rangle$
- jejíž doména je $D = \{ \text{'Atlas'}, \text{'Lucerna'}, \dots, \text{'M. Keaton'} \}$
- s parametrem **'T. Hanks'**,
- definující formule $\varphi(x_{\text{cinema}}, x_{\text{time}}, y_{\text{actor}})$:

$$(\exists z_{\text{title}})(\exists z_{\text{director}})(\text{Program}(x_{\text{cinema}}, z_{\text{title}}, x_{\text{time}}) \wedge \\ \text{Movies}(z_{\text{title}}, z_{\text{director}}, y_{\text{actor}}))$$

6.9 Vztah výrokové a predikátové logiky

- asociativita \wedge a \vee :

$$x \wedge (y \wedge z) = (x \wedge y) \wedge z$$

$$x \vee (y \vee z) = (x \vee y) \vee z$$

- komutativita \wedge a \vee :

$$x \wedge y = y \wedge x$$

$$x \vee y = y \vee x$$

- distributivita \wedge vůči \vee , \vee vůči \wedge :

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$$

- absorpce:

$$x \wedge (x \vee y) = x$$

$$x \vee (x \wedge y) = x$$

- komplementace:

$$x \wedge (-x) = \perp$$

$$x \vee (-x) = \top$$

- netrivialita:

$$-(\perp = \top)$$

- asociativita \wedge a \vee :

$$x \wedge (y \wedge z) = (x \wedge y) \wedge z$$

$$x \vee (y \vee z) = (x \vee y) \vee z$$

- komutativita \wedge a \vee :

$$x \wedge y = y \wedge x$$

$$x \vee y = y \vee x$$

- distributivita \wedge vůči \vee , \vee vůči \wedge :

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$$

- absorpce:

$$x \wedge (x \vee y) = x$$

$$x \vee (x \wedge y) = x$$

- komplementace:

$$x \wedge (-x) = \perp$$

$$x \vee (-x) = \top$$

- netrivialita:

$$-(\perp = \top)$$

- dualita: záměnou \wedge s \vee a \perp s \top získáme tytéž axiomy

- asociativita \wedge a \vee :

$$x \wedge (y \wedge z) = (x \wedge y) \wedge z$$

$$x \vee (y \vee z) = (x \vee y) \vee z$$

- komutativita \wedge a \vee :

$$x \wedge y = y \wedge x$$

$$x \vee y = y \vee x$$

- distributivita \wedge vůči \vee , \vee vůči \wedge :

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$$

- absorpce:

$$x \wedge (x \vee y) = x$$

$$x \vee (x \wedge y) = x$$

- komplementace:

$$x \wedge (-x) = \perp$$

$$x \vee (-x) = \top$$

- netrivialita:

$$-(\perp = \top)$$

- dualita: záměnou \wedge s \vee a \perp s \top získáme tytéž axiomy
- nejmenší model: 2-prvková B. algebra $\langle \{0, 1\}, f_{\neg}, f_{\wedge}, f_{\vee}, 0, 1 \rangle$

- asociativita \wedge a \vee :

$$x \wedge (y \wedge z) = (x \wedge y) \wedge z$$

$$x \vee (y \vee z) = (x \vee y) \vee z$$

- komutativita \wedge a \vee :

$$x \wedge y = y \wedge x$$

$$x \vee y = y \vee x$$

- distributivita \wedge vůči \vee , \vee vůči \wedge :

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$$

- absorpce:

$$x \wedge (x \vee y) = x$$

$$x \vee (x \wedge y) = x$$

- komplementace:

$$x \wedge (-x) = \perp$$

$$x \vee (-x) = \top$$

- netrivialita:

$$-(\perp = \top)$$

- dualita: záměnou \wedge s \vee a \perp s \top získáme tytéž axiomy
- nejmenší model: 2-prvková B. algebra $\langle \{0, 1\}, f_{\neg}, f_{\wedge}, f_{\vee}, 0, 1 \rangle$
- konečné modely, až na izomorfismus (f^n je f po složkách):

$$\langle \{0, 1\}^n, f_{\neg}^n, f_{\wedge}^n, f_{\vee}^n, (0, \dots, 0), (1, \dots, 1) \rangle$$

- asociativita \wedge a \vee :

$$x \wedge (y \wedge z) = (x \wedge y) \wedge z$$

$$x \vee (y \vee z) = (x \vee y) \vee z$$

- komutativita \wedge a \vee :

$$x \wedge y = y \wedge x$$

$$x \vee y = y \vee x$$

- distributivita \wedge vůči \vee , \vee vůči \wedge :

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$$

- absorpce:

$$x \wedge (x \vee y) = x$$

$$x \vee (x \wedge y) = x$$

- komplementace:

$$x \wedge (-x) = \perp$$

$$x \vee (-x) = \top$$

- netrivialita:

$$-(\perp = \top)$$

- dualita: záměnou \wedge s \vee a \perp s \top získáme tytéž axiomy
- nejmenší model: 2-prvková B. algebra $\langle \{0, 1\}, f_{\neg}, f_{\wedge}, f_{\vee}, 0, 1 \rangle$
- konečné modely, až na izomorfismus (f^n je f po složkách):
$$\langle \{0, 1\}^n, f_{\neg}^n, f_{\wedge}^n, f_{\vee}^n, (0, \dots, 0), (1, \dots, 1) \rangle$$
- jsou izomorfní potenčním algebrám $\mathcal{P}(\{1, \dots, n\})$ pomocí bijekce mezi podmnožinami a charakteristickými vektory

- výrokovou logiku lze 'simulovat' v predikátové logice v teorii Booleových algeber

- výrokovou logiku lze 'simulovat' v predikátové logice v teorii Booleových algeber
- výroky jsou **Booleovské termy**, konstanty \perp , \top představují pravdu a lež

- výrokovou logiku lze 'simulovat' v predikátové logice v teorii Booleových algeber
- výroky jsou **Booleovské termy**, konstanty \perp , \top představují pravdu a lež
- pravdivostní hodnota výroku (při daném pravdivostním ohodnocení) je hodnota termu v 2-prvkové Booleově algebře

- výrokovou logiku lze 'simulovat' v predikátové logice v teorii Booleových algeber
- výroky jsou **Booleovské termy**, konstanty \perp , \top představují pravdu a lež
- pravdivostní hodnota výroku (při daném pravdivostním ohodnocení) je hodnota termu v 2-prvkové Booleově algebře
- kromě toho, **algebra výroků** daného výrokového jazyka nebo teorie je Booleovou algebrou (i pro nekonečné jazyky)

Na druhou stranu...

- máme-li **otevřenou** formuli φ (bez rovnosti), můžeme reprezentovat atomické formule pomocí prvovýroků, a získat tak výrok, který platí, právě když platí φ

- máme-li **otevřenou** formuli φ (bez rovnosti), můžeme reprezentovat atomické formule pomocí prvovýroků, a získat tak výrok, který platí, právě když platí φ
- viz Kapitola 8: Rezoluce v predikátové logice, kde se nejprve zbavíme kvantifikátorů pomocí tzv. **Skolemizace**

- máme-li **otevřenou** formuli φ (bez rovnosti), můžeme reprezentovat atomické formule pomocí prvovýroků, a získat tak výrok, který platí, právě když platí φ
- viz Kapitola 8: Rezoluce v predikátové logice, kde se nejprve zbavíme kvantifikátorů pomocí tzv. **Skolemizace**
- výrokovou logiku lze také zavést jako fragment logiky predikátové, pokud povolíme **nulární relace**

- máme-li **otevřenou** formuli φ (bez rovnosti), můžeme reprezentovat atomické formule pomocí prvovýroků, a získat tak výrok, který platí, právě když platí φ
- viz Kapitola 8: Rezoluce v predikátové logice, kde se nejprve zbavíme kvantifikátorů pomocí tzv. **Skolemizace**
- výrokovou logiku lze také zavést jako fragment logiky predikátové, pokud povolíme **nulární relace**
- $A^0 = \{\emptyset\}$, tedy na libovolné množině jsou právě dvě nulární relace $R^A \subseteq A^0$: $R^A = \emptyset = 0$ a $R^A = \{\emptyset\} = \{0\} = 1$

KAPITOLA 7: TABLO METODA V PREDIKÁTOVÉ LOGICE

7.1 Neformální úvod

Úvodní příklady: dva tablo důkazy

Úvodní příklady: dva tablo důkazy

$$F(\exists x)\neg P(x) \rightarrow \neg(\forall x)P(x)$$

$$T(\exists x)\neg P(x)$$

$$F\neg(\forall x)P(x)$$

$$T(\forall x)P(x)$$

$$T\neg P(c_0)$$

$$FP(c_0)$$

$$T(\forall x)P(x)$$

$$TP(c_0)$$



Úvodní příklady: dva tablo důkazy

$$F(\exists x)\neg P(x) \rightarrow \neg(\forall x)P(x)$$

$$\mid$$
$$T(\exists x)\neg P(x)$$

$$\mid$$
$$F\neg(\forall x)P(x)$$

$$\mid$$
$$T(\forall x)P(x)$$

$$\mid$$
$$T\neg P(c_0)$$

$$\mid$$
$$FP(c_0)$$

$$\mid$$
$$T(\forall x)P(x)$$

$$\mid$$
$$TP(c_0)$$

⊗

$$F\neg(\forall x)P(x) \rightarrow (\exists x)\neg P(x)$$

$$\mid$$
$$T\neg(\forall x)P(x)$$

$$\mid$$
$$F(\exists x)\neg P(x)$$

$$\mid$$
$$F(\forall x)P(x)$$

$$\mid$$
$$FP(c_0)$$

$$\mid$$
$$F(\exists x)\neg P(x)$$

$$\mid$$
$$F\neg P(c_0)$$

$$\mid$$
$$TP(c_0)$$

⊗

- opět vždy předpokládáme, že jazyk L je spočetný (nejprve bez rovnosti, později metodu rozšíříme pro rovnost)

Tablo metoda v predikátové logice

- opět vždy předpokládáme, že jazyk L je spočetný (nejprve bez rovnosti, později metodu rozšíříme pro rovnost)
- v položkách musí být **sentence**: pravdivostní hodnota nesmí záviset na ohodnocení (ale můžeme vzít **generální uzávěry**)

Tablo metoda v predikátové logice

- opět vždy předpokládáme, že jazyk L je spočetný (nejprve bez rovnosti, později metodu rozšíříme pro rovnost)
- v položkách musí být **sentence**: pravdivostní hodnota nesmí záviset na ohodnocení (ale můžeme vzít **generální uzávěry**)
- **redukce položek**: stejná atomická tabla pro logické spojky (kde φ, ψ jsou sentence), ale čtyři nové případy **pro kvantifikátory**:

Tablo metoda v predikátové logice

- opět vždy předpokládáme, že jazyk L je spočetný (nejprve bez rovnosti, později metodu rozšíříme pro rovnost)
- v položkách musí být **sentence**: pravdivostní hodnota nesmí záviset na ohodnocení (ale můžeme vzít **generální uzávěry**)
- **redukce položek**: stejná atomická tabla pro logické spojky (kde φ, ψ jsou sentence), ale čtyři nové případy **pro kvantifikátory**:
 - typ “**svědek**”: položky tvaru $T(\exists x)\varphi(x)$ a $F(\forall x)\varphi(x)$

Tablo metoda v predikátové logice

- opět vždy předpokládáme, že jazyk L je spočetný (nejprve bez rovnosti, později metodu rozšíříme pro rovnost)
- v položkách musí být **sentence**: pravdivostní hodnota nesmí záviset na ohodnocení (ale můžeme vzít **generální uzávěry**)
- **redukce položek**: stejná atomická tabla pro logické spojky (kde φ, ψ jsou sentence), ale čtyři nové případy **pro kvantifikátory**:
 - typ “**svědek**”: položky tvaru $T(\exists x)\varphi(x)$ a $F(\forall x)\varphi(x)$
 - typ “**všichni**”: položky tvaru $T(\forall x)\varphi(x)$ a $F(\exists x)\varphi(x)$

Tablo metoda v predikátové logice

- opět vždy předpokládáme, že jazyk L je spočetný (nejprve bez rovnosti, později metodu rozšíříme pro rovnost)
- v položkách musí být **sentence**: pravdivostní hodnota nesmí záviset na ohodnocení (ale můžeme vzít **generální uzávěry**)
- **redukce položek**: stejná atomická tabla pro logické spojky (kde φ, ψ jsou sentence), ale čtyři nové případy **pro kvantifikátory**:
 - typ “**svědek**”: položky tvaru $T(\exists x)\varphi(x)$ a $F(\forall x)\varphi(x)$
 - typ “**všichni**”: položky tvaru $T(\forall x)\varphi(x)$ a $F(\exists x)\varphi(x)$
- kvantifikátor nelze odstranit, $\varphi(x)$ by typicky nebyla sentence

Tablo metoda v predikátové logice

- opět vždy předpokládáme, že jazyk L je spočetný (nejprve bez rovnosti, později metodu rozšíříme pro rovnost)
- v položkách musí být **sentence**: pravdivostní hodnota nesmí záviset na ohodnocení (ale můžeme vzít **generální uzávěry**)
- **redukce položek**: stejná atomická tabla pro logické spojky (kde φ, ψ jsou sentence), ale čtyři nové případy **pro kvantifikátory**:
 - typ “**svědek**”: položky tvaru $T(\exists x)\varphi(x)$ a $F(\forall x)\varphi(x)$
 - typ “**všichni**”: položky tvaru $T(\forall x)\varphi(x)$ a $F(\exists x)\varphi(x)$
- kvantifikátor nelze odstranit, $\varphi(x)$ by typicky nebyla sentence
- místo toho za x **substituujeme konstantní term** t : $\varphi(x/t)$

Tablo metoda v predikátové logice

- opět vždy předpokládáme, že jazyk L je spočetný (nejprve bez rovnosti, později metodu rozšíříme pro rovnost)
- v položkách musí být **sentence**: pravdivostní hodnota nesmí záviset na ohodnocení (ale můžeme vzít **generální uzávěry**)
- **redukce položek**: stejná atomická tabla pro logické spojky (kde φ, ψ jsou sentence), ale čtyři nové případy **pro kvantifikátory**:
 - typ “**svědek**”: položky tvaru $T(\exists x)\varphi(x)$ a $F(\forall x)\varphi(x)$
 - typ “**všichni**”: položky tvaru $T(\forall x)\varphi(x)$ a $F(\exists x)\varphi(x)$
- kvantifikátor nelze odstranit, $\varphi(x)$ by typicky nebyla sentence
- místo toho za x **substituujeme konstantní term** t : $\varphi(x/t)$
- jaký? podle typu položky (“**svědek**” vs. “**všichni**”)

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C
- vždy máme k dispozici nový, dosud nepoužitý symbol $c \in C$

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C
- vždy máme k dispozici nový, dosud nepoužitý symbol $c \in C$
- **typ “svědek”**: dosadíme nový $c \in C$ (dosud na větvi není)

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C
- vždy máme k dispozici nový, dosud nepoužitý symbol $c \in C$
- **typ “svědek”**: dosadíme nový $c \in C$ (dosud na větvi není)
 - pro $\mathsf{T}(\exists x)\varphi(x)$ tedy máme $\mathsf{T}\varphi(x/c)$

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C
- vždy máme k dispozici nový, dosud nepoužitý symbol $c \in C$
- **typ “svědek”**: dosadíme nový $c \in C$ (dosud na větvi není)
 - pro $T(\exists x)\varphi(x)$ tedy máme $T\varphi(x/c)$
 - c hraje roli prvku, který položku ‘splňuje’

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C
- vždy máme k dispozici nový, dosud nepoužitý symbol $c \in C$
- **typ “svědek”**: dosadíme nový $c \in C$ (dosud na větvi není)
 - pro $\mathsf{T}(\exists x)\varphi(x)$ tedy máme $\mathsf{T}\varphi(x/c)$
 - c hraje roli prvku, který položku ‘splňuje’
- **typ “všichni”**: substituujeme libovolný konstantní L_C -term

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C
- vždy máme k dispozici nový, dosud nepoužitý symbol $c \in C$
- **typ “svědek”**: dosadíme nový $c \in C$ (dosud na větvi není)
 - pro $T(\exists x)\varphi(x)$ tedy máme $T\varphi(x/c)$
 - c hraje roli prvku, který položku ‘splňuje’
- **typ “všichni”**: substituujeme libovolný konstantní L_C -term
 - pro $T(\forall x)\varphi(x)$ tedy máme $T\varphi(x/t)$

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C
- vždy máme k dispozici nový, dosud nepoužitý symbol $c \in C$
- **typ “svědek”**: dosadíme nový $c \in C$ (dosud na větvi není)
 - pro $T(\exists x)\varphi(x)$ tedy máme $T\varphi(x/c)$
 - c hraje roli prvku, který položku ‘splňuje’
- **typ “všichni”**: substituujeme libovolný konstantní L_C -term
 - pro $T(\forall x)\varphi(x)$ tedy máme $T\varphi(x/t)$
 - bezesporná větev je dokončená jen pokud dosadíme všechny t (‘použijeme vše, co víme’)

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C
- vždy máme k dispozici nový, dosud nepoužitý symbol $c \in C$
- **typ “svědek”**: dosadíme nový $c \in C$ (dosud na větvi není)
 - pro $T(\exists x)\varphi(x)$ tedy máme $T\varphi(x/c)$
 - c hraje roli prvku, který položku ‘splňuje’
- **typ “všichni”**: substituujeme libovolný konstantní L_C -term
 - pro $T(\forall x)\varphi(x)$ tedy máme $T\varphi(x/t)$
 - bezesporná větev je dokončená jen pokud dosadíme všechny t (‘použijeme vše, co víme’)
- **konvence**: kořeny atomických tabel nekreslíme kromě položek typu “všichni” (po jednom dosazení ještě nejsme hotovi!)

Redukce položek s kvantifikátorem

- jazyk L rozšíříme o spočetně mnoho nových (pomocných) konstantních symbolů $C = \{c_0, c_1, c_2, \dots\}$, označíme L_C
- vždy máme k dispozici nový, dosud nepoužitý symbol $c \in C$
- **typ “svědek”**: dosadíme nový $c \in C$ (dosud na větvi není)
 - pro $T(\exists x)\varphi(x)$ tedy máme $T\varphi(x/c)$
 - c hraje roli prvku, který položku ‘splňuje’
- **typ “všichni”**: substituujeme libovolný konstantní L_C -term
 - pro $T(\forall x)\varphi(x)$ tedy máme $T\varphi(x/t)$
 - bezesporná větev je dokončená jen pokud dosadíme všechny t (‘použijeme vše, co víme’)
- **konvence**: kořeny atomických tabel nekreslíme kromě položek typu “všichni” (po jednom dosazení ještě nejsme hotovi!)
- **typický postup**: nejprve zredukujeme položky typu “svědek”, poté zjistíme, co ‘o svědcích říkají’ položky typu “všichni”

7.2 Formální definice

- buď L spočetný jazyk bez rovnosti.

- buď L **spočetný** jazyk **bez rovnosti**.
- označme L_C rozšíření L o spočetně mnoho nových **pomocných** konstantních symbolů $C = \{c_i \mid i \in \mathbb{N}\}$

- buď L **spočetný** jazyk **bez rovnosti**.
- označme L_C rozšíření L o spočetně mnoho nových **pomocných** konstantních symbolů $C = \{c_i \mid i \in \mathbb{N}\}$
- zvolme očíslování konstantních L_C -termů: $\{t_i \mid i \in \mathbb{N}\}$

- buď L **spočetný** jazyk **bez rovnosti**.
- označme L_C rozšíření L o spočetně mnoho nových **pomocných** konstantních symbolů $C = \{c_i \mid i \in \mathbb{N}\}$
- zvolme očíslování konstantních L_C -termů: $\{t_i \mid i \in \mathbb{N}\}$
- mějme nějakou L -teorii T a L -sentenci φ

- buď L **spočetný** jazyk **bez rovnosti**.
- označme L_C rozšíření L o spočetně mnoho nových **pomocných** konstantních symbolů $C = \{c_i \mid i \in \mathbb{N}\}$
- zvolme očíslování konstantních L_C -termů: $\{t_i \mid i \in \mathbb{N}\}$
- mějme nějakou L -teorii T a L -sentenci φ
- **položka** je nápis $T\varphi$ nebo $F\varphi$, kde φ je L_C -sentence

- buď L **spočetný** jazyk **bez rovnosti**.
- označme L_C rozšíření L o spočetně mnoho nových **pomocných** konstantních symbolů $C = \{c_i \mid i \in \mathbb{N}\}$
- zvolme očíslování konstantních L_C -termů: $\{t_i \mid i \in \mathbb{N}\}$
- mějme nějakou L -teorii T a L -sentenci φ
- **položka** je nápis $T\varphi$ nebo $F\varphi$, kde φ je L_C -sentence
- položky tvaru $T(\exists x)\varphi(x)$ a $F(\forall x)\varphi(x)$ jsou **typu “svědek”**

- buď L **spočetný** jazyk **bez rovnosti**.
- označme L_C rozšíření L o spočetně mnoho nových **pomocných** konstantních symbolů $C = \{c_i \mid i \in \mathbb{N}\}$
- zvolme očíslování konstantních L_C -termů: $\{t_i \mid i \in \mathbb{N}\}$
- mějme nějakou L -teorii T a L -sentenci φ
- **položka** je nápis $T\varphi$ nebo $F\varphi$, kde φ je L_C -sentence
- položky tvaru $T(\exists x)\varphi(x)$ a $F(\forall x)\varphi(x)$ jsou **typu** “**svědek**”
- položky tvaru $T(\forall x)\varphi(x)$ a $F(\exists x)\varphi(x)$ jsou **typu** “**všichni**”

- buď L **spočetný** jazyk **bez rovnosti**.
- označme L_C rozšíření L o spočetně mnoho nových **pomocných** konstantních symbolů $C = \{c_i \mid i \in \mathbb{N}\}$
- zvolme očíslování konstantních L_C -termů: $\{t_i \mid i \in \mathbb{N}\}$
- mějme nějakou L -teorii T a L -sentenci φ
- **položka** je nápis $T\varphi$ nebo $F\varphi$, kde φ je L_C -sentence
- položky tvaru $T(\exists x)\varphi(x)$ a $F(\forall x)\varphi(x)$ jsou **typu** “**svědek**”
- položky tvaru $T(\forall x)\varphi(x)$ a $F(\exists x)\varphi(x)$ jsou **typu** “**všichni**”
- **atomická tabla** jsou násl. položkami označované stromy:

Atomická tabla pro kvantifikátory

φ je libovolná L_C -sentence, x proměnná, t_i konstantní L_C -term, $c_i \in C$ je nový pomocný konstantní symbol (při konstrukci tabla nesměl dosud být na dané větvi)

	\forall	\exists
True	$\begin{array}{c} T(\forall x)\varphi(x) \\ \\ T\varphi(x/t_i) \end{array}$	$\begin{array}{c} T(\exists x)\varphi(x) \\ \\ T\varphi(x/c_i) \end{array}$
False	$\begin{array}{c} F(\forall x)\varphi(x) \\ \\ F\varphi(x/c_i) \end{array}$	$\begin{array}{c} F(\exists x)\varphi(x) \\ \\ F\varphi(x/t_i) \end{array}$

Atomická tabla pro logické spojky

φ a ψ jsou libovolné L_C -sentence

	\neg	\wedge	\vee	\rightarrow	\leftrightarrow
True	$T\neg\varphi$	$T\varphi \wedge \psi$ $T\varphi$	$T\varphi \vee \psi$ / \ $T\varphi$ $T\psi$	$T\varphi \rightarrow \psi$ / \ $F\varphi$ $T\psi$	$T\varphi \leftrightarrow \psi$ / \ $T\varphi$ $F\varphi$ $T\psi$ $F\psi$
	$F\varphi$	$T\psi$			
False	$F\neg\varphi$	$F\varphi \wedge \psi$ / \ $F\varphi$ $F\psi$	$F\varphi \vee \psi$ $F\varphi$ $F\psi$	$F\varphi \rightarrow \psi$ $T\varphi$ $F\psi$	$F\varphi \leftrightarrow \psi$ / \ $T\varphi$ $F\varphi$ $F\psi$ $T\psi$
	$T\varphi$				

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T
 - pro libovolnou položku P na libovolné větvi V můžeme na konec větve V připojit atomické tablo pro položku P

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T
 - pro libovolnou položku P na libovolné větvi V můžeme na konec větve V připojit atomické tablo pro položku P
je-li P typu “**svědek**”, můžeme použít jen $c_i \in C$, který dosud na V není (pro typ “**všichni**” lze použít lib. konst. L_C -term t_i)

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T
 - pro libovolnou položku P na libovolné větvi V můžeme na konec větve V připojit atomické tablo pro položku P
je-li P typu “svědek”, můžeme použít jen $c_i \in C$, který dosud na V není (pro typ “všichni” lze použít lib. konst. L_C -term t_i)
 - na konec libovolné větve můžeme připojit položku $T\alpha$ pro libovolný axiom $\alpha \in T$

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T
 - pro libovolnou položku P na libovolné větvi V můžeme na konec větve V připojit atomické tablo pro položku P
je-li P typu “**svědek**”, můžeme použít jen $c_i \in C$, který dosud na V není (pro typ “**všichni**” lze použít lib. konst. L_C -term t_i)
 - na konec libovolné větve můžeme připojit položku $T\alpha$ pro libovolný axiom $\alpha \in T$
- **tablo z teorie T** je buď konečné, nebo i nekonečné: v tom případě je spočetné a definujeme ho jako $\tau = \bigcup_{i \geq 0} \tau_i$, kde:

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T
 - pro libovolnou položku P na libovolné větvi V můžeme na konec větve V připojit atomické tablo pro položku P
je-li P typu “**svědek**”, můžeme použít jen $c_i \in C$, který dosud na V není (pro typ “**všichni**” lze použít lib. konst. L_C -term t_i)
 - na konec libovolné větve můžeme připojit položku $T\alpha$ pro libovolný axiom $\alpha \in T$
- **tablo z teorie T** je buď konečné, nebo i nekonečné: v tom případě je spočetné a definujeme ho jako $\tau = \bigcup_{i \geq 0} \tau_i$, kde:
 - τ_i jsou konečná tabla z T

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T
 - pro libovolnou položku P na libovolné větvi V můžeme na konec větve V připojit atomické tablo pro položku P
je-li P typu “svědek”, můžeme použít jen $c_i \in C$, který dosud na V není (pro typ “všichni” lze použít lib. konst. L_C -term t_i)
 - na konec libovolné větve můžeme připojit položku $T\alpha$ pro libovolný axiom $\alpha \in T$
- **tablo z teorie T** je buď konečné, nebo i nekonečné: v tom případě je spočetné a definujeme ho jako $\tau = \bigcup_{i \geq 0} \tau_i$, kde:
 - τ_i jsou konečná tabla z T
 - τ_0 je jednoprvkové tablo

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T
 - pro libovolnou položku P na libovolné větvi V můžeme na konec větve V připojit atomické tablo pro položku P
je-li P typu “**svědek**”, můžeme použít jen $c_i \in C$, který dosud na V není (pro typ “**všichni**” lze použít lib. konst. L_C -term t_i)
 - na konec libovolné větve můžeme připojit položku $T\alpha$ pro libovolný axiom $\alpha \in T$
- **tablo z teorie T** je buď konečné, nebo i nekonečné: v tom případě je spočetné a definujeme ho jako $\tau = \bigcup_{i \geq 0} \tau_i$, kde:
 - τ_i jsou konečná tabla z T
 - τ_0 je jednoprvkové tablo
 - τ_{i+1} vzniklo z τ_i v jednom kroku

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T
 - pro libovolnou položku P na libovolné větvi V můžeme na konec větve V připojit atomické tablo pro položku P
je-li P typu “svědek”, můžeme použít jen $c_i \in C$, který dosud na V není (pro typ “všichni” lze použít lib. konst. L_C -term t_i)
 - na konec libovolné větve můžeme připojit položku $T\alpha$ pro libovolný axiom $\alpha \in T$
- **tablo z teorie T** je buď konečné, nebo i nekonečné: v tom případě je spočetné a definujeme ho jako $\tau = \bigcup_{i \geq 0} \tau_i$, kde:
 - τ_i jsou konečná tabla z T
 - τ_0 je jednoprvkové tablo
 - τ_{i+1} vzniklo z τ_i v jednom kroku
- **tablo pro položku P** je tablo, které má položku P v kořeni

Formální definice tabla

- **konečné tablo z teorie T** je uspoř., položkami označ. strom zkonstruovaný aplikací konečně mnoha následujících pravidel:
 - jednoprvkový strom s libovolnou položkou je tablo z teorie T
 - pro libovolnou položku P na libovolné větvi V můžeme na konec větve V připojit atomické tablo pro položku P
je-li P typu “svědek”, můžeme použít jen $c_i \in C$, který dosud na V není (pro typ “všichni” lze použít lib. konst. L_C -term t_i)
 - na konec libovolné větve můžeme připojit položku $T\alpha$ pro libovolný axiom $\alpha \in T$
- **tablo z teorie T** je buď konečné, nebo i nekonečné: v tom případě je spočetné a definujeme ho jako $\tau = \bigcup_{i \geq 0} \tau_i$, kde:
 - τ_i jsou konečná tabla z T
 - τ_0 je jednoprvkové tablo
 - τ_{i+1} vzniklo z τ_i v jednom kroku
- **tablo pro položku P** je tablo, které má položku P v kořeni

konvence: kořen atom. tabla nezapisujeme není-li P typu “všichni”

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.
- Větev je **sporná**, pokud obsahuje položky $T\psi$ a $F\psi$ pro nějakou **sentenci** ψ , jinak je **bezesporná**.

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.
- Větev je **sporná**, pokud obsahuje položky $T\psi$ a $F\psi$ pro nějakou **sentenci** ψ , jinak je **bezesporná**.
- Tablo je **dokončené**, pokud je každá jeho větev dokončená.

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.
- Větev je **sporná**, pokud obsahuje položky $T\psi$ a $F\psi$ pro nějakou **sentenci** ψ , jinak je **bezesporná**.
- Tablo je **dokončené**, pokud je každá jeho větev dokončená.
- Větev je **dokončená**, pokud je sporná, nebo

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.
- Větev je **sporná**, pokud obsahuje položky $T\psi$ a $F\psi$ pro nějakou **sentenci** ψ , jinak je **bezesporná**.
- Tablo je **dokončené**, pokud je každá jeho větev dokončená.
- Větev je **dokončená**, pokud je sporná, nebo
 - každá její položka je na této větvi **redukována**,

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.
- Větev je **sporná**, pokud obsahuje položky $T\psi$ a $F\psi$ pro nějakou **sentenci** ψ , jinak je **bezesporná**.
- Tablo je **dokončené**, pokud je každá jeho větev dokončená.
- Větev je **dokončená**, pokud je sporná, nebo
 - každá její položka je na této větvi **redukována**,
 - a zároveň obsahuje položku $T\alpha$ pro každý axiom $\alpha \in T$.

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.
- Větev je **sporná**, pokud obsahuje položky $T\psi$ a $F\psi$ pro nějakou **sentenci** ψ , jinak je **bezesporná**.
- Tablo je **dokončené**, pokud je každá jeho větev dokončená.
- Větev je **dokončená**, pokud je sporná, nebo
 - každá její položka je na této větvi **redukována**,
 - a zároveň obsahuje položku $T\alpha$ pro každý axiom $\alpha \in T$.
- Položka P je **redukována** na větvi V procházející P , pokud

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.
- Větev je **sporná**, pokud obsahuje položky $T\psi$ a $F\psi$ pro nějakou **sentenci** ψ , jinak je **bezesporná**.
- Tablo je **dokončené**, pokud je každá jeho větev dokončená.
- Větev je **dokončená**, pokud je sporná, nebo
 - každá její položka je na této větvi **redukována**,
 - a zároveň obsahuje položku $T\alpha$ pro každý axiom $\alpha \in T$.
- Položka P je **redukována** na větvi V procházející P , pokud
 - je tvaru $T\psi$ resp. $F\psi$ pro **atomickeou sentenci**, nebo

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.
- Větev je **sporná**, pokud obsahuje položky $T\psi$ a $F\psi$ pro nějakou **sentenci** ψ , jinak je **bezesporná**.
- Tablo je **dokončené**, pokud je každá jeho větev dokončená.
- Větev je **dokončená**, pokud je sporná, nebo
 - každá její položka je na této větvi **redukována**,
 - a zároveň obsahuje položku $T\alpha$ pro každý axiom $\alpha \in T$.
- Položka P je **redukována** na větvi V procházející P , pokud
 - je tvaru $T\psi$ resp. $F\psi$ pro **atomickou sentenci**, nebo
 - **není typu "všichni"** a vyskytuje se na V jako kořen atomického tabla (tj., typicky, již došlo k jejímu rozvoji na V), nebo

Dokončené a sporné tablo

- Tablo je **sporné**, pokud je každá jeho větev sporná.
- Větev je **sporná**, pokud obsahuje položky $T\psi$ a $F\psi$ pro nějakou **sentenci** ψ , jinak je **bezesporná**.
- Tablo je **dokončené**, pokud je každá jeho větev dokončená.
- Větev je **dokončená**, pokud je sporná, nebo
 - každá její položka je na této větvi **redukována**,
 - a zároveň obsahuje položku $T\alpha$ pro každý axiom $\alpha \in T$.
- Položka P je **redukována** na větvi V procházející P , pokud
 - je tvaru $T\psi$ resp. $F\psi$ pro **atomickou sentenci**, nebo
 - není typu "**všichni**" a vyskytuje se na V jako kořen atomického tabla (tj., typicky, již došlo k jejímu rozvoji na V), nebo
 - je typu "**všichni**" a všechny její **výskyty** na větvi V jsou na V **redukovány**.

Kdy je výskyt položky typu “všichni” redukovaný?

Kdy je výskyt položky typu “všichni” redukováný?

Výskyt položky P typu “všichni” na V je i -tý, má-li právě $i - 1$ předků označených P , a i -tý výskyt je redukováný na V , pokud

Kdy je výskyt položky typu “všichni” redukováný?

Výskyt položky P typu “všichni” na V je i -tý, má-li právě $i - 1$ předků označených P , a i -tý výskyt je redukováný na V , pokud

- P má $(i + 1)$ -ní výskyt na V , a zároveň

Kdy je výskyt položky typu “všichni” redukováný?

Výskyt položky P typu “všichni” na V je i -tý, má-li právě $i - 1$ předků označených P , a i -tý výskyt je redukováný na V , pokud

- P má $(i + 1)$ -ní výskyt na V , a zároveň
- na V je položka $\mathsf{T}\varphi(x/t_i)$ (je-li $P = \mathsf{T}(\forall x)\varphi(x)$) resp. $\mathsf{F}\varphi(x/t_i)$ (je-li $P = \mathsf{F}(\exists x)\varphi(x)$), kde t_i je i -tý konstantní L_C -term (tj., typicky, už jsme za x substituovali t_i)

Kdy je výskyt položky typu “všichni” redukováný?

Výskyt položky P typu “všichni” na V je i -tý, má-li právě $i - 1$ předků označených P , a i -tý výskyt je redukováný na V , pokud

- P má $(i + 1)$ -ní výskyt na V , a zároveň
- na V je položka $\mathsf{T}\varphi(x/t_i)$ (je-li $P = \mathsf{T}(\forall x)\varphi(x)$) resp. $\mathsf{F}\varphi(x/t_i)$ (je-li $P = \mathsf{F}(\exists x)\varphi(x)$), kde t_i je i -tý konstantní L_C -term (tj., typicky, už jsme za x substituovali t_i)

NB: je-li položka typu “všichni” na V redukována, má na V nekonečně výskytů, a dosadili jsme všechny konstantní L_C -termy

- **tablo důkaz** sentence φ z teorie T je **sporné** tablo z teorie T s položkou $F\varphi$ v kořeni

- **tablo důkaz** sentence φ z teorie T je **sporné** tablo z teorie T s položkou $F\varphi$ v kořeni
- pokud existuje, je φ **(tablo) dokazatelný** z T , píšeme $T \vdash \varphi$

- **tablo důkaz** sentence φ z teorie T je **sporné** tablo z teorie T s položkou $F\varphi$ v kořeni
- pokud existuje, je φ **(tablo) dokazatelný** z T , píšeme $T \vdash \varphi$
- podobně, **tablo zamítnutí** je sporné tablo s $T\varphi$ v kořeni

- **tablo důkaz** sentence φ z teorie T je **sporné** tablo z teorie T s položkou $F\varphi$ v kořeni
- pokud existuje, je φ **(tablo) dokazatelný** z T , píšeme $T \vdash \varphi$
- podobně, **tablo zamítnutí** je sporné tablo s $T\varphi$ v kořeni
- existuje-li, je φ **(tablo) zamítnutelný** z T , tj. platí $T \vdash \neg\varphi$

Příklad: tablo důkaz (v logice)

$$F(\forall x)(P(x) \rightarrow Q(x)) \rightarrow ((\forall x)P(x) \rightarrow (\forall x)Q(x))$$

$$T(\forall x)(P(x) \rightarrow Q(x))$$

$$F(\forall x)P(x) \rightarrow (\forall x)Q(x)$$

$$T(\forall x)P(x)$$

$$F(\forall x)Q(x)$$

$$FQ(c_0)$$

$$T(\forall x)P(x)$$

$$TP(c_0)$$

$$T(\forall x)(P(x) \rightarrow Q(x))$$

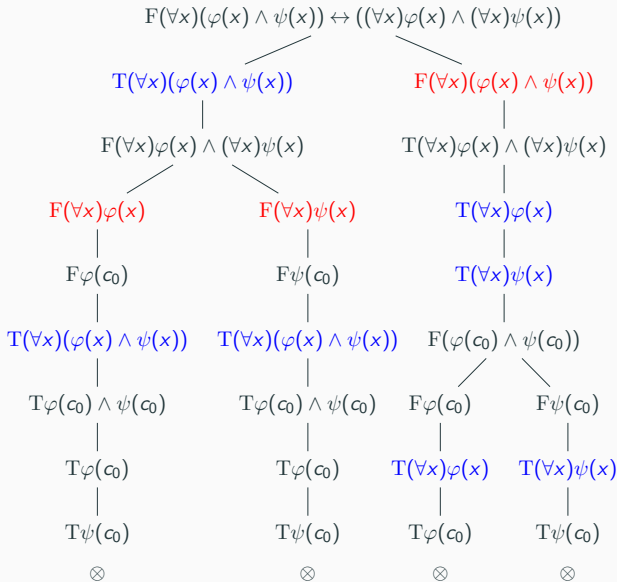
$$TP(c_0) \rightarrow Q(c_0)$$

$$FP(c_0)$$

$$TQ(c_0)$$



Ještě příklad (φ, ψ jsou formule s jedinou volnou proměnnou x)



(c_0 lze použít jako **nový** ve všech případech: **na dané větvi** se dosud nevyskytuje)

Systematické tablo

musí někdy zredukovat každou položku, použít každý axiom, a nově ve všech položkách typu “**všichni**” dosadit každý L_C term t_i

Systematické tablo

musí někdy zredukovat každou položku, použít každý axiom, a nově ve všech položkách typu “všichni” dosadit každý L_C term t_i

Systematické tablo z $T = \{\alpha_0, \alpha_1, \alpha_2, \dots\}$ pro položku R je $\tau = \bigcup_{i \geq 0} \tau_i$, kde τ_0 je jednoprvkové s položkou R , a pro $i \geq 0$:

Systematické tablo

musí někdy zredukovat každou položku, použít každý axiom, a nově ve všech položkách typu “všichni” dosadit každý L_C term t_i

Systematické tablo z $T = \{\alpha_0, \alpha_1, \alpha_2, \dots\}$ pro položku R je $\tau = \bigcup_{i \geq 0} \tau_i$, kde τ_0 je jednoprvkové s položkou R , a pro $i \geq 0$:

- buď P nejlevější položka v co nejmenší úrovni, která není redukována na nějaké bezesporné větvi procházející P (resp. je-li typu “všichni”, její výskyt není redukován)

Systematické tablo

musí někdy zredukovat každou položku, použít každý axiom, a nově ve všech položkách typu “všichni” dosadit každý L_C term t_i

Systematické tablo z $T = \{\alpha_0, \alpha_1, \alpha_2, \dots\}$ pro položku R je $\tau = \bigcup_{i \geq 0} \tau_i$, kde τ_0 je jednoprvkové s položkou R , a pro $i \geq 0$:

- buď P nejlevější položka v co nejmenší úrovni, která není redukována na nějaké bezesporné větvi procházející P (resp. je-li typu “všichni”, její výskyt není redukován)
- nejprve definujeme τ'_i vzniklé z τ_i připojením atomického tabla pro P na každou bezespornou větev procházející P , kde

Systematické tablo

musí někdy zredukovat každou položku, použít každý axiom, a nově ve všech položkách typu “**všichni**” dosadit každý L_C term t_i

Systematické tablo z $T = \{\alpha_0, \alpha_1, \alpha_2, \dots\}$ pro položku R je $\tau = \bigcup_{i \geq 0} \tau_i$, kde τ_0 je jednoprvkové s položkou R , a pro $i \geq 0$:

- buď P nejlevější položka v co nejmenší úrovni, která není redukována na nějaké bezesporné větvi procházející P (resp. je-li typu “**všichni**”, její **výskyt** není redukováný)
- nejprve definujeme τ'_i vzniklé z τ_i připojením atomického tabla pro P na každou bezespornou větev procházející P , kde je-li P typu “**všichni**” a má-li ve vrcholu k -tý výskyt, dosadíme k -tý L_C -term t_k , je-li typu “**svědek**”, substituujeme $c_i \in C$ s nejmenším i , které na větvi zatím není

Systematické tablo

musí někdy zredukovat každou položku, použít každý axiom, a nově ve všech položkách typu “**všichni**” dosadit každý L_C term t_i

Systematické tablo z $T = \{\alpha_0, \alpha_1, \alpha_2, \dots\}$ pro položku R je $\tau = \bigcup_{i \geq 0} \tau_i$, kde τ_0 je jednoprvkové s položkou R , a pro $i \geq 0$:

- buď P nejlevější položka v co nejmenší úrovni, která není redukována na nějaké bezesporné větvi procházející P (resp. je-li typu “**všichni**”, její **výskyt** není redukováný)
- nejprve definujeme τ'_i vzniklé z τ_i připojením atomického tabla pro P na každou bezespornou větev procházející P , kde je-li P typu “**všichni**” a má-li ve vrcholu k -tý výskyt, dosadíme k -tý L_C -term t_k , je-li typu “**svědek**”, substituujeme $c_i \in C$ s nejmenším i , které na větvi zatím není
- pokud taková položka P neexistuje, potom $\tau'_i = \tau_i$

Systematické tablo

musí někdy zredukovat každou položku, použít každý axiom, a nově ve všech položkách typu “**všichni**” dosadit každý L_C term t_i

Systematické tablo z $T = \{\alpha_0, \alpha_1, \alpha_2, \dots\}$ pro položku R je $\tau = \bigcup_{i \geq 0} \tau_i$, kde τ_0 je jednoprvkové s položkou R , a pro $i \geq 0$:

- buď P nejlevější položka v co nejmenší úrovni, která není redukována na nějaké bezesporné větvi procházející P (resp. je-li typu “**všichni**”, její **výskyt** není redukován)
- nejprve definujeme τ'_i vzniklé z τ_i připojením atomického tabla pro P na každou bezespornou větev procházející P , kde je-li P typu “**všichni**” a má-li ve vrcholu k -tý výskyt, dosadíme k -tý L_C -term t_k , je-li typu “**svědek**”, substituujeme $c_i \in C$ s nejmenším i , které na větvi zatím není
- pokud taková položka P neexistuje, potom $\tau'_i = \tau_i$
- τ_{i+1} vznikne z τ'_i připojením $T\alpha_{i+1}$ na vš. bezesporné větve (pokud už jsme použili všechny axiomy, definujeme $\tau_{i+1} = \tau'_i$)

Lemma: Systematické tablo je dokončené.

Lemma: Systematické tablo je dokončené.

Důkaz: k -tý výskyt položky typu “**všichni**” redukuje se když na něj narazíme: připojíme $(k + 1)$ -ní výskyt a dosadíme k -tý L_C -term t_k . Zbytek důkazu jako ve výrokové logice. \square

Konečnost a systematicčnost důkazů

Lemma: Systematické tablo je dokončené.

Důkaz: k -tý výskyt položky typu “**všichni**” redukuje se když na něj narazíme: připojíme $(k + 1)$ -ní výskyt a dosadíme k -tý L_C -term t_k . Zbytek důkazu jako ve výrokové logice. \square

Neprodlužujeme-li sporné větve (což nemusíme), je sporné tablo vždy konečné. Důkaz stejný jako ve výrokové logice:

Konečnost a systematicčnost důkazů

Lemma: Systematické tablo je dokončené.

Důkaz: k -tý výskyt položky typu “**všichni**” redukuje se když na něj narazíme: připojíme $(k + 1)$ -ní výskyt a dosadíme k -tý L_C -term t_k . Zbytek důkazu jako ve výrokové logice. \square

Neprodlužujeme-li sporné větve (což nemusíme), je sporné tablo vždy konečné. Důkaz stejný jako ve výrokové logice:

Důsledek (Konečnost důkazů): Pokud $T \vdash \varphi$, potom existuje i konečný tablo důkaz φ z T .

Konečnost a systematicčnost důkazů

Lemma: Systematické tablo je dokončené.

Důkaz: k -tý výskyt položky typu “**všichni**” redukuje se když na něj narazíme: připojíme $(k + 1)$ -ní výskyt a dosadíme k -tý L_C -term t_k . Zbytek důkazu jako ve výrokové logice. \square

Neprodlužujeme-li sporné větve (což nemusíme), je sporné tablo vždy konečné. Důkaz stejný jako ve výrokové logice:

Důsledek (Konečnost důkazů): Pokud $T \vdash \varphi$, potom existuje i konečný tablo důkaz φ z T .

Stejně jako ve výrokové logice z důkazu plyne:

Konečnost a systematicčnost důkazů

Lemma: Systematické tablo je dokončené.

Důkaz: k -tý výskyt položky typu “všichni” redukuje se když na něj narazíme: připojíme $(k + 1)$ -ní výskyt a dosadíme k -tý L_C -term t_k . Zbytek důkazu jako ve výrokové logice. \square

Neprodlužujeme-li sporné větve (což nemusíme), je sporné tablo vždy konečné. Důkaz stejný jako ve výrokové logice:

Důsledek (Konečnost důkazů): Pokud $T \vdash \varphi$, potom existuje i konečný tablo důkaz φ z T .

Stejně jako ve výrokové logice z důkazu plyne:

Důsledek (Systematicčnost důkazů): Pokud $T \vdash \varphi$, potom systematické tablo je (konečným) tablo důkazem φ z T .

7.3 Jazyky s rovností

$1 + 0 = 0 + 1$? identita celých čísel, výrazů, množin,
unifikovatelnost termů (v Prologu), ...

$1 + 0 = 0 + 1$? identita celých čísel, výrazů, množin,
unifikovatelnost termů (v Prologu), ...

Tablo je čistě **syntaktický** objekt, ale $=^A$ má být **identita** na A . Jak toho docílit?

$1 + 0 = 0 + 1$? identita celých čísel, výrazů, množin,
unifikovatelnost termů (v Prologu), ...

Tablo je čistě **syntaktický** objekt, ale $=^A$ má být **identita** na A . Jak toho docílit?

Mějme dokončenou bezespornou větev tabla s položkou $Tc_1 = c_2$.
V **kanonickém modelu** musí platit nejen $(c_1^A, c_2^A) \in =^A$, ale také:

$1 + 0 = 0 + 1$? identita celých čísel, výrazů, množin,
unifikovatelnost termů (v Prologu), ...

Tablo je čistě **syntaktický** objekt, ale $=^A$ má být **identita** na A . Jak toho docílit?

Mějme dokončenou bezespornou větev tabla s položkou $Tc_1 = c_2$.

V **kanonickém modelu** musí platit nejen $(c_1^A, c_2^A) \in =^A$, ale také:

- $c_2^A =^A c_1^A$

$1 + 0 = 0 + 1$? identita celých čísel, výrazů, množin,
unifikovatelnost termů (v Prologu), ...

Tablo je čistě **syntaktický** objekt, ale $=^A$ má být **identita** na A . Jak toho docílit?

Mějme dokončenou bezespornou větev tabla s položkou $Tc_1 = c_2$.

V **kanonickém modelu** musí platit nejen $(c_1^A, c_2^A) \in =^A$, ale také:

- $c_2^A =^A c_1^A$
- $f^A(c_1^A) =^A f^A(c_2^A)$

$1 + 0 = 0 + 1$? identita celých čísel, výrazů, množin,
unifikovatelnost termů (v Prologu), ...

Tablo je čistě **syntaktický** objekt, ale $=^A$ má být **identita** na A . Jak toho docílit?

Mějme dokončenou bezespornou větev tabla s položkou $Tc_1 = c_2$.

V **kanonickém modelu** musí platit nejen $(c_1^A, c_2^A) \in =^A$, ale také:

- $c_2^A =^A c_1^A$
- $f^A(c_1^A) =^A f^A(c_2^A)$
- $c_1^A \in P^A$ právě když $c_2^A \in P^A$

$1 + 0 = 0 + 1$? identita celých čísel, výrazů, množin,
unifikovatelnost termů (v Prologu), ...

Tablo je čistě **syntaktický** objekt, ale $=^A$ má být **identita** na A . Jak toho docílit?

Mějme dokončenou bezespornou větev tabla s položkou $Tc_1 = c_2$.

V **kanonickém modelu** musí platit nejen $(c_1^A, c_2^A) \in =^A$, ale také:

- $c_2^A =^A c_1^A$
- $f^A(c_1^A) =^A f^A(c_2^A)$
- $c_1^A \in P^A$ právě když $c_2^A \in P^A$

To vynutíme přidáním **axiomů rovnosti**, $=^A$ bude **kongruence** \mathcal{A} (ekvivalence, která se chová dobře k funkcím a relacím).

$1 + 0 = 0 + 1$? identita celých čísel, výrazů, množin,
unifikovatelnost termů (v Prologu), ...

Tablo je čistě **syntaktický** objekt, ale $=^A$ má být **identita** na A . Jak toho docílit?

Mějme dokončenou bezespornou větev tabla s položkou $Tc_1 = c_2$.
V **kanonickém modelu** musí platit nejen $(c_1^A, c_2^A) \in =^A$, ale také:

- $c_2^A =^A c_1^A$
- $f^A(c_1^A) =^A f^A(c_2^A)$
- $c_1^A \in P^A$ právě když $c_2^A \in P^A$

To vynutíme přidáním **axiomů rovnosti**, $=^A$ bude **kongruence** \mathcal{A} (ekvivalence, která se chová dobře k funkcím a relacím).

Poté vezmeme **faktorstrukturu** $\mathcal{B} = \mathcal{A}/_{=^A}$, v ní už je $=^B$ **identita**.

Bud' \sim ekvivalence na A , $f: A^n \rightarrow A$, $R \subseteq A^n$. Říkáme, že \sim je:

Kongruence a faktorstruktura

Bud' \sim ekvivalence na A , $f: A^n \rightarrow A$, $R \subseteq A^n$. Říkáme, že \sim je:

- **kongruence pro f** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $f(a_1, \dots, a_n) \sim f(b_1, \dots, b_n)$

Kongruence a faktorstruktura

Bud' \sim ekvivalence na A , $f: A^n \rightarrow A$, $R \subseteq A^n$. Říkáme, že \sim je:

- **kongruence pro f** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $f(a_1, \dots, a_n) \sim f(b_1, \dots, b_n)$
- **kongruence pro R** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $R(a_1, \dots, a_n) \Leftrightarrow R(b_1, \dots, b_n)$

Kongruence a faktorstruktura

Bud' \sim ekvivalence na A , $f: A^n \rightarrow A$, $R \subseteq A^n$. Říkáme, že \sim je:

- **kongruence pro f** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $f(a_1, \dots, a_n) \sim f(b_1, \dots, b_n)$
- **kongruence pro R** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $R(a_1, \dots, a_n) \Leftrightarrow R(b_1, \dots, b_n)$

Kongruence struktury \mathcal{A} je ekvivalence na A , která je kongruencí pro všechny funkce a relace \mathcal{A} .

Kongruence a faktorstruktura

Bud' \sim ekvivalence na A , $f: A^n \rightarrow A$, $R \subseteq A^n$. Říkáme, že \sim je:

- **kongruence pro f** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $f(a_1, \dots, a_n) \sim f(b_1, \dots, b_n)$
- **kongruence pro R** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $R(a_1, \dots, a_n) \Leftrightarrow R(b_1, \dots, b_n)$

Kongruence struktury \mathcal{A} je ekvivalence na A , která je kongruencí pro všechny funkce a relace \mathcal{A} .

Faktorstruktura (podílová struktura) \mathcal{A} podle \sim je struktura \mathcal{A}/\sim v témž jazyce, doména A/\sim je množina všech rozkladových tříd A podle \sim , funkce a relace definujeme **pomocí reprezentantů**:

Kongruence a faktorstruktura

Bud' \sim ekvivalence na A , $f: A^n \rightarrow A$, $R \subseteq A^n$. Říkáme, že \sim je:

- **kongruence pro f** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $f(a_1, \dots, a_n) \sim f(b_1, \dots, b_n)$
- **kongruence pro R** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $R(a_1, \dots, a_n) \Leftrightarrow R(b_1, \dots, b_n)$

Kongruence struktury \mathcal{A} je ekvivalence na A , která je kongruencí pro všechny funkce a relace \mathcal{A} .

Faktorstruktura (podílová struktura) \mathcal{A} podle \sim je struktura \mathcal{A}/\sim v témž jazyce, doména A/\sim je množina všech rozkladových tříd A podle \sim , funkce a relace definujeme **pomocí reprezentantů**:

- $f^{\mathcal{A}/\sim}([a_1]_\sim, \dots, [a_n]_\sim) = [f^{\mathcal{A}}(a_1, \dots, a_n)]_\sim$

Kongruence a faktorstruktura

Bud' \sim ekvivalence na A , $f: A^n \rightarrow A$, $R \subseteq A^n$. Říkáme, že \sim je:

- **kongruence pro f** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $f(a_1, \dots, a_n) \sim f(b_1, \dots, b_n)$
- **kongruence pro R** , pokud pro všechna $a_i, b_i \in A$ taková, že $a_i \sim b_i$ ($1 \leq i \leq n$), platí $R(a_1, \dots, a_n) \Leftrightarrow R(b_1, \dots, b_n)$

Kongruence struktury \mathcal{A} je ekvivalence na A , která je kongruencí pro všechny funkce a relace \mathcal{A} .

Faktorstruktura (podílová struktura) \mathcal{A} podle \sim je struktura \mathcal{A}/\sim v témž jazyce, doména A/\sim je množina všech rozkladových tříd A podle \sim , funkce a relace definujeme **pomocí reprezentantů**:

- $f^{\mathcal{A}/\sim}([a_1]_\sim, \dots, [a_n]_\sim) = [f^{\mathcal{A}}(a_1, \dots, a_n)]_\sim$
- $R^{\mathcal{A}/\sim}([a_1]_\sim, \dots, [a_n]_\sim) \Leftrightarrow R^{\mathcal{A}}(a_1, \dots, a_n)$

Axiomy rovnosti

Axiomy rovnosti pro jazyk L s rovností:

Axiomy rovnosti

Axiomy rovnosti pro jazyk L s rovností:

(i) $x = x$

Axiomy rovnosti pro jazyk L s rovností:

(i) $x = x$

(ii) pro každý n -ární funkční symbol f jazyka L :

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow f(x_1, \dots, x_n) = f(y_1, \dots, y_n)$$

Axiomy rovnosti

Axiomy rovnosti pro jazyk L s rovností:

(i) $x = x$

(ii) pro každý n -ární funkční symbol f jazyka L :

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow f(x_1, \dots, x_n) = f(y_1, \dots, y_n)$$

(iii) pro každý n -ární relační symbol R jazyka L včetně rovnosti:

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow (R(x_1, \dots, x_n) \rightarrow R(y_1, \dots, y_n))$$

Axiomy rovnosti

Axiomy rovnosti pro jazyk L s rovností:

(i) $x = x$

(ii) pro každý n -ární funkční symbol f jazyka L :

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow f(x_1, \dots, x_n) = f(y_1, \dots, y_n)$$

(iii) pro každý n -ární relační symbol R jazyka L včetně rovnosti:

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow (R(x_1, \dots, x_n) \rightarrow R(y_1, \dots, y_n))$$

- symetrie a tranzitivita plynou z (iii) pro $=$ (dokažte si)

Axiomy rovnosti

Axiomy rovnosti pro jazyk L s rovností:

(i) $x = x$

(ii) pro každý n -ární funkční symbol f jazyka L :

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow f(x_1, \dots, x_n) = f(y_1, \dots, y_n)$$

(iii) pro každý n -ární relační symbol R jazyka L **včetně rovnosti**:

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow (R(x_1, \dots, x_n) \rightarrow R(y_1, \dots, y_n))$$

- symetrie a tranzitivita plynou z (iii) pro $=$ (dokažte si)
- z axiomů (i) a (iii) tedy plyne, že relace $=^A$ je ekvivalence

Axiomy rovnosti pro jazyk L s rovností:

(i) $x = x$

(ii) pro každý n -ární funkční symbol f jazyka L :

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow f(x_1, \dots, x_n) = f(y_1, \dots, y_n)$$

(iii) pro každý n -ární relační symbol R jazyka L včetně rovnosti:

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow (R(x_1, \dots, x_n) \rightarrow R(y_1, \dots, y_n))$$

- symetrie a tranzitivita plynou z (iii) pro $=$ (dokažte si)
- z axiomů (i) a (iii) tedy plyne, že relace $=^A$ je ekvivalence
- axiomy (ii) a (iii) vyjadřují, že $=^A$ je kongruence

Axiomy rovnosti

Axiomy rovnosti pro jazyk L s rovností:

(i) $x = x$

(ii) pro každý n -ární funkční symbol f jazyka L :

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow f(x_1, \dots, x_n) = f(y_1, \dots, y_n)$$

(iii) pro každý n -ární relační symbol R jazyka L **včetně rovnosti**:

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow (R(x_1, \dots, x_n) \rightarrow R(y_1, \dots, y_n))$$

- symetrie a tranzitivita plynou z (iii) pro $=$ (dokažte si)
- z axiomů (i) a (iii) tedy plyne, že relace $=^A$ je ekvivalence
- axiomy (ii) a (iii) vyjadřují, že $=^A$ je kongruence

V tablo metodě pro jazyk s rovností implicitně přidáme axiomy rovnosti (přesněji jejich generální uzávěry, potřebujeme sentence).

Tablo důkaz s rovností

Tablo důkaz s rovností

Je-li T teorie v jazyce L s rovností, označme jako T^* rozšíření T o generální uzávěry axiomů rovnosti pro L .

Tablo důkaz s rovností

Je-li T teorie v jazyce L s rovností, označme jako T^* rozšíření T o generální uzávěry axiomů rovnosti pro L .

- **tablo důkaz** z teorie T je **tablo důkaz** z T^*

Tablo důkaz s rovností

Je-li T teorie v jazyce L s rovností, označme jako T^* rozšíření T o generální uzávěry axiomů rovnosti pro L .

- **tablo důkaz** z teorie T je **tablo důkaz** z T^*
- podobně pro tablo zamítnutí, a obecně jakékoliv tablo z T

Tablo důkaz s rovností

Je-li T teorie v jazyce L s rovností, označme jako T^* rozšíření T o generální uzávěry axiomů rovnosti pro L .

- **tablo důkaz** z teorie T je **tablo důkaz** z T^*
- podobně pro tablo zamítnutí, a obecně jakékoliv tablo z T

Pozorování:

Tablo důkaz s rovností

Je-li T teorie v jazyce L s rovností, označme jako T^* rozšíření T o generální uzávěry axiomů rovnosti pro L .

- **tablo důkaz** z teorie T je **tablo důkaz** z T^*
- podobně pro tablo zamítnutí, a obecně jakékoliv tablo z T

Pozorování:

- Je-li $\mathcal{A} \models T^*$, potom i $\mathcal{A}/_{=\mathcal{A}} \models T^*$, a ve struktuře $\mathcal{A}/_{=\mathcal{A}}$ je symbol rovnosti interpretován jako identita.

Tablo důkaz s rovností

Je-li T teorie v jazyce L s rovností, označme jako T^* rozšíření T o generální uzávěry axiomů rovnosti pro L .

- **tablo důkaz** z teorie T je **tablo důkaz** z T^*
- podobně pro tablo zamítnutí, a obecně jakékoliv tablo z T

Pozorování:

- Je-li $\mathcal{A} \models T^*$, potom i $\mathcal{A}/_{=\mathcal{A}} \models T^*$, a ve struktuře $\mathcal{A}/_{=\mathcal{A}}$ je symbol rovnosti interpretován jako identita.
- Na druhou stranu, v každém modelu, ve kterém je symbol rovnosti interpretován jako identita, platí axiomy rovnosti.

Tablo důkaz s rovností

Je-li T teorie v jazyce L s rovností, označme jako T^* rozšíření T o generální uzávěry axiomů rovnosti pro L .

- **tablo důkaz** z teorie T je **tablo důkaz** z T^*
- podobně pro tablo zamítnutí, a obecně jakékoliv tablo z T

Pozorování:

- Je-li $\mathcal{A} \models T^*$, potom i $\mathcal{A}/_{=\mathcal{A}} \models T^*$, a ve struktuře $\mathcal{A}/_{=\mathcal{A}}$ je symbol rovnosti interpretován jako identita.
- Na druhou stranu, v každém modelu, ve kterém je symbol rovnosti interpretován jako identita, platí axiomy rovnosti.

(Použijeme při konstrukci **kanonického modelu** v důkazu úplnosti.)