

První přednáška

NAIL062 Výroková a predikátová logika

Jakub Bulín (KTIML MFF UK)

Zimní semestr 2024

Velmi doporučuji tento online minikurz o efektivním učení:

<https://www.samford.edu/departments/academic-success-center/how-to-study>

Investujte 35 minut nyní, ušetřete mnoho hodin později!

Cesta k jistému úspěchu u zkoušky

- Před přednáškou alespoň zběžně projděte skripta, snažte se pochopit motivaci a smysl definic a hlavních tvrzení.
- Po přednášce skripta podrobně přečtěte, nejasnosti ujasněte.
- Ujistěte se, že umíte pracovat i s formalizmem.
- Věnujte pozornost i cvičení, pomůže vám vše pochopit.
- Studujte průběžně, a průběžně testujte své znalosti.

Program

- úvod do logiky
- neformální představení výrokové a predikátové logiky (“upoutávka”)
- syntaxe výrokové logiky
- sémantika výrokové logiky (začátek)

Materiály

Zápisky z přednášky, Kapitola 1 a Sekce 2.1-2.2.4 z Kapitoly 2

KAPITOLA 1: ÚVOD DO LOGIKY

Dvě definice:

1. soubor principů, které jsou základem uspořádání prvků nějakého systému (např. programu, zařízení, protokolu)
2. věda o uvažování prováděném podle striktních pravidel zachovávajících platnost

Dvě definice:

1. soubor principů, které jsou základem uspořádání prvků nějakého systému (např. programu, zařízení, protokolu)
2. věda o uvažování prováděném podle striktních pravidel zachovávajících platnost

V informatice obojí: daný systém nejprve *formálně popíšeme*, a poté o něm *formálně uvažujeme* (automaticky!), tj. odvozujeme *platné inference* za použití nějakého *dokazovacího systému*

Filozofie → Matematika → Teoretická informatika →

Aplikovaná informatika

Filozofie → Matematika → Teoretická informatika →

Aplikovaná informatika

- logic programming
- discrete optimization (SAT solving, scheduling, planning)
- database theory
- verification (software, hardware, protocol)
- automated reasoning and proving
- knowledge-based representation
- artificial intelligence

1.1 Výroková logika

Příklad ze života: Hledání pokladu

Při hledání pokladu jsme narazili na rozcestí dvou chodeb. Víme, že na konci každé chodby je buď poklad, nebo drak, ale ne obojí.

Příklad ze života: Hledání pokladu

Při hledání pokladu jsme narazili na rozcestí dvou chodeb. Víme, že na konci každé chodby je buď poklad, nebo drak, ale ne obojí.

Trpaslík nám řekl, že:

- *“Alespoň jedna z těchto dvou chodeb vede k pokladu”,* a že
- *“První chodba vede k drakovi.”*

Příklad ze života: Hledání pokladu

Při hledání pokladu jsme narazili na rozcestí dvou chodeb. Víme, že na konci každé chodby je buď poklad, nebo drak, ale ne obojí.

Trpaslík nám řekl, že:

- *“Alespoň jedna z těchto dvou chodeb vede k pokladu”,* a že
- *“První chodba vede k drakovi.”*

Je známo, že trpaslíci buď vždy mluví pravdu, nebo vždy lžou. Kterou cestou se máme vydat?

Výroky neformálně

Výrok je tvrzení, kterému lze přiřadit pravdivostní hodnotu:

pravdivý (*True*, 1), nebo lživý (*False*, 0)

Výroky neformálně

Výrok je tvrzení, kterému lze přiřadit pravdivostní hodnotu:

pravdivý (*True*, 1), nebo lživý (*False*, 0)

Prvovýroky (atomické výroky, výrokové proměnné) zkombinované pomocí logických spojek a závorek do složených výroků:

Výroky neformálně

Výrok je tvrzení, kterému lze přiřadit pravdivostní hodnotu:

pravdivý (*True*, 1), nebo **lživý** (*False*, 0)

Prvovýroky (**atomické výroky**, **výrokové proměnné**) zkombinované pomocí logických spojek a závorek do **složených výroků**:

“(Trpaslík lže,) *právě když* (druhá chodba vede k drakovi.)”

Výroky neformálně

Výrok je tvrzení, kterému lze přiřadit pravdivostní hodnotu:

pravdivý (*True*, 1), nebo lživý (*False*, 0)

Prvovýroky (atomické výroky, výrokové proměnné) zkombinované pomocí logických spojek a závorek do složených výroků:

“(Trpaslík lže,) právě když (druhá chodba vede k drakovi.)”

- \neg “neplatí X”, *negace*
- \wedge “X a Y”, *konjunkce*
- \vee “X nebo Y”, *disjunkce* (není exkluzivní)
- \rightarrow “pokud X, potom Y”, *implikace* (čistě logická)
- \leftrightarrow “X, právě když Y”, *ekvivalence*

Volba množiny prvovýroků: **bity informace popisující daný systém**

Formalizace ve výrokové logice

Volba množiny prvovýroků: *bity informace popisující daný systém*

$p_1 = \text{"Poklad je v první chodbě."}$

$p_2 = \text{"Poklad je ve druhé chodbě."}$

Formalizace ve výrokové logice

Volba množiny prvovýroků: *bity informace popisující daný systém*

$p_1 = \text{"Poklad je v první chodbě."}$

$p_2 = \text{"Poklad je ve druhé chodbě."}$

(Co nejmenší, např. hodnota $t = \text{"Trpaslík mluví pravdu."}$ je jednoznačně určena hodnotami $\mathbb{P} = \{p_1, p_2\}$.)

Formalizace ve výrokové logice

Volba množiny prvovýroků: *bity informace popisující daný systém*

$p_1 = \text{"Poklad je v první chodbě."}$

$p_2 = \text{"Poklad je ve druhé chodbě."}$

(Co nejmenší, např. hodnota $t = \text{"Trpaslík mluví pravdu."}$ je jednoznačně určená hodnotami $\mathbb{P} = \{p_1, p_2\}$.)

- *Poklad nebo drak, ale ne obojí:* zakódované do volby \mathbb{P} (přítomnost draka je absence pokladu)
- *"První chodba vede k drakovi."* $\Leftrightarrow \neg p_1$
- *"Alespoň jedna z chodeb vede k pokladu."* $\Leftrightarrow p_1 \vee p_2$
- *Trpaslík buď mluví pravdu, nebo lže:*

$$\varphi = (\neg p_1 \wedge (p_1 \vee p_2)) \vee (\neg(\neg p_1) \wedge \neg(p_1 \vee p_2))$$

Formalizace ve výrokové logice

Volba množiny prvovýroků: *bity informace popisující daný systém*

$p_1 = \text{"Poklad je v první chodbě."}$

$p_2 = \text{"Poklad je ve druhé chodbě."}$

(Co nejmenší, např. hodnota $t = \text{"Trpaslík mluví pravdu."}$ je jednoznačně určená hodnotami $\mathbb{P} = \{p_1, p_2\}$.)

- *Poklad nebo drak, ale ne obojí:* zakódované do volby \mathbb{P} (přítomnost draka je absence pokladu)
- *"První chodba vede k drakovi."* $\Leftrightarrow \neg p_1$
- *"Alespoň jedna z chodeb vede k pokladu."* $\Leftrightarrow p_1 \vee p_2$
- *Trpaslík buď mluví pravdu, nebo lže:*

$$\varphi = (\neg p_1 \wedge (p_1 \vee p_2)) \vee (\neg(\neg p_1) \wedge \neg(p_1 \vee p_2))$$

Teorie $T = \{\varphi\}$ v **jazyce** $\mathbb{P} = \{p_1, p_2\}$, φ je **axiom** T .

Modely a důsledky

Lze určit, kde je poklad? Je p_1 nebo p_2 **důsledkem** φ resp. T ?

“**Svět**”, ve kterém je např. v první chodbě poklad a ve druhé drak, popíšeme pomocí **pravdivostního ohodnocení** $p_1 = 1, p_2 = 0$, neboli **modelu** $v = (1, 0)$ jazyka \mathbb{P} . Celkem máme 4 “světy” a modely:

$$M_{\mathbb{P}} = \{(0, 0), (0, 1), (1, 0), (1, 1)\}.$$

Modely a důsledky

Lze určit, kde je poklad? Je p_1 nebo p_2 **důsledkem** φ resp. T ?

“**Svět**”, ve kterém je např. v první chodbě poklad a ve druhé drak, popíšeme pomocí **pravdivostního ohodnocení** $p_1 = 1, p_2 = 0$, neboli **modelu** $v = (1, 0)$ jazyka \mathbb{P} . Celkem máme 4 “světy” a modely:

$$M_{\mathbb{P}} = \{(0, 0), (0, 1), (1, 0), (1, 1)\}.$$

Je “svět” popsáný modelem $v = (1, 0)$ *konzistentní* s tím, co víme, tj. **platí** v modelu v výrok φ resp. teorie T ? Vyhodnotíme podle stromové struktury φ :

$$v(p_1) = 1, v(p_2) = 0, v(\neg p_1) = 0, v(p_1 \vee p_2) = 1, \dots, \mathbf{v(\varphi) = 0}$$

Modely a důsledky

Lze určit, kde je poklad? Je p_1 nebo p_2 **důsledkem** φ resp. T ?

“Svět”, ve kterém je např. v první chodbě poklad a ve druhé drak, popíšeme pomocí **pravdivostního ohodnocení** $p_1 = 1, p_2 = 0$, neboli **modelu** $v = (1, 0)$ jazyka \mathbb{P} . Celkem máme 4 “světy” a modely:

$$M_{\mathbb{P}} = \{(0, 0), (0, 1), (1, 0), (1, 1)\}.$$

Je “svět” popsán modelem $v = (1, 0)$ *konzistentní* s tím, co víme, tj. **platí** v modelu v výrok φ resp. teorie T ? Vyhodnotíme podle stromové struktury φ :

$$v(p_1) = 1, v(p_2) = 0, v(\neg p_1) = 0, v(p_1 \vee p_2) = 1, \dots, v(\varphi) = 0$$

Množina **modelů výroku** φ (resp. *modelů teorie* T):

$$M_{\mathbb{P}}(\varphi) = M_{\mathbb{P}}(T) = \{(0, 1)\}.$$

V **každém modelu** teorie T platí výrok p_2 , neboli p_2 je **důsledek** T .

Ověřovat všechny modely je nepraktické, pro $|\mathbb{P}| = n$ máme 2^n modelů, a \mathbb{P} může být i nekonečná.

Ověřovat všechny modely je nepraktické, pro $|\mathbb{P}| = n$ máme 2^n modelů, a \mathbb{P} může být i nekonečná.

Dokazovací systém

- **důkaz** výroku ψ z teorie T je formálně definovaný syntaktický objekt, snadno (mechanicky) ověřitelný
- lze hledat algoritmicky čistě na základě struktury ψ a axiomů T (“*syntaxe*”), nemusíme se zabývat modely (“*sémantikou*”).

Ověřovat všechny modely je nepraktické, pro $|\mathbb{P}| = n$ máme 2^n modelů, a \mathbb{P} může být i nekonečná.

Dokazovací systém

- **důkaz** výroku ψ z teorie T je formálně definovaný syntaktický objekt, snadno (mechanicky) ověřitelný
- lze hledat algoritmicky čistě na základě struktury ψ a axiomů T (“*syntaxe*”), nemusíme se zabývat modely (“*sémantikou*”).

Klíčové vlastnosti:

Ověřovat všechny modely je nepraktické, pro $|\mathbb{P}| = n$ máme 2^n modelů, a \mathbb{P} může být i nekonečná.

Dokazovací systém

- **důkaz** výroku ψ z teorie T je formálně definovaný syntaktický objekt, snadno (mechanicky) ověřitelný
- lze hledat algoritmicky čistě na základě struktury ψ a axiomů T (“syntaxe”), nemusíme se zabývat modely (“sémantikou”).

Klíčové vlastnosti:

- **korektnost**: pokud existuje důkaz ψ z T , potom ψ platí v T
- **úplnost**, pokud ψ platí v T , potom existuje důkaz ψ z T

Ověřovat všechny modely je nepraktické, pro $|\mathbb{P}| = n$ máme 2^n modelů, a \mathbb{P} může být i nekonečná.

Dokazovací systém

- **důkaz** výroku ψ z teorie T je formálně definovaný syntaktický objekt, snadno (mechanicky) ověřitelný
- lze hledat algoritmicky čistě na základě struktury ψ a axiomů T (“syntaxe”), nemusíme se zabývat modely (“sémantikou”).

Klíčové vlastnosti:

- **korektnost**: pokud existuje důkaz ψ z T , potom ψ platí v T
- **úplnost**, pokud ψ platí v T , potom existuje důkaz ψ z T

Ukážeme si **metodu analytického tabla** a **rezoluční metodu**. Obě dokazují *sporem*: předpokládají platnost T a $\neg\psi$, hledají spor.

Metoda analytického tabla

- důkaz je strom olabelovaný předpoklady o platnosti výroků
- v kořeni: **neplatí** dokazovaný výrok ψ (důkaz sporem)
- připojíme platnost axiomů z T

Metoda analytického tabla

- důkaz je strom olabelovaný předpoklady o platnosti výroků
- v kořeni: **neplatí** dokazovaný výrok ψ (důkaz sporem)
- připojíme platnost axiomů z T
- při konstrukci zjednodušujeme výroky ve vrcholech, **invariant**:

Metoda analytického tabla

- důkaz je strom olabelovaný předpoklady o platnosti výroků
- v kořeni: **neplatí** dokazovaný výrok ψ (důkaz sporem)
- připojíme platnost axiomů z T
- při konstrukci zjednodušujeme výroky ve vrcholech, **invariant**:

Každý model teorie T , ve kterém neplatí ψ , se musí shodovat s některou z větví tabla.

Metoda analytického tabla

- důkaz je strom olabelovaný předpoklady o platnosti výroků
- v kořeni: **neplatí** dokazovaný výrok ψ (důkaz sporem)
- připojíme platnost axiomů z T
- při konstrukci zjednodušujeme výroky ve vrcholech, **invariant**:

Každý model teorie T , ve kterém neplatí ψ , se musí shodovat s některou z větví tabla.

např.:

True $(\varphi_1 \rightarrow \varphi_2)$ zredukujeme rozvětvením na **False** φ_1 a **True** φ_2 ,
False $(\varphi_1 \rightarrow \varphi_2)$ zredukujeme připojením **True** φ_1 a **False** φ_2 .

Metoda analytického tabla

- důkaz je strom olabelovaný předpoklady o platnosti výroků
- v kořeni: **neplatí** dokazovaný výrok ψ (důkaz sporem)
- připojíme platnost axiomů z T
- při konstrukci zjednodušujeme výroky ve vrcholech, **invariant**:

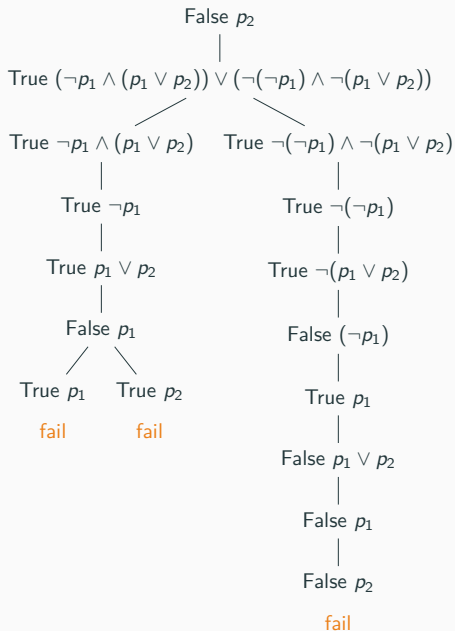
Každý model teorie T , ve kterém neplatí ψ , se musí shodovat s některou z větví tabla.

např.:

True ($\varphi_1 \rightarrow \varphi_2$) zredukujeme rozvětvením na **False** φ_1 a **True** φ_2 ,
False ($\varphi_1 \rightarrow \varphi_2$) zredukujeme připojením **True** φ_1 a **False** φ_2 .

- **sporná** větev = předpokládá True i False stejného výroku
- **důkaz** = všechny větve sporné (tj. nemůže existovat model T , ve kterém neplatí ψ)

Příklad tablo důkazu



Konjunktivní normální forma (CNF)

literál $p, \neg p$ klauzule disjunkce literálů CNF konjunkce klauzulí

každý výrok má ekvivalentní CNF ($\psi \sim \psi'$, stejné modely)

Konjunktivní normální forma (CNF)

literál $p, \neg p$ klauzule disjunkce literálů CNF konjunkce klauzulí

každý výrok má ekvivalentní CNF ($\psi \sim \psi'$, stejné modely)

např. pro výrok $(\neg p_1 \wedge (p_1 \vee p_2)) \vee (\neg(\neg p_1) \wedge \neg(p_1 \vee p_2))$

Konjunktivní normální forma (CNF)

literál $p, \neg p$ klauzule disjunkce literálů CNF konjunkce klauzulí

každý výrok má ekvivalentní CNF ($\psi \sim \psi'$, stejné modely)

např. pro výrok $(\neg p_1 \wedge (p_1 \vee p_2)) \vee (\neg(\neg p_1) \wedge \neg(p_1 \vee p_2))$

nahradíme $\neg(\neg p_1) \sim p_1$ a $\neg(p_1 \vee p_2) \sim (\neg p_1 \wedge \neg p_2)$ (*De Morgan*)

Konjunktivní normální forma (CNF)

literál $p, \neg p$ klauzule disjunkce literálů CNF konjunkce klauzulí

každý výrok má ekvivalentní CNF ($\psi \sim \psi'$, stejné modely)

např. pro výrok $(\neg p_1 \wedge (p_1 \vee p_2)) \vee (\neg(\neg p_1) \wedge \neg(p_1 \vee p_2))$

nahradíme $\neg(\neg p_1) \sim p_1$ a $\neg(p_1 \vee p_2) \sim (\neg p_1 \wedge \neg p_2)$ (*De Morgan*)

$$(\neg p_1 \wedge (p_1 \vee p_2)) \vee (p_1 \wedge \neg p_1 \wedge \neg p_2)$$

Konjunktivní normální forma (CNF)

literál $p, \neg p$ **klauzule** disjunkce literálů **CNF** konjunkce klauzulí

každý výrok má **ekvivalentní** CNF ($\psi \sim \psi'$, stejné modely)

např. pro výrok $(\neg p_1 \wedge (p_1 \vee p_2)) \vee (\neg(\neg p_1) \wedge \neg(p_1 \vee p_2))$

nahradíme $\neg(\neg p_1) \sim p_1$ a $\neg(p_1 \vee p_2) \sim (\neg p_1 \wedge \neg p_2)$ (*De Morgan*)

$$(\neg p_1 \wedge (p_1 \vee p_2)) \vee (p_1 \wedge \neg p_1 \wedge \neg p_2)$$

a dále opakovaně použijeme **distributivitu** \vee vůči \wedge :

Konjunktivní normální forma (CNF)

literál $p, \neg p$ **klauzule** disjunkce literálů **CNF** konjunkce klauzulí

každý výrok má **ekvivalentní** CNF ($\psi \sim \psi'$, stejné modely)

např. pro výrok $(\neg p_1 \wedge (p_1 \vee p_2)) \vee (\neg(\neg p_1) \wedge \neg(p_1 \vee p_2))$

nahradíme $\neg(\neg p_1) \sim p_1$ a $\neg(p_1 \vee p_2) \sim (\neg p_1 \wedge \neg p_2)$ (*De Morgan*)

$$(\neg p_1 \wedge (p_1 \vee p_2)) \vee (p_1 \wedge \neg p_1 \wedge \neg p_2)$$

a dále opakovaně použijeme **distributivitu** \vee vůči \wedge :

$$\begin{aligned} &(\neg p_1 \vee p_1) \wedge (\neg p_1 \vee \neg p_1) \wedge (\neg p_1 \vee \neg p_2) \wedge (p_1 \vee p_2 \vee p_1) \wedge \\ &\quad (p_1 \vee p_2 \vee \neg p_1) \wedge (p_1 \vee p_2 \vee \neg p_2) \end{aligned}$$

Konjunktivní normální forma (CNF)

literál $p, \neg p$ **klauzule** disjunkce literálů **CNF** konjunkce klauzulí

každý výrok má **ekvivalentní** CNF ($\psi \sim \psi'$, stejné modely)

např. pro výrok $(\neg p_1 \wedge (p_1 \vee p_2)) \vee (\neg(\neg p_1) \wedge \neg(p_1 \vee p_2))$

nahradíme $\neg(\neg p_1) \sim p_1$ a $\neg(p_1 \vee p_2) \sim (\neg p_1 \wedge \neg p_2)$ (*De Morgan*)

$$(\neg p_1 \wedge (p_1 \vee p_2)) \vee (p_1 \wedge \neg p_1 \wedge \neg p_2)$$

a dále opakovaně použijeme **distributivitu** \vee vůči \wedge :

$$\begin{aligned} &(\neg p_1 \vee p_1) \wedge (\neg p_1 \vee \neg p_1) \wedge (\neg p_1 \vee \neg p_2) \wedge (p_1 \vee p_2 \vee p_1) \wedge \\ &\quad (p_1 \vee p_2 \vee \neg p_1) \wedge (p_1 \vee p_2 \vee \neg p_2) \end{aligned}$$

už je CNF, ještě zjednodušíme: odstraníme duplicitní literály, a klauzule obsahující p_i a zároveň $\neg p_i$ (to jsou **tautologie**)

Konjunktivní normální forma (CNF)

literál $p, \neg p$ **klauzule** disjunkce literálů **CNF** konjunkce klauzulí

každý výrok má **ekvivalentní** CNF ($\psi \sim \psi'$, stejné modely)

např. pro výrok $(\neg p_1 \wedge (p_1 \vee p_2)) \vee (\neg(\neg p_1) \wedge \neg(p_1 \vee p_2))$

nahradíme $\neg(\neg p_1) \sim p_1$ a $\neg(p_1 \vee p_2) \sim (\neg p_1 \wedge \neg p_2)$ (*De Morgan*)

$$(\neg p_1 \wedge (p_1 \vee p_2)) \vee (p_1 \wedge \neg p_1 \wedge \neg p_2)$$

a dále opakovaně použijeme **distributivitu** \vee vůči \wedge :

$$\begin{aligned} &(\neg p_1 \vee p_1) \wedge (\neg p_1 \vee \neg p_1) \wedge (\neg p_1 \vee \neg p_2) \wedge (p_1 \vee p_2 \vee p_1) \wedge \\ &\quad (p_1 \vee p_2 \vee \neg p_1) \wedge (p_1 \vee p_2 \vee \neg p_2) \end{aligned}$$

už je CNF, ještě zjednodušíme: odstraníme duplicitní literály, a klauzule obsahující p_i a zároveň $\neg p_i$ (to jsou **tautologie**)

$$\neg p_1 \wedge (\neg p_1 \vee \neg p_2) \wedge (p_1 \vee p_2)$$

Dk sporem, převed' **negaci** dokazovaného do CNF a přidej k T

Dk sporem, převed' **negaci** dokazovaného do CNF a přidej k T

p_2 platí v T , právě když je následující CNF výrok **nesplnitelný**:

$$\neg p_1 \wedge (\neg p_1 \vee \neg p_2) \wedge (p_1 \vee p_2) \wedge \neg p_2$$

Rezoluční důkaz

Dk sporem, převed' **negaci** dokazovaného do CNF a přidej k T

p_2 platí v T , právě když je následující CNF výrok **nesplnitelný**:

$$\neg p_1 \wedge (\neg p_1 \vee \neg p_2) \wedge (p_1 \vee p_2) \wedge \neg p_2$$

množinový zápis: $S = \{\{\neg p_1\}, \{\neg p_1, \neg p_2\}, \{p_1, p_2\}, \{\neg p_2\}\}$

Rezoluční důkaz

Dk sporem, převed' **negaci** dokazovaného do CNF a přidej k T
 p_2 platí v T , právě když je následující CNF výrok **nesplnitelný**:

$$\neg p_1 \wedge (\neg p_1 \vee \neg p_2) \wedge (p_1 \vee p_2) \wedge \neg p_2$$

množinový zápis: $S = \{\{\neg p_1\}, \{\neg p_1, \neg p_2\}, \{p_1, p_2\}, \{\neg p_2\}\}$

rezoluční pravidlo: je-li $p \in C_1$ a $\neg p \in C_2$, potom *rezolventa*

$$C = (C_1 \setminus \{p\}) \cup (C_2 \setminus \{\neg p\})$$

platí v každém modelu, ve kterém platí C_1 i C_2

Rezoluční důkaz

Dk sporem, převed' **negaci** dokazovaného do CNF a přidej k T

p_2 platí v T , právě když je následující CNF výrok **nesplnitelný**:

$$\neg p_1 \wedge (\neg p_1 \vee \neg p_2) \wedge (p_1 \vee p_2) \wedge \neg p_2$$

množinový zápis: $S = \{\{\neg p_1\}, \{\neg p_1, \neg p_2\}, \{p_1, p_2\}, \{\neg p_2\}\}$

rezoluční pravidlo: je-li $p \in C_1$ a $\neg p \in C_2$, potom *rezolventa*

$$C = (C_1 \setminus \{p\}) \cup (C_2 \setminus \{\neg p\})$$

platí v každém modelu, ve kterém platí C_1 i C_2

rezoluční zamítnutí S : posloupnost klauzulí, kde každá je buď z S nebo rezolventa předchozích, poslední je prázdná klauzule \square

Rezoluční důkaz

Dk sporem, převed' **negaci** dokazovaného do CNF a přidej k T p_2 platí v T , právě když je následující CNF výrok **nesplnitelný**:

$$\neg p_1 \wedge (\neg p_1 \vee \neg p_2) \wedge (p_1 \vee p_2) \wedge \neg p_2$$

množinový zápis: $S = \{\{\neg p_1\}, \{\neg p_1, \neg p_2\}, \{p_1, p_2\}, \{\neg p_2\}\}$

rezoluční pravidlo: je-li $p \in C_1$ a $\neg p \in C_2$, potom *rezolventa*

$$C = (C_1 \setminus \{p\}) \cup (C_2 \setminus \{\neg p\})$$

platí v každém modelu, ve kterém platí C_1 i C_2

rezoluční zamítnutí S : posloupnost klauzulí, kde každá je buď z S nebo rezolventa předchozích, poslední je prázdná klauzule \square

myšlenka: protože \square nemá žádný model, je i S nesplnitelná

Příklad rezolučního důkazu

rezoluční zamítnutí (3. klauzule je rezolventou 1.&2., 5. je z 3.&4.)

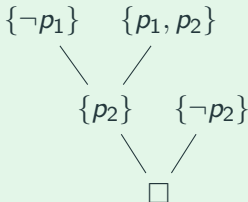
$\{\neg p_1\}, \{p_1, p_2\}, \{p_2\}, \{\neg p_2\}, \square$

Příklad rezolučního důkazu

rezoluční zamítnutí (3. klauzule je rezolventou 1.&2., 5. je z 3.&4.)

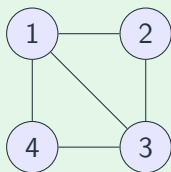
$$\{\neg p_1\}, \{p_1, p_2\}, \{p_2\}, \{\neg p_2\}, \square$$

rezoluční strom (listy klauzule z S , vnitřní vrcholy rezolventy synů)



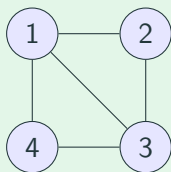
Příklad: Barvení grafů

Najděte vrcholové obarvení následujícího grafu třemi barvami.



Příklad: Barvení grafů

Najděte vrcholové obarvení následujícího grafu třemi barvami.

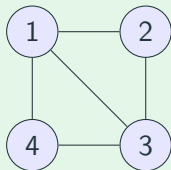


graf: množina vrcholů a množina (libovolně) **orientovaných** hran

$$\mathcal{G} = \langle V; E \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

Příklad: Barvení grafů

Najděte vrcholové obarvení následujícího grafu třemi barvami.



graf: množina vrcholů a množina (libovolně) **orientovaných** hran

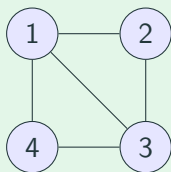
$$\mathcal{G} = \langle V; E \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

jak formalizovat? pro $v \in V$ a $c \in C = \{R, G, B\}$:

$$p_v^c = \text{"vrchol } v \text{ má barvu } c"$$

Příklad: Barvení grafů

Najděte vrcholové obarvení následujícího grafu třemi barvami.



graf: množina vrcholů a množina (libovolně) **orientovaných** hran

$$\mathcal{G} = \langle V; E \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

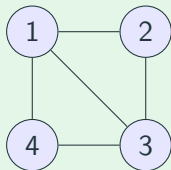
jak formalizovat? pro $v \in V$ a $c \in C = \{R, G, B\}$:

$$p_v^c = \text{"vrchol } v \text{ má barvu } c"$$

$$\mathbb{P} = \{p_v^c \mid c \in C, v \in V\} = \{p_1^R, p_1^G, p_1^B, p_2^R, p_2^G, p_2^B, p_3^R, p_3^G, p_3^B, p_4^R, p_4^G, p_4^B\}$$

Příklad: Barvení grafů

Najděte vrcholové obarvení následujícího grafu třemi barvami.



graf: množina vrcholů a množina (libovolně) **orientovaných** hran

$$\mathcal{G} = \langle V; E \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

jak formalizovat? pro $v \in V$ a $c \in C = \{R, G, B\}$:

$$p_v^c = \text{"vrchol } v \text{ má barvu } c"$$

$$\mathbb{P} = \{p_v^c \mid c \in C, v \in V\} = \{p_1^R, p_1^G, p_1^B, p_2^R, p_2^G, p_2^B, p_3^R, p_3^G, p_3^B, p_4^R, p_4^G, p_4^B\}$$

máme celkem $|\mathbb{M}_{\mathbb{P}}| = 2^{12} = 4096$ **modelů jazyka** (12-dim. vektorů)

Formalizace hranového obarvení

- každý vrchol má nejvýše jednu barvu: $4^4 = 2^8 = 256$ modelů

$$T_1 = \{(\neg p_v^R \vee \neg p_v^G) \wedge (\neg p_v^R \vee \neg p_v^B) \wedge (\neg p_v^G \vee \neg p_v^B) \mid v \in V\}$$

Formalizace hranového obarvení

- každý vrchol má nejvýše jednu barvu: $4^4 = 2^8 = 256$ modelů

$$T_1 = \{(\neg p_v^R \vee \neg p_v^G) \wedge (\neg p_v^R \vee \neg p_v^B) \wedge (\neg p_v^G \vee \neg p_v^B) \mid v \in V\}$$

- a každý vrchol má alespoň jednu barvu: $3^4 = 81$ modelů

$$T_2 = T_1 \cup \{p_v^R \vee p_v^G \vee p_v^B \mid v \in V\} = T_1 \cup \left\{ \bigvee_{c \in C} p_v^c \mid v \in V \right\}$$

Formalizace hranového obarvení

- každý vrchol má nejvýše jednu barvu: $4^4 = 2^8 = 256$ modelů

$$T_1 = \{(\neg p_v^R \vee \neg p_v^G) \wedge (\neg p_v^R \vee \neg p_v^B) \wedge (\neg p_v^G \vee \neg p_v^B) \mid v \in V\}$$

- a každý vrchol má alespoň jednu barvu: $3^4 = 81$ modelů

$$T_2 = T_1 \cup \{p_v^R \vee p_v^G \vee p_v^B \mid v \in V\} = T_1 \cup \left\{ \bigvee_{c \in C} p_v^c \mid v \in V \right\}$$

T_2 je **extenze** teorie T_1 neboť **každý důsledek** T_1 **platí i v** T_2 ,
zde dokonce $M_{\mathbb{P}}(T_2) \subseteq M_{\mathbb{P}}(T_1)$

Formalizace hranového obarvení

- každý vrchol má nejvýše jednu barvu: $4^4 = 2^8 = 256$ modelů

$$T_1 = \{(\neg p_v^R \vee \neg p_v^G) \wedge (\neg p_v^R \vee \neg p_v^B) \wedge (\neg p_v^G \vee \neg p_v^B) \mid v \in V\}$$

- a každý vrchol má alespoň jednu barvu: $3^4 = 81$ modelů

$$T_2 = T_1 \cup \{p_v^R \vee p_v^G \vee p_v^B \mid v \in V\} = T_1 \cup \left\{ \bigvee_{c \in C} p_v^c \mid v \in V \right\}$$

T_2 je **extenze** teorie T_1 neboť **každý důsledek** T_1 **platí i v** T_2 ,
zde dokonce $M_{\mathbb{P}}(T_2) \subseteq M_{\mathbb{P}}(T_1)$

- nakonec přidáme **hranovou podmínku**:

$$T_3 = T_2 \cup \left\{ \bigwedge_{c \in C} (\neg p_u^c \vee \neg p_v^c) \mid (u, v) \in E \right\}$$

Formalizace hranového obarvení

- každý vrchol má nejvýše jednu barvu: $4^4 = 2^8 = 256$ modelů

$$T_1 = \{(\neg p_v^R \vee \neg p_v^G) \wedge (\neg p_v^R \vee \neg p_v^B) \wedge (\neg p_v^G \vee \neg p_v^B) \mid v \in V\}$$

- a každý vrchol má alespoň jednu barvu: $3^4 = 81$ modelů

$$T_2 = T_1 \cup \{p_v^R \vee p_v^G \vee p_v^B \mid v \in V\} = T_1 \cup \left\{ \bigvee_{c \in C} p_v^c \mid v \in V \right\}$$

T_2 je **extenze** teorie T_1 neboť **každý důsledek** T_1 **platí i v** T_2 ,
zde dokonce $M_{\mathbb{P}}(T_2) \subseteq M_{\mathbb{P}}(T_1)$

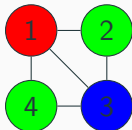
- nakonec přidáme **hranovou podmínku**:

$$T_3 = T_2 \cup \left\{ \bigwedge_{c \in C} (\neg p_u^c \vee \neg p_v^c) \mid (u, v) \in E \right\}$$

Výsledná teorie T_3 je **splnitelná** (má model), právě když je
graf \mathcal{G} 3-obarvitelný.

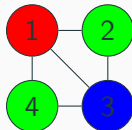
Všetchna obarvení?

T_3 má 6 modelů: $v = (1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0)$ a další získané permutací barev



Všetchna obarvení?

T_3 má 6 modelů: $v = (1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0)$ a další získané permutací barev

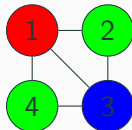


Obarvení, ve kterých je vrchol 1 modrý a vrchol 2 zelený?

Odpovídají modelům teorie $T_3 \cup \{p_1^B, p_2^G\}$

Všetchna obarvení?

T_3 má 6 modelů: $v = (1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0)$ a další získané permutací barev



Obarvení, ve kterých je vrchol 1 modrý a vrchol 2 zelený?

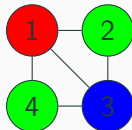
Odpovídají modelům teorie $T_3 \cup \{p_1^B, p_2^G\}$

Důkaz, že vrcholy 2 a 4 musí mít stejnou barvu?

Tablo s kořenem False $(p_2^R \wedge p_4^R) \vee (p_2^G \wedge p_4^G) \vee (p_2^B \wedge p_4^B)$

Všechna obarvení?

T_3 má 6 modelů: $v = (1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0)$ a další získané permutací barev



Obarvení, ve kterých je vrchol 1 modrý a vrchol 2 zelený?

Odpovídají modelům teorie $T_3 \cup \{p_1^B, p_2^G\}$

Důkaz, že vrcholy 2 a 4 musí mít stejnou barvu?

Tablo s kořenem False $(p_2^R \wedge p_4^R) \vee (p_2^G \wedge p_4^G) \vee (p_2^B \wedge p_4^B)$

Nebo **rezolucí**: přidáme **negaci** $(p_2^R \wedge p_4^R) \vee (p_2^G \wedge p_4^G) \vee (p_2^B \wedge p_4^B)$,
vše převedeme do CNF a zamítneme

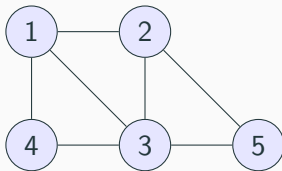
1.2 Predikátová logika

Nevýhody formalizace ve výrokové logice

Teorie T_3 je poměrně velká, a 'natvrdo' kóduje graf \mathcal{G} .

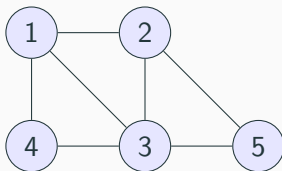
Nevýhody formalizace ve výrokové logice

Teorie T_3 je poměrně velká, a 'natvrdo' kóduje graf \mathcal{G} .



Nevýhody formalizace ve výrokové logice

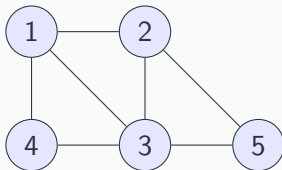
Teorie T_3 je poměrně velká, a 'natvrdo' kóduje graf \mathcal{G} .



Obohatit jazyk $\mathbb{P}' = \mathbb{P} \cup \{p_5^R, p_5^G, p_5^B\}$ a vytvořit ještě větší teorii T'_3 přidáním axiomů o vrcholu 5 a hranách $(2, 5), (3, 5)$?

Nevýhody formalizace ve výrokové logice

Teorie T_3 je poměrně velká, a 'natvrdo' kóduje graf \mathcal{G} .

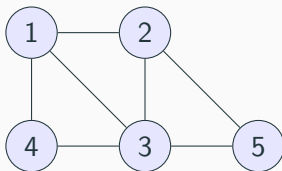


Obohatit jazyk $\mathbb{P}' = \mathbb{P} \cup \{p_5^R, p_5^G, p_5^B\}$ a vytvořit ještě větší teorii T'_3 přidáním axiomů o vrcholu 5 a hranách $(2, 5), (3, 5)$?

A co vlastnosti obecně platné o všech nebo mnoha grafech?

Nevýhody formalizace ve výrokové logice

Teorie T_3 je poměrně velká, a 'natvrdo' kóduje graf \mathcal{G} .



Obohatit jazyk $\mathbb{P}' = \mathbb{P} \cup \{p_5^R, p_5^G, p_5^B\}$ a vytvořit ještě větší teorii T'_3 přidáním axiomů o vrcholu 5 a hranách $(2, 5), (3, 5)$?

A co vlastnosti obecně platné o všech nebo mnoha grafech?

V **predikátové logice** můžeme mluvit o **vrcholech** grafu pomocí **proměnných** a přirozeně vyjádřit vlastnosti jako:

- “z vrcholu u vede hrana do vrcholu v ”
- “vrchol u je zelený”

Modely už nejsou 0–1 vektory, ale **struktury**, např. naše (orientované) grafy:

$$\mathcal{G} = \langle V^{\mathcal{G}}; E^{\mathcal{G}} \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

$$\mathcal{G}' = \langle V^{\mathcal{G}'}; E^{\mathcal{G}'} \rangle = \langle \{1, 2, 3, 4, 5\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4), (2, 5), (3, 5)\} \rangle$$

Modely už nejsou 0–1 vektory, ale **struktury**, např. naše (orientované) grafy:

$$\mathcal{G} = \langle V^{\mathcal{G}}; E^{\mathcal{G}} \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

$$\mathcal{G}' = \langle V^{\mathcal{G}'}; E^{\mathcal{G}'} \rangle = \langle \{1, 2, 3, 4, 5\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4), (2, 5), (3, 5)\} \rangle$$

- množina vrcholů, a binární relace na této množině

Modely už nejsou 0–1 vektory, ale **struktury**, např. naše (orientované) grafy:

$$\mathcal{G} = \langle V^{\mathcal{G}}; E^{\mathcal{G}} \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

$$\mathcal{G}' = \langle V^{\mathcal{G}'}; E^{\mathcal{G}'} \rangle = \langle \{1, 2, 3, 4, 5\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4), (2, 5), (3, 5)\} \rangle$$

- množina vrcholů, a binární relace na této množině
- **jazyk** specifikuje kolik **relací** jakých arit má struktura mít, a symboly pro ně

Modely už nejsou 0–1 vektory, ale **struktury**, např. naše (orientované) grafy:

$$\mathcal{G} = \langle V^{\mathcal{G}}; E^{\mathcal{G}} \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

$$\mathcal{G}' = \langle V^{\mathcal{G}'}; E^{\mathcal{G}'} \rangle = \langle \{1, 2, 3, 4, 5\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4), (2, 5), (3, 5)\} \rangle$$

- množina vrcholů, a binární relace na této množině
- **jazyk** specifikuje kolik **relací** jakých arit má struktura mít, a symboly pro ně
- např. **jazyk grafů** $\mathcal{L} = \langle E \rangle$ (kde E je binární relační symbol)

Modely už nejsou 0–1 vektory, ale **struktury**, např. naše (orientované) grafy:

$$\mathcal{G} = \langle V^{\mathcal{G}}; E^{\mathcal{G}} \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

$$\mathcal{G}' = \langle V^{\mathcal{G}'}; E^{\mathcal{G}'} \rangle = \langle \{1, 2, 3, 4, 5\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4), (2, 5), (3, 5)\} \rangle$$

- množina vrcholů, a binární relace na této množině
- **jazyk** specifikuje kolik **relací** jakých arit má struktura mít, a symboly pro ně
- např. **jazyk grafů** $\mathcal{L} = \langle E \rangle$ (kde E je binární relační symbol)
- \mathcal{G} a \mathcal{G}' jsou **struktury v jazyce** \mathcal{L} (**\mathcal{L} -struktury**)

Modely už nejsou 0–1 vektory, ale **struktury**, např. naše (orientované) grafy:

$$\mathcal{G} = \langle V^{\mathcal{G}}; E^{\mathcal{G}} \rangle = \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\} \rangle$$

$$\mathcal{G}' = \langle V^{\mathcal{G}'}; E^{\mathcal{G}'} \rangle = \langle \{1, 2, 3, 4, 5\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4), (2, 5), (3, 5)\} \rangle$$

- množina vrcholů, a binární relace na této množině
- **jazyk** specifikuje kolik **relací** jakých arit má struktura mít, a symboly pro ně
- např. **jazyk grafů** $\mathcal{L} = \langle E \rangle$ (kde E je binární relační symbol)
- \mathcal{G} a \mathcal{G}' jsou **struktury v jazyce** \mathcal{L} (**\mathcal{L} -struktury**)
- můžeme mít také **funkce** a **konstanty**, a symbol $=$ pro **rovnost**

Syntaxe: místo prvovýroků **atomické formule**, např. $E(x, y)$, kde x, y jsou **proměnné** reprezentující vrcholy; stejné logické spojky, ale navíc **kvantifikátory**:

Predikátová logika: syntaxe a sémantika

Syntaxe: místo prvovýroků **atomické formule**, např. $E(x, y)$, kde x, y jsou **proměnné** reprezentující vrcholy; stejné logické spojky, ale navíc **kvantifikátory**:

$(\forall x)$ “pro všechny vrcholy x ”

$(\exists y)$ “existuje vrchol y ”

Predikátová logika: syntaxe a sémantika

Syntaxe: místo prvovýroků **atomické formule**, např. $E(x, y)$, kde x, y jsou **proměnné** reprezentující vrcholy; stejné logické spojky, ale navíc **kvantifikátory**:

$(\forall x)$ “pro všechny vrcholy x ”

$(\exists y)$ “existuje vrchol y ”

(hrají roli “konjunkce” a “disjunkce” přes všechny prvky)

Syntaxe: místo prvovýroků **atomické formule**, např. $E(x, y)$, kde x, y jsou **proměnné** reprezentující vrcholy; stejné logické spojky, ale navíc **kvantifikátory**:

$(\forall x)$ “pro všechny vrcholy x ”

$(\exists y)$ “existuje vrchol y ”

(hrají roli “konjunkce” a “disjunkce” přes všechny prvky)

- “V grafu nejsou smyčky”: $(\forall x)(\neg E(x, x))$

- “Existuje vrchol výstupního stupně 1”:

$(\exists x)(\exists y)(E(x, y) \wedge (\forall z)(E(x, z) \rightarrow y = z))$

Syntaxe: místo prvovýroků **atomické formule**, např. $E(x, y)$, kde x, y jsou **proměnné** reprezentující vrcholy; stejné logické spojky, ale navíc **kvantifikátory**:

$(\forall x)$ “pro všechny vrcholy x ”

$(\exists y)$ “existuje vrchol y ”

(hrají roli “konjunkce” a “disjunkce” přes všechny prvky)

- “V grafu nejsou smyčky”: $(\forall x)(\neg E(x, x))$

- “Existuje vrchol výstupního stupně 1”:

$(\exists x)(\exists y)(E(x, y) \wedge (\forall z)(E(x, z) \rightarrow y = z))$

Sémantika: V daném grafu \mathcal{G} a při **dosazení** vrcholu u za proměnnou x a vrcholu v za proměnnou y **vyhodnotíme** $E(x, y)$ jako **True**, právě když $(u, v) \in E^{\mathcal{G}}$.

Barvení grafů v predikátové logice

Jazyk $\mathcal{L}' = \langle E, R, G, B \rangle$, kde E je binární a R, G, B jsou unární relační symboly ($R(x)$ znamená “*vrchol x je červený*”)

Barvení grafů v predikátové logice

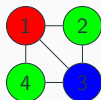
Jazyk $\mathcal{L}' = \langle E, R, G, B \rangle$, kde E je binární a R, G, B jsou unární relační symboly ($R(x)$ znamená “*vrchol x je červený*”)

\mathcal{L}' -struktura: graf s trojicí množin vrcholů

Barvení grafů v predikátové logice

Jazyk $\mathcal{L}' = \langle E, R, G, B \rangle$, kde E je binární a R, G, B jsou unární relační symboly ($R(x)$ znamená “vrchol x je červený”)

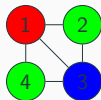
\mathcal{L}' -struktura: graf s trojicí množin vrcholů



Barvení grafů v predikátové logice

Jazyk $\mathcal{L}' = \langle E, R, G, B \rangle$, kde E je binární a R, G, B jsou unární relační symboly ($R(x)$ znamená “vrchol x je červený”)

\mathcal{L}' -struktura: graf s trojicí množin vrcholů

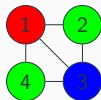


$$\begin{aligned}\mathcal{G}_C &= \langle V^{\mathcal{G}_C}; E^{\mathcal{G}_C}, R^{\mathcal{G}_C}, G^{\mathcal{G}_C}, B^{\mathcal{G}_C} \rangle \\ &= \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\}, \{1\}, \{2, 4\}, \{3\} \rangle\end{aligned}$$

Barvení grafů v predikátové logice

Jazyk $\mathcal{L}' = \langle E, R, G, B \rangle$, kde E je binární a R, G, B jsou unární relační symboly ($R(x)$ znamená “vrchol x je červený”)

\mathcal{L}' -struktura: graf s trojicí množin vrcholů



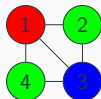
$$\begin{aligned}\mathcal{G}_C &= \langle V^{\mathcal{G}_C}; E^{\mathcal{G}_C}, R^{\mathcal{G}_C}, G^{\mathcal{G}_C}, B^{\mathcal{G}_C} \rangle \\ &= \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\}, \{1\}, \{2, 4\}, \{3\} \rangle\end{aligned}$$

\mathcal{G}_C je **expanze** \mathcal{L} -struktury \mathcal{G} do jazyka \mathcal{L}'

Barvení grafů v predikátové logice

Jazyk $\mathcal{L}' = \langle E, R, G, B \rangle$, kde E je binární a R, G, B jsou unární relační symboly ($R(x)$ znamená “vrchol x je červený”)

\mathcal{L}' -struktura: graf s trojicí množin vrcholů



$$\begin{aligned}\mathcal{G}_C &= \langle V^{\mathcal{G}_C}; E^{\mathcal{G}_C}, R^{\mathcal{G}_C}, G^{\mathcal{G}_C}, B^{\mathcal{G}_C} \rangle \\ &= \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\}, \{1\}, \{2, 4\}, \{3\} \rangle\end{aligned}$$

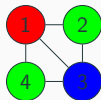
\mathcal{G}_C je **expanze** \mathcal{L} -struktury \mathcal{G} **do jazyka** \mathcal{L}'

Nejvýše jedna barva, alespoň jedna barva, hranová podmínka:

Barvení grafů v predikátové logice

Jazyk $\mathcal{L}' = \langle E, R, G, B \rangle$, kde E je binární a R, G, B jsou unární relační symboly ($R(x)$ znamená “vrchol x je červený”)

\mathcal{L}' -struktura: graf s trojicí množin vrcholů



$$\begin{aligned}\mathcal{G}_C &= \langle V^{\mathcal{G}_C}; E^{\mathcal{G}_C}, R^{\mathcal{G}_C}, G^{\mathcal{G}_C}, B^{\mathcal{G}_C} \rangle \\ &= \langle \{1, 2, 3, 4\}; \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4)\}, \{1\}, \{2, 4\}, \{3\} \rangle\end{aligned}$$

\mathcal{G}_C je **expanze** \mathcal{L} -struktury \mathcal{G} **do jazyka** \mathcal{L}'

Nejvýše jedna barva, alespoň jedna barva, hranová podmínka:

- $(\forall x)((\neg R(x) \vee \neg G(x)) \wedge (\neg R(x) \vee \neg B(x)) \wedge (\neg G(x) \vee \neg B(x)))$
- $(\forall x)(R(x) \vee G(x) \vee B(x))$
- $(\forall x)(\forall y)(E(x, y) \rightarrow ((\neg R(x) \vee \neg R(y)) \wedge (\neg G(x) \vee \neg G(y)) \wedge (\neg B(x) \vee \neg B(y))))$

1.3 Další druhy logických systémů

- Predikátová logika, kde proměnné reprezentují jednotlivé vrcholy, je logika **prvního řádu** (**first-order**, **FO**)

- Predikátová logika, kde proměnné reprezentují jednotlivé vrcholy, je logika **prvního řádu** (**first-order**, **FO**)
- Logika **druhého řádu** (**second-order**, **SO**): proměnné i pro množiny vrcholů a n -tic vrcholů (tj. relace, funkce)

- Predikátová logika, kde proměnné reprezentují jednotlivé vrcholy, je logika **prvního řádu** (**first-order**, **FO**)
- Logika **druhého řádu** (**second-order**, **SO**): proměnné i pro množiny vrcholů a n -tic vrcholů (tj. relace, funkce)

- Predikátová logika, kde proměnné reprezentují jednotlivé vrcholy, je logika **prvního řádu** (**first-order**, **FO**)
- Logika **druhého řádu** (**second-order**, **SO**): proměnné i pro množiny vrcholů a n -tic vrcholů (tj. relace, funkce)

$$(\exists S)(\forall x)(\forall y)(E(x, y) \rightarrow (S(x) \leftrightarrow \neg S(y)))$$

- Predikátová logika, kde proměnné reprezentují jednotlivé vrcholy, je logika **prvního řádu** (**first-order**, **FO**)
- Logika **druhého řádu** (**second-order**, **SO**): proměnné i pro množiny vrcholů a n -tic vrcholů (tj. relace, funkce)

$$(\exists S)(\forall x)(\forall y)(E(x, y) \rightarrow (S(x) \leftrightarrow \neg S(y)))$$

“Graf je bipartitní.”

- Predikátová logika, kde proměnné reprezentují jednotlivé vrcholy, je logika **prvního řádu** (**first-order**, **FO**)
- Logika **druhého řádu** (**second-order**, **SO**): proměnné i pro množiny vrcholů a n -tic vrcholů (tj. relace, funkce)

$$(\exists S)(\forall x)(\forall y)(E(x, y) \rightarrow (S(x) \leftrightarrow \neg S(y)))$$

“Graf je bipartitní.”

- A v logice *třetího řádu* máme i množiny množin (např. v topologii).

Kromě toho lze zobecnit pojem platnosti (pravdy):

- **temporální logiky** (platnost 'vždy', 'někdy v budoucnosti', 'dokud' apod.) – např. v paralelním programování

Kromě toho lze zobecnit pojem platnosti (pravdy):

- **temporální logiky** (platnost 'vždy', 'někdy v budoucnosti', 'dokud' apod.) – např. v paralelním programování
- **modální logiky** ('je možné', 'je nutné') – v umělé inteligenci, uvažování autonomních agentů o svém okolí

Kromě toho lze zobecnit pojem platnosti (pravdy):

- **temporální logiky** (platnost 'vždy', 'někdy v budoucnosti', 'dokud' apod.) – např. v paralelním programování
- **modální logiky** ('je možné', 'je nutné') – v umělé inteligenci, uvažování autonomních agentů o svém okolí
- **fuzzy logiky** ('je 0.35 pravdivé') – v automatických pračkách

Kromě toho lze zobecnit pojem platnosti (pravdy):

- **temporální logiky** (platnost 'vždy', 'někdy v budoucnosti', 'dokud' apod.) – např. v paralelním programování
- **modální logiky** ('je možné', 'je nutné') – v umělé inteligenci, uvažování autonomních agentů o svém okolí
- **fuzzy logiky** ('je 0.35 pravdivé') – v automatických pračkách
- **intuicionistická logika** (povoluje jen konstruktivní důkazy, nemá *zákon vyloučeného třetího*)

1.4 O přednášce

I. Výroková logika

- Syntaxe a sémantika
- Problém SAT
- Tablo metoda
- Rezoluční metoda

II. Predikátová logika

- Syntaxe a sémantika
- Tablo metoda v predikátové logice
- Rezoluční metoda v predikátové logice
- Aplikace: databáze, Prolog

III. Pokročilé partie

- Teorie modelů
- Nerozhodnutelnost a neúplnost

ČÁST I – VÝROKOVÁ LOGIKA

KAPITOLA 2: SYNTAXE A SÉMANTIKA VÝROKOVÉ LOGIKY

syntaxe dává pravidla pro tvoření korektních formálních výrazů sestávajících ze symbolů, a pro operace s nimi (*výrok, důkaz, ...*)

sémantika popisuje význam syntaktických objektů “v reálném světě” (*model, ...*)

syntaxe dává pravidla pro tvoření korektních formálních výrazů sestávajících ze symbolů, a pro operace s nimi (*výrok*, *důkaz*, ...)

sémantika popisuje význam syntaktických objektů “v reálném světě” (*model*, ...)

Klíčem k logice je **vztah mezi syntaxí a sémantikou**:

- sémantické objekty studujeme pomocí syntaxe (‘jaké výroky platí v modelu?’)
- syntaktické pomocí sémantiky, např. ekvivalence výroků:
 $\psi \sim \psi'$ právě když $M_{\mathbb{P}}(\psi) = M_{\mathbb{P}}(\psi')$

2.1 Syntaxe výrokové logiky

- určený množinou **prvovýroků** (**výrokových proměnných**, **atomických výroků**) – neprázdná, konečná nebo i *nekonečná*

$$\mathbb{P}_1 = \{p, q, r\}$$

$$\mathbb{P}_2 = \{p_0, p_1, p_2, p_3, \dots\} = \{p_i \mid i \in \mathbb{N}\}$$

(obvykle *spočetná, uspořádaná*)

- určený množinou **prvovýroků** (**výrokových proměnných**, **atomických výroků**) – neprázdná, konečná nebo i *nekonečná*

$$\mathbb{P}_1 = \{p, q, r\}$$

$$\mathbb{P}_2 = \{p_0, p_1, p_2, p_3, \dots\} = \{p_i \mid i \in \mathbb{N}\}$$

(obvykle *spočetná, uspořádaná*)

- dále do jazyka patří **logické symboly**:
 - logické spojky $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
 - závorky $(,)$

Výrok (výroková formule) v jazyce \mathbb{P} je prvek množiny $VF_{\mathbb{P}}$ definované *induktivně*: $VF_{\mathbb{P}}$ je nejmenší množina splňující

- pro každý prvovýrok $p \in \mathbb{P}$ platí $p \in VF_{\mathbb{P}}$,
- pro každý výrok $\varphi \in VF_{\mathbb{P}}$ je $(\neg\varphi)$ také prvek $VF_{\mathbb{P}}$
- pro každé $\varphi, \psi \in VF_{\mathbb{P}}$ jsou $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, a $(\varphi \leftrightarrow \psi)$ také prvky $VF_{\mathbb{P}}$.

Výrok (**výroková formule**) v jazyce \mathbb{P} je prvek množiny $VF_{\mathbb{P}}$ definované *induktivně*: $VF_{\mathbb{P}}$ je nejmenší množina splňující

- pro každý prvovýrok $p \in \mathbb{P}$ platí $p \in VF_{\mathbb{P}}$,
- pro každý výrok $\varphi \in VF_{\mathbb{P}}$ je $(\neg\varphi)$ také prvek $VF_{\mathbb{P}}$
- pro každé $\varphi, \psi \in VF_{\mathbb{P}}$ jsou $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, a $(\varphi \leftrightarrow \psi)$ také prvky $VF_{\mathbb{P}}$.

Výroky jsou nutně *konečné* řetězce!

Výrok

Výrok (**výroková formule**) v jazyce \mathbb{P} je prvek množiny $VF_{\mathbb{P}}$ definované *induktivně*: $VF_{\mathbb{P}}$ je nejmenší množina splňující

- pro každý prvovýrok $p \in \mathbb{P}$ platí $p \in VF_{\mathbb{P}}$,
- pro každý výrok $\varphi \in VF_{\mathbb{P}}$ je $(\neg\varphi)$ také prvek $VF_{\mathbb{P}}$
- pro každé $\varphi, \psi \in VF_{\mathbb{P}}$ jsou $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, a $(\varphi \leftrightarrow \psi)$ také prvky $VF_{\mathbb{P}}$.

Výroky jsou nutně *konečné* řetězce!

Var(φ): množina všech prvovýroků ve φ (vždy konečná)

Výrok

Výrok (**výroková formule**) v jazyce \mathbb{P} je prvek množiny $VF_{\mathbb{P}}$ definované *induktivně*: $VF_{\mathbb{P}}$ je nejmenší množina splňující

- pro každý prvovýrok $p \in \mathbb{P}$ platí $p \in VF_{\mathbb{P}}$,
- pro každý výrok $\varphi \in VF_{\mathbb{P}}$ je $(\neg\varphi)$ také prvek $VF_{\mathbb{P}}$
- pro každé $\varphi, \psi \in VF_{\mathbb{P}}$ jsou $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, a $(\varphi \leftrightarrow \psi)$ také prvky $VF_{\mathbb{P}}$.

Výroky jsou nutně *konečné* řetězce!

Var(φ): množina všech prvovýroků ve φ (vždy konečná)

podvýrok: podřetězec, který je sám výrok

Výrok

Výrok (**výroková formule**) v jazyce \mathbb{P} je prvek množiny $VF_{\mathbb{P}}$ definované *induktivně*: $VF_{\mathbb{P}}$ je nejmenší množina splňující

- pro každý prvovýrok $p \in \mathbb{P}$ platí $p \in VF_{\mathbb{P}}$,
- pro každý výrok $\varphi \in VF_{\mathbb{P}}$ je $(\neg\varphi)$ také prvek $VF_{\mathbb{P}}$
- pro každé $\varphi, \psi \in VF_{\mathbb{P}}$ jsou $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, a $(\varphi \leftrightarrow \psi)$ také prvky $VF_{\mathbb{P}}$.

Výroky jsou nutně *konečné* řetězce!

Var(φ): množina všech prvovýroků ve φ (vždy konečná)

podvýrok: podřetězec, který je sám výrok

$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$, $\text{Var}(\varphi) = \{p, q, r\}$

podvýroky: $p, q, (\neg q), (p \vee (\neg q)), r, (p \wedge q), (r \rightarrow (p \wedge q)), \varphi$

Výrok

Výrok (**výroková formule**) v jazyce \mathbb{P} je prvek množiny $VF_{\mathbb{P}}$ definované *induktivně*: $VF_{\mathbb{P}}$ je nejmenší množina splňující

- pro každý prvovýrok $p \in \mathbb{P}$ platí $p \in VF_{\mathbb{P}}$,
- pro každý výrok $\varphi \in VF_{\mathbb{P}}$ je $(\neg\varphi)$ také prvek $VF_{\mathbb{P}}$
- pro každé $\varphi, \psi \in VF_{\mathbb{P}}$ jsou $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, a $(\varphi \leftrightarrow \psi)$ také prvky $VF_{\mathbb{P}}$.

Výroky jsou nutně *konečné* řetězce!

Var(φ): množina všech prvovýroků ve φ (vždy konečná)

podvýrok: podřetězec, který je sám výrok

$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$, $\text{Var}(\varphi) = \{p, q, r\}$

podvýroky: $p, q, (\neg q), (p \vee (\neg q)), r, (p \wedge q), (r \rightarrow (p \wedge q)), \varphi$

pravda: $\top = (p \vee (\neg p))$, **spor**: $\perp = (p \wedge (\neg p))$ ($p \in \mathbb{P}$ je pevně daný)

při *zápisu* výroků můžeme vynechat některé závorky:

$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$ lze zapsat jako $p \vee \neg q \leftrightarrow (r \rightarrow p \wedge q)$

při *zápisu* výroků můžeme vynechat některé závorky:

$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$ lze zapsat jako $p \vee \neg q \leftrightarrow (r \rightarrow p \wedge q)$

- priorita operátorů: \neg nejvyšší, dále \wedge a \vee , nakonec \rightarrow a \leftrightarrow
- asociativita \wedge a \vee : nápis $p \wedge q \wedge r$ znamená výrok $(p \wedge (q \wedge r))$
- vnější závorky nemusíme psát

Konvence zápisu

při *zápisu* výroků můžeme vynechat některé závorky:

$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$ lze zapsat jako $p \vee \neg q \leftrightarrow (r \rightarrow p \wedge q)$

- priorita operátorů: \neg nejvyšší, dále \wedge a \vee , nakonec \rightarrow a \leftrightarrow
- asociativita \wedge a \vee : nápis $p \wedge q \wedge r$ znamená výrok $(p \wedge (q \wedge r))$
- vnější závorky nemusíme psát

Poznámka: v definici jsme mohli místo *infixového* zápisu zvolit *prefixový* (“polskou notaci”): “každý prvovýrok je výrok, jsou-li φ, ψ výroky, jsou výroky také $\neg\varphi$, $\wedge\varphi\psi$, $\vee\varphi\psi$, $\rightarrow\varphi\psi$, a $\leftrightarrow\varphi\psi$ ” nebo i *postfixový*

$\varphi = \leftrightarrow \vee p \neg q \rightarrow r \wedge p q$

$\varphi = p q \neg \vee r p q \wedge \rightarrow \leftrightarrow$

Konvence zápisu

při *zápisu* výroků můžeme vynechat některé závorky:

$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$ lze zapsat jako $p \vee \neg q \leftrightarrow (r \rightarrow p \wedge q)$

- priorita operátorů: \neg nejvyšší, dále \wedge a \vee , nakonec \rightarrow a \leftrightarrow
- asociativita \wedge a \vee : nápis $p \wedge q \wedge r$ znamená výrok $(p \wedge (q \wedge r))$
- vnější závorky nemusíme psát

Poznámka: v definici jsme mohli místo *infixového* zápisu zvolit *prefixový* (“polskou notaci”): “každý prvovýrok je výrok, jsou-li φ, ψ výroky, jsou výroky také $\neg\varphi$, $\wedge\varphi\psi$, $\vee\varphi\psi$, $\rightarrow\varphi\psi$, a $\leftrightarrow\varphi\psi$ ” nebo i *postfixový*

$\varphi = \leftrightarrow \vee p \neg q \rightarrow r \wedge p q$

$\varphi = p q \neg \vee r p q \wedge \rightarrow \leftrightarrow$

Důležitá je jen **stromová struktura** výroků!

Strom výroku

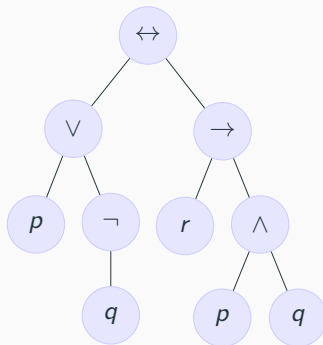
$\text{Tree}(\varphi)$ je zakořeněný uspořádaný strom, definovaný induktivně:

- $\varphi = p \in \mathbb{P}$: jediný vrchol, s labelem p
- $\varphi = (\neg\varphi')$: kořen s labelem \neg , jediný syn je kořen $\text{Tree}(\varphi')$.
- $\varphi = (\varphi' \square \varphi'')$ pro $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$: kořen s labelem \square a dvěma syny: levý syn je kořen $\text{Tree}(\varphi')$, pravý $\text{Tree}(\varphi'')$.

$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$

rekonstrukce φ průchodem stromu,
podvýroky odpovídají podstromům

$\text{Tree}(\varphi)$ je jednoznačně určený!



Teorie v jazyce \mathbb{P} je libovolná množina výroků $T \subseteq VF_{\mathbb{P}}$. Výrokům $\varphi \in T$ říkáme také **axiomy**.

$T = \emptyset$ a $T = VF_{\mathbb{P}}$ nad libovolným jazykem,

$T = \{p \wedge q, q \rightarrow (p \vee r)\}$ v jazyce $\mathbb{P} = \{p, q, r\}$

$T = \{p_0\} \cup \{p_i \rightarrow p_{i+1} \mid i \in \mathbb{N}\}$ nad *nekonečným* $\mathbb{P} = \{p_i \mid i \in \mathbb{N}\}$

Teorie v jazyce \mathbb{P} je libovolná množina výroků $T \subseteq VF_{\mathbb{P}}$. Výrokům $\varphi \in T$ říkáme také **axiomy**.

$T = \emptyset$ a $T = VF_{\mathbb{P}}$ nad libovolným jazykem,

$T = \{p \wedge q, q \rightarrow (p \vee r)\}$ v jazyce $\mathbb{P} = \{p, q, r\}$

$T = \{p_0\} \cup \{p_i \rightarrow p_{i+1} \mid i \in \mathbb{N}\}$ nad *nekonečným* $\mathbb{P} = \{p_i \mid i \in \mathbb{N}\}$

Poznámka: *Konečnou* teorii by bylo možné (byť ne praktické!) nahradit jediným výrokem: konjunkcí všech axiomů.

Připouštíme ale i *nekonečné teorie*; hodí se např. pro popis systému v (diskrétním) čase $t = 0, 1, 2, \dots$

2.2 Sémantika výrokové logiky

Pravdivostní hodnota: příklad

pravdivostní ohodnocení **výrokových proměnných** jednoznačně určuje pravdivostní hodnotu výroku (vyhodnoť od listů ke kořeni)

$$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$$

Pravdivostní hodnota: příklad

pravdivostní ohodnocení **výrokových proměnných** jednoznačně určuje pravdivostní hodnotu výroku (vyhodnoť od listů ke kořeni)

$$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$$

(a) φ **platí** při ohodnocení

$$p = 0, q = 0, r = 0$$

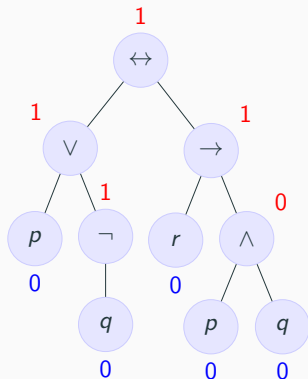
Pravdivostní hodnota: příklad

pravdivostní ohodnocení **výrokových proměnných** jednoznačně určuje pravdivostní hodnotu výroku (vyhodnoť od listů ke kořeni)

$$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$$

(a) φ **platí** při ohodnocení

$$p = 0, q = 0, r = 0$$



Pravdivostní hodnota: příklad

pravdivostní ohodnocení **výrokových proměnných** jednoznačně určuje pravdivostní hodnotu výroku (vyhodnoť od listů ke kořeni)

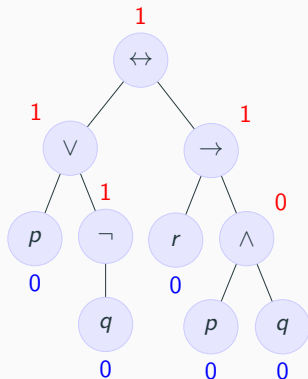
$$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$$

(a) φ **platí** při ohodnocení

$$p = 0, q = 0, r = 0$$

(b) φ **neplatí** při ohodnocení

$$p = 1, q = 0, r = 1$$



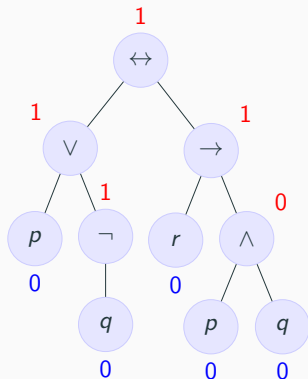
Pravdivostní hodnota: příklad

pravdivostní ohodnocení **výrokových proměnných** jednoznačně určuje pravdivostní hodnotu výroku (vyhodnoť od listů ke kořeni)

$$\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$$

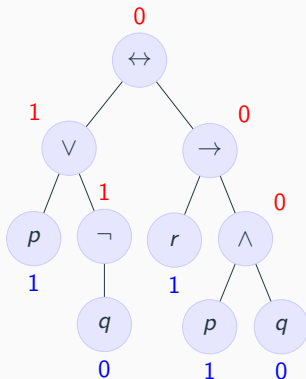
(a) φ **platí** při ohodnocení

$$p = 0, q = 0, r = 0$$



(b) φ **neplatí** při ohodnocení

$$p = 1, q = 0, r = 1$$



Sémantika logických spojek

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

Sémantika logických spojek

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

$$\begin{array}{c|c} 0 & 1 \\ 1 & 0 \end{array} \quad f_{\neg}(x) = 1 - x$$

$$\begin{array}{c|c} 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 1 & 1 \end{array} \quad f_{\wedge}(x, y) = \min(x, y)$$

$$\begin{array}{c|c} 0 & 1 \\ 0 & 0 \\ 1 & 1 \\ 1 & 1 \end{array} \quad f_{\vee}(x, y) = \max(x, y)$$

$$\begin{array}{c|c} 0 & 1 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{array} \quad f_{\rightarrow}(x, y)$$

$$\begin{array}{c|c} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{array} \quad f_{\leftrightarrow}(x, y)$$

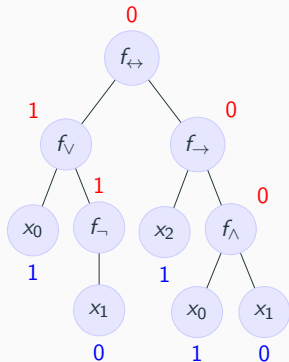
sémantika logických spojek je daná booleovskými funkcemi, každý výrok určuje *složenou* booleovskou funkci, tzv. **pravdivostní funkci**

Výroky a booleovské funkce

sémantika logických spojek je daná booleovskými funkcemi, každý výrok určuje *složenou* booleovskou funkci, tzv. **pravdivostní funkci**

např. $\varphi = ((p \vee (\neg q)) \leftrightarrow (r \rightarrow (p \wedge q)))$ v jazyce $\mathbb{P}' = \{p, q, r, s\}$

$$f_{\varphi, \mathbb{P}'}(x_0, x_1, x_2, x_3) = f_{\leftrightarrow}(f_{\vee}(x_0, f_{\neg}(x_1)), f_{\rightarrow}(x_2, f_{\wedge}(x_0, x_1)))$$



pravdivostní hodnota φ při ohodnocení
 $p = 1, q = 0, r = 1, s = 1$:

$$\begin{aligned} f_{\varphi, \mathbb{P}'}(1, 0, 1, 1) &= f_{\leftrightarrow}(f_{\vee}(1, f_{\neg}(0)), f_{\rightarrow}(1, f_{\wedge}(1, 0))) \\ &= f_{\leftrightarrow}(f_{\vee}(1, 1), f_{\rightarrow}(1, 0)) \\ &= f_{\leftrightarrow}(1, 0) \\ &= 0 \end{aligned}$$

Pravdivostní funkce formálně

Pravdivostní funkce výroku φ v konečném jazyce \mathbb{P} je funkce $f_{\varphi, \mathbb{P}}: \{0, 1\}^{|\mathbb{P}|} \rightarrow \{0, 1\}$ definovaná induktivně:

- je-li φ i -tý prvovýrok z \mathbb{P} : $f_{\varphi, \mathbb{P}}(x_0, \dots, x_{n-1}) = x_i$
- je-li $\varphi = (\neg\varphi')$: $f_{\varphi, \mathbb{P}}(x_0, \dots, x_{n-1}) = f_{\neg}(f_{\varphi', \mathbb{P}}(x_0, \dots, x_{n-1}))$
- je-li $(\varphi' \square \varphi'')$ kde $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$: $f_{\varphi, \mathbb{P}}(x_0, \dots, x_{n-1}) = f_{\square}(f_{\varphi', \mathbb{P}}(x_0, \dots, x_{n-1}), f_{\varphi'', \mathbb{P}}(x_0, \dots, x_{n-1}))$

Pravdivostní funkce formálně

Pravdivostní funkce výroku φ v *konečném* jazyce \mathbb{P} je funkce $f_{\varphi, \mathbb{P}}: \{0, 1\}^{|\mathbb{P}|} \rightarrow \{0, 1\}$ definovaná induktivně:

- je-li φ i -tý prvovýrok z \mathbb{P} : $f_{\varphi, \mathbb{P}}(x_0, \dots, x_{n-1}) = x_i$
- je-li $\varphi = (\neg\varphi')$: $f_{\varphi, \mathbb{P}}(x_0, \dots, x_{n-1}) = f_{\neg}(f_{\varphi', \mathbb{P}}(x_0, \dots, x_{n-1}))$
- je-li $(\varphi' \square \varphi'')$ kde $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$: $f_{\varphi, \mathbb{P}}(x_0, \dots, x_{n-1}) = f_{\square}(f_{\varphi', \mathbb{P}}(x_0, \dots, x_{n-1}), f_{\varphi'', \mathbb{P}}(x_0, \dots, x_{n-1}))$

Poznámka: Pravdivostní funkce $f_{\varphi, \mathbb{P}}$ závisí pouze na proměnných odpovídajících prvovýrokům z $\text{Var}(\varphi) \subseteq \mathbb{P}$.

Je-li výrok v *nekonečném* jazyce \mathbb{P} , můžeme se omezit na jazyk $\text{Var}(\varphi)$ (který je konečný) a uvažovat pravdivostní funkci nad ním.

Pravdivostní ohodnocení reprezentuje 'reálný svět' (systém) v námi zvoleném 'formálním světě', proto mu také říkáme **model**

¹Formálně ztotožňujeme $\{0, 1\}^{\mathbb{P}}$ s $\{0, 1\}^{|\mathbb{P}|}$, množina \mathbb{P} je uspořádaná.

Pravdivostní ohodnocení reprezentuje 'reálný svět' (systém) v námi zvoleném 'formálním světě', proto mu také říkáme **model**

Model jazyka \mathbb{P} : libovolné pravdivostní ohodnocení $v: \mathbb{P} \rightarrow \{0, 1\}$

Množina všech modelů: $M_{\mathbb{P}} = \{v \mid v: \mathbb{P} \rightarrow \{0, 1\}\} = \{0, 1\}^{\mathbb{P}}$

¹Formálně ztotožňujeme $\{0, 1\}^{\mathbb{P}}$ s $\{0, 1\}^{|\mathbb{P}|}$, množina \mathbb{P} je uspořádaná.

Pravdivostní ohodnocení reprezentuje 'reálný svět' (systém) v námi zvoleném 'formálním světě', proto mu také říkáme **model**

Model jazyka \mathbb{P} : libovolné pravdivostní ohodnocení $v: \mathbb{P} \rightarrow \{0, 1\}$

Množina všech modelů: $M_{\mathbb{P}} = \{v \mid v: \mathbb{P} \rightarrow \{0, 1\}\} = \{0, 1\}^{\mathbb{P}}$

$\mathbb{P} = \{p, q, r\}$, ohodnocení p je pravda, q nepravda, a r pravda:
formálně $v = \{(p, 1), (q, 0), (r, 1)\}$ ale píšeme¹ jen $v = (1, 0, 1)$

$$M_{\mathbb{P}} = \{(0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1), \\ (1, 0, 0), (1, 0, 1), (1, 1, 0), (1, 1, 1)\}$$

¹Formálně ztotožňujeme $\{0, 1\}^{\mathbb{P}}$ s $\{0, 1\}^{|\mathbb{P}|}$, množina \mathbb{P} je uspořádaná.

výrok platí v modelu, pokud je jeho pravdivostní hodnota rovna 1

výrok platí v modelu, pokud je jeho pravdivostní hodnota rovna 1

Výrok φ v jazyce \mathbb{P} , model $v \in M_{\mathbb{P}}$. Pokud $f_{\varphi, \mathbb{P}}(v) = 1$, potom říkáme, že φ **platí** v modelu v , v je **modelem** φ , a píšeme $v \models \varphi$.

výrok platí v modelu, pokud je jeho pravdivostní hodnota rovna 1

Výrok φ v jazyce \mathbb{P} , model $v \in M_{\mathbb{P}}$. Pokud $f_{\varphi, \mathbb{P}}(v) = 1$, potom říkáme, že φ **platí** v modelu v , v je **modelem** φ , a píšeme $v \models \varphi$.

Množina všech modelů resp. *nemodelů* φ :

$$M_{\mathbb{P}}(\varphi) = \{v \in M_{\mathbb{P}} \mid v \models \varphi\} = f_{\varphi, \mathbb{P}}^{-1}[1]$$

$$\overline{M_{\mathbb{P}}(\varphi)} = M_{\mathbb{P}} \setminus M_{\mathbb{P}}(\varphi) = \{v \in M_{\mathbb{P}} \mid v \not\models \varphi\} = f_{\varphi, \mathbb{P}}^{-1}[0]$$

výrok platí v modelu, pokud je jeho pravdivostní hodnota rovna 1

Výrok φ v jazyce \mathbb{P} , model $v \in M_{\mathbb{P}}$. Pokud $f_{\varphi, \mathbb{P}}(v) = 1$, potom říkáme, že φ **platí** v modelu v , v je **modelem** φ , a píšeme $v \models \varphi$.

Množina všech modelů resp. *nemodelů* φ :

$$M_{\mathbb{P}}(\varphi) = \{v \in M_{\mathbb{P}} \mid v \models \varphi\} = f_{\varphi, \mathbb{P}}^{-1}[1]$$
$$\overline{M_{\mathbb{P}}(\varphi)} = M_{\mathbb{P}} \setminus M_{\mathbb{P}}(\varphi) = \{v \in M_{\mathbb{P}} \mid v \not\models \varphi\} = f_{\varphi, \mathbb{P}}^{-1}[0]$$

Je-li jazyk zřejmý z kontextu, můžeme vynechat, ale jinak ne!

$$M_{\{p, q\}}(p \rightarrow q) = \{(0, 0), (0, 1), (1, 1)\}$$

$$M_{\{p, q, r\}}(p \rightarrow q) = \{(0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1), (1, 1, 0), (1, 1, 1)\}$$

Platnost teorie, model teorie

Teorie T **platí** v modelu v , pokud každý axiom $\varphi \in T$ platí ve v .
Podobně jako pro výrok: v je **modelem** T , $v \models T$, $v \in M_{\mathbb{P}}(T)$.

Platnost teorie, model teorie

Teorie T **platí** v modelu v , pokud každý axiom $\varphi \in T$ platí ve v .

Podobně jako pro výrok: v je **modelem** T , $v \models T$, $v \in M_{\mathbb{P}}(T)$.

Někdy píšeme $M_{\mathbb{P}}(T, \varphi)$ místo $M_{\mathbb{P}}(T \cup \{\varphi\})$, $M_{\mathbb{P}}(\varphi_1, \varphi_2, \dots, \varphi_n)$ místo $M_{\mathbb{P}}(\{\varphi_1, \varphi_2, \dots, \varphi_n\})$.

- $M_{\mathbb{P}}(T, \varphi) = M_{\mathbb{P}}(T) \cap M_{\mathbb{P}}(\varphi)$
- $M_{\mathbb{P}}(T) = \bigcap_{\varphi \in T} M_{\mathbb{P}}(\varphi)$
- $M_{\mathbb{P}}(\varphi_1) \supseteq M_{\mathbb{P}}(\varphi_1, \varphi_2) \supseteq \dots \supseteq M_{\mathbb{P}}(\varphi_1, \varphi_2, \dots, \varphi_n)$

Platnost teorie, model teorie

Teorie T **platí** v modelu v , pokud každý axiom $\varphi \in T$ platí ve v .

Podobně jako pro výrok: v je **modelem** T , $v \models T$, $v \in M_{\mathbb{P}}(T)$.

Někdy píšeme $M_{\mathbb{P}}(T, \varphi)$ místo $M_{\mathbb{P}}(T \cup \{\varphi\})$, $M_{\mathbb{P}}(\varphi_1, \varphi_2, \dots, \varphi_n)$ místo $M_{\mathbb{P}}(\{\varphi_1, \varphi_2, \dots, \varphi_n\})$.

- $M_{\mathbb{P}}(T, \varphi) = M_{\mathbb{P}}(T) \cap M_{\mathbb{P}}(\varphi)$
- $M_{\mathbb{P}}(T) = \bigcap_{\varphi \in T} M_{\mathbb{P}}(\varphi)$
- $M_{\mathbb{P}}(\varphi_1) \supseteq M_{\mathbb{P}}(\varphi_1, \varphi_2) \supseteq \dots \supseteq M_{\mathbb{P}}(\varphi_1, \varphi_2, \dots, \varphi_n)$

Najděme modely $T = \{p \vee q \vee r, q \rightarrow r, \neg r\}$ (v jazyce $\mathbb{P} = \{p, q, r\}$):

$$M_{\mathbb{P}}(\neg r) = \{(0, 0, 0), (0, 1, 0), (1, 0, 0), (1, 1, 0)\}$$

$$M_{\mathbb{P}}(\neg r, q \rightarrow r) = \{(0, 0, 0), (1, 0, 0)\}$$

$$M_{\mathbb{P}}(T) = \{(1, 0, 0)\}$$