Úvod do teoretické informatiky

Zdeněk Sawa

Katedra informatiky, FEI,
Vysoká škola báňská – Technická universita Ostrava
17. listopadu 15, Ostrava-Poruba 708 33
Česká republika

15. února 2012

Přednášející

Garant předmětu:

Jméno: Ing. Zdeněk Sawa, Ph.D.

E-mail: zdenek.sawa@vsb.cz

Místnost: A1024

Tutor:

Jméno: Ing. Martin Kot, Ph.D.

E-mail: martin.kot@vsb.cz

Místnost: A1024

WWW stránky

Webové stránky k předmětu naleznete na adrese:

http://www.cs.vsb.cz/sawa/uti

Na těchto stránkách najdete:

- Informace o předmětu
- Učební texty
- Slidy z přednášek
- Zadání příkladů na cvičení
- Aktuální informace
- Animace

Webové stránky se specifickými informacemi pro kombinované studenty:

http://www.cs.vsb.cz/sawa/kot/UTIkomb

Požadavky

- **Zápočet** (22 bodů):
  - Zápočtová písemka (22 bodů) — bude se psát na 5. tutoriálu

  Minimum pro získání zápočtu je 7 bodů.

- **Zkouška** (78 bodů)
  - Písemná zkouška skládající se ze tří částí po 26 bodech, přičemž z každé části je nutné získat nejméně 10 bodů.

Uznané zápočty

- Studenti, kteří předmět opakují a mají nárok na uznání zápočtu, ale **nemají** ho dosud v Edisonu zapsaný a **chtějí** ho uznat, musí v průběhu prvních dvou týdnů semestru požádat svého cvičícího o uznání zápočtu.
- Podobně studenti, kteří mají nárok na uznání zápočtu, **mají** v Edisonu zapsaný, ale **nechtějí** ho uznat, musí v průběhu prvních dvou týdnů semestru požádat svého cvičícího o jeho zrušení.
- Studenti, kteří mají uznaný zápočet, nebudou psát zápočtovou písemku.

Náplň předmětu

Cílem tohoto předmětu je poskytnout studentům stručný úvod do následujících oblastí:

- **Logika**
- **Formální jazyky a automaty**
- **Vyčíslitelnost a složitost**

Výukové materiály

Hlavními výukovými texty jsou:

- prof. RNDr. Petr Jančar, CSc.
  Úvod do teoretické informatiky (učební text),
  VŠB-TU Ostrava, 2007.

  Poznámka: Pro zájemce s hlubším zájmem o problematiku je k dispozici i rozšířená verze tohoto textu určená pro studenty magisterského studia (pro předmět Teoretická informatika).

- doc. RNDr. Marie Duží, CSc.
  Matematická logika (učební text),
  VŠB-TU Ostrava, 2003.

Výukové materiály

Kromě výukových textů jsou k dispozici:

- **Slidy** z přednášek (na web budou doplňovány aktuální verze)
- **Animace** vytvořené M. Kotem, Z. Sawou a některými studenty v rámci diplomových prací
- **Zadání příkladů do cvičení**

Další literatura (pro zájemce)

- M. Sipser: *Introduction to the Theory of Computation*, PWS Publishing Company, 1997.
- D. Kozen: *Automata and Computability*, Undergraduate Text in Computer Science, Springer Verlag, 1997.
- Ch. Papadimitriou: *Computational Complexity*, Addison-Wesley, 1993.
- J. E. Hopcroft, R. Motwani, J. D. Ullman: *Introduction to Automata Theory, Languages, and Computation* (3rd Edition), Addison Wesley, 2006.
- H.D. Ebbinghaus, J. Flum, W. Thomas: *Mathematical Logic* (2nd edition), Springer, 1994.
- M. Huth, M. Ryan.: *Logic in Computer Science: Modelling and Reasoning about Systems*, Cambridge University Press, 2004.
- V. Švejdar: *Logika - neúplnost, složitost a nutnost*, Academia, 2002.

Další literatura (pro zájemce)

- Ding-Zhu Du, Ker-I Ko: *Problem Solving in Automata, Languages, and Complexity*, Wiley, 2001. Pozn.: v rámci sítě VŠB je tato publikace dostupná v elektronické podobě (jako PDF) na adrese `http://knihovna.vsb.cz/sluzby/e-knihy-wiley.htm`

Základní pojmy

Množiny

Množina – kolekce vzájemně odlišitelných objektů, které nazýváme jejími **prvky**.

- $x \in S$ – objekt x je prvkem množiny S
- $x \notin S$ – objekt x není prvkem množiny S

Jednou z možností, jak popsat množinu, je explicitně vyjmenovat všechny její prvky, např.:

$$S = \{1, 2, 3\}$$

Množina nemůže obsahovat prvek více než jednou a prvky množiny nejsou nijak seřazeny.

Množiny

Množiny A a B jsou si **rovny** ($A = B$), jestliže obsahují tytéž prvky.

Například

$$\{1, 2, 3\} = \{2, 1, 3\} = \{3, 2, 1\}$$

Množina neobsahující žádné prvky se nazývá **prázdná množina** a označuje se symbolem $\emptyset$.

Poznámka: Kromě množin se také někdy používají **multimnožiny**. Na rozdíl od množiny může multimnožina obsahovat více výskytů jednoho prvku.

$$M = \{1, 1, 1, 2, 3, 3, 3, 3, 3\}$$

Množiny

Příklady některých důležitých množin:

- $\mathbb{N}$ – množina všech **přirozených** čísel, tj. $\mathbb{N} = \{0, 1, 2, \ldots\}$,
- $\mathbb{N}_+$ – množina všech **kladných přirozených** čísel, tj. $\mathbb{N}_+ = \{1, 2, 3, \ldots\}$,
- $\mathbb{Z}$ – množina všech **celých** čísel, tj. $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$,
- $\mathbb{Q}$ – množina všech **racionálních** čísel (zlomky),
- $\mathbb{R}$ – množina všech **reálných** čísel.

Množiny

- $A \subseteq B$ – označuje, že A je **podmnožinou** B, tj.

$$\forall x(x \in A \Rightarrow x \in B)$$

(každý prvek množiny A patří rovněž do množiny B).

- $A \subset B$ – označuje, že A je **vlastní podmnožinou** B, tj.

$$A \subseteq B \wedge \exists x(x \in B \wedge x \notin A)$$

(tj. $A \subseteq B$, ale $A \neq B$).

Poznámka: Někdy se též používá zápis $A \subset B$ pro označení, že A je podmnožinou B (tj. připouští i možnost $A = B$), a zápis $A \subsetneq B$ pro označení, že A je vlastní podmnožinou B.

Množiny

Pro danou množinu A můžeme definovat množinu $B \subseteq A$ tvořenou těmi prvky množiny A, které mají určitou vlastnost (splňují nějakou podmínku) $\varphi(x)$.

$$B = \{x \in A \mid \varphi(x)\}$$

Příklad: Podmnožina X množiny přirozených čísel $\mathbb{N}$ tvořená těmi čísly, která dávají po dělení pěti zbytek dvě.

$$X = \{x \in \mathbb{N} \mid x \bmod 5 = 2\}$$

Množiny

Množinové operace:

- **Průnik** množin A a B je množina

$$A \cap B = \{x \mid x \in A \wedge x \in B\}$$

- **Sjednocení** množin A a B je množina

$$A \cup B = \{x \mid x \in A \vee x \in B\}$$

- **Rozdíl** množin A a B je množina

$$A - B = \{x \mid x \in A \wedge x \notin B\}$$

Poznámka: Pro rozdíl množin se též používá zápis $A \setminus B$.

Příklad: Jestliže $A = \{a, b, c, d\}$ a $B = \{b, c, e, f\}$, pak

$$A \cap B = \{b, c\}, \qquad A \cup B = \{a, b, c, d, e, f\}, \qquad A - B = \{a, d\}.$$

Množiny

Někdy jsou všechny množiny, které uvažujeme, podmnožinami nějaké jedné množiny U nazývané **universum**.

Příklad: Pokud se například bavíme o množinách přirozených čísel, pak je universem množina $\mathbb{N}$.

Pro dané universum U definujeme **doplněk** množiny A jako

$$\overline{A} = U - A$$

Pro libovolné množiny $A, B \subseteq U$ platí de Morganova pravidla:

$$\overline{A \cap B} = \overline{A} \cup \overline{B} \qquad \overline{A \cup B} = \overline{A} \cap \overline{B}$$

Množiny A a B jsou **disjunktní**, jestliže nemají žádný společný prvek, tj. jestliže $A \cap B = \emptyset$.

Množiny

Velikost dané množiny S se nazývá její **kardinalita** a označuje se $|S|$.

- V případě **konečné** množiny, je její kardinalita přirozené číslo odpovídající počtu jejích prvků, např. $|\emptyset| = 0$.
- Dvě (obecné) množiny mají stejnou kardinalitu, jesliže existuje bijekce (tj. vzájemně jednoznačné zobrazení) mezi jejich prvky.
- Množina S se nazývá **spočetná**, jestliže existuje bijekce mezi S a $\mathbb{N}$. Spočetné množiny jsou „nejmenší“ mezi nekonečnými množinami.
- Nekonečná množina, která není spočetná, se nazývá **nespočetná**.

Příklad: Množiny $\mathbb{N}$, $\mathbb{Z}$ a $\mathbb{Q}$ jsou spočetné, množina $\mathbb{R}$ je nespočetná.

Množiny

Množina všech podmnožin množiny S se nazývá **potenční množina** množiny S a označuje se zápisem $\mathcal{P}(S)$.

Příklad: Pokud $S = \{a, b, c\}$, pak

$$\mathcal{P}(S) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}.$$

Pokud je množina S konečná, pak $|\mathcal{P}(S)| = 2^{|S|}$.

Poznámka: Často se také používá pro označení potenční množiny místo $\mathcal{P}(S)$ výraz 2^S.

Množiny

Uspořádaná dvojice prvků a a b se označuje (a, b).

Na rozdíl od množiny u uspořádané dvojice záleží na pořadí prvků, (a, b) je něco jiného než (b, a).

Poznámka: Formálně je možno uspořádanou dvojici (a, b) pomocí množin např. takto:

$$(a, b) = \{a, \{a, b\}\}$$

Analogicky můžeme definovat uspořádané trojice, čtveřice atd.

Kartézský součin množin A a B, označovaný $A \times B$, je množina všech uspořádaných dvojic, kde první prvek z dvojice patří do množiny A a druhý do množiny B:

$$A \times B = \{(a, b) \mid a \in A,\ b \in B\}$$

Příklad: $\{a, b\} \times \{a, b, c\} = \{(a, a), (a, b), (a, c), (b, a), (b, b), (b, c)\}$

Množiny

Jestliže A a B jsou konečné množiny, pak $|A \times B| = |A| \cdot |B|$.

Kartézský součin n množin $A_1, A_2, \ldots, A_n$ je množina ***n*-tic**

$$A_1 \times A_2 \times \cdots \times A_n = \{(a_1, a_2, \ldots, a_n) \mid a_i \in A_i,\ i = 1, 2, \ldots, n\}$$

Jestliže všechna A_i jsou konečné množiny, platí

$$|A_1 \times A_2 \times \cdots \times A_n| = |A_1| \cdot |A_2| \cdots |A_n|$$

Místo kartézského součinu $A \times A \times \cdots \times A$, kde se množina A vyskytuje n krát, píšeme A^n.

Pro konečnou množinu A platí $|A^n| = |A|^n$.

Relace

Relace na množinách $A_1, A_2, \ldots, A_n$ je libovolná podmnožina kartézského součinu $A_1 \times A_2 \times \cdots \times A_n$.

Relace na n množinách se nazývá **n-ární** relace.

Jestliže $n = 2$, jedná se o **binární** relaci.
Jestliže $n = 3$, jedná se o **ternární** relaci.

V případě, že $A_1 = A_2 = \cdots = A_n$, hovoříme o **homogenní** relaci, v opačném případě o relaci **heterogenní**.

Když říkáme, že R je n-ární relace na množině A, máme tím na mysli, že $R \subseteq A^n$.

Relace

Příklad: Relace „menší než“ na množině přirozených čísel je množina

$$\{(a, b) \in \mathbb{N} \times \mathbb{N} \mid a < b\}$$

Poznámka: Jestliže $R \subseteq A \times B$ je binární relace, někdy místo $(a, b) \in R$ používáme infixový zápis a píšeme $a\ R\ b$.

Relace

Binární relace $R \subseteq A \times A$ je:

- **reflexivní**, jestliže pro všechna $a \in A$ platí $(a, a) \in R$,
- **ireflexivní**, jestliže pro všechna $a \in A$ platí $(a, a) \notin R$,
- **symetrická**, jestliže pro všechna $a, b \in A$ platí, že pokud $(a, b) \in R$, pak $(b, a) \in R$,
- **asymetrická**, jestliže pro všechna $a, b \in A$ platí, že pokud $(a, b) \in R$, pak $(b, a) \notin R$,
- **antisymetrická**, jestliže pro všechna $a, b \in A$ platí, že pokud $(a, b) \in R$ a $(b, a) \in R$, pak $a = b$,
- **tranzitivní**, jestliže pro všechna $a, b, c \in A$ platí, že pokud $(a, b) \in R$ a $(b, c) \in R$, pak $(a, c) \in R$.

Relace

Příklad:

- Relace "$=$" na $\mathbb{N}$ je reflexivní, symetrická, antisymetrická a tranzitivní, ale není ireflexivní ani asymetrická.
- Relace "$\leq$" na $\mathbb{N}$ je reflexivní, antisymetrická a tranzitivní, ale není ireflexivní, symetrická ani asymetrická.
- Relace "$<$" na $\mathbb{N}$ je ireflexivní, asymetrická, antisymetrická a tranzitivní, ale není reflexivní ani symetrická.

Relace

- **Reflexivní uzávěr** relace $R \subseteq A \times A$ je nejmenší reflexivní relace $R' \subseteq A \times A$ taková, že $R \subseteq R'$.

  Poznámka: Pojmem „nejmenší“ zde máme na mysli to, že neexistuje žádná reflexivní relace R'' taková, že $R \subseteq R'' \subset R'$.

- **Symetrický uzávěr** relace $R \subseteq A \times A$ je nejmenší symetrická relace $R' \subseteq A \times A$ taková, že $R \subseteq R'$.

- **Tranzitivní uzávěr** relace $R \subseteq A \times A$ je nejmenší tranzitivní relace $R' \subseteq A \times A$ taková, že $R \subseteq R'$.

- **Reflexivní a tranzitivní uzávěr** relace $R \subseteq A \times A$ je nejmenší relace $R' \subseteq A \times A$ taková, že $R \subseteq R'$ a R' je současně reflexivní i tranzitivní.

Ekvivalence

Binární relace R na množině A je **ekvivalence** právě tehdy, když je reflexivní, symetrická a tranzitivní.

Příklad: Následující relace $\equiv_5$ je ekvivalence

$$\equiv_5 = \{(a, b) \in \mathbb{Z}^2 \mid (a \bmod 5) = (b \bmod 5)\}$$

obecně pro libovolné $n > 0$ je relace $\equiv_n$ ekvivalence

$$\equiv_n = \{(a, b) \in \mathbb{Z}^2 \mid (a \bmod n) = (b \bmod n)\}$$

Ekvivalence

Jestliže R je ekvivalence na množině A, pak **třídou ekvivalence** prvku $a \in A$ je množina $[a]_R = \{b \in A \mid (a, b) \in R\}$, tj. množina všech prvků s ním ekvivalentních.

Mějme množinu A. Množina jejích podmnožin $\mathcal{A} = \{A_i \mid i \in I\}$ (pro nějakou indexovou množinu I) tvoří **rozklad** na množině A, jestliže:

- všechny množiny A_i jsou vzájemně disjunktní, tj. jestliže pro libovolné $A_i, A_j \in \mathcal{A}$ platí $A_i \cap A_j = \emptyset$ pokud $i \neq j$, a
- sjednocení množin z $\mathcal{A}$ je množina A, tj.

$$A = \bigcup_{A_i \in \mathcal{A}} A_i$$

Ekvivalence

Ekvivalence $R \subseteq A \times A$ definuje na A rozklad $\{\, [a]_R \mid a \in A \,\}$.

Naopak rozklad $\mathcal{A} = \{A_i \mid i \in I\}$ na množině A definuje ekvivalenci

$$R = \{(a, b) \subseteq A \times A \mid a, b \in A_i \text{ pro nějaké } A_i \in \mathcal{A}\}\,.$$

Příklad: Ekvivalence $\equiv_5$ definuje rozklad $\mathcal{A} = \{A_0, A_1, A_2, A_3, A_4\}$ na $\mathbb{N}$, kde

- $A_0 = \{\ldots, -15, -10, -5, 0, 5, 10, 15, \ldots\}$
- $A_1 = \{\ldots, -14, -9, -4, 1, 6, 11, 16, \ldots\}$
- $A_2 = \{\ldots, -13, -8, -3, 2, 7, 12, 17, \ldots\}$
- $A_3 = \{\ldots, -12, -7, -2, 3, 8, 13, 18, \ldots\}$
- $A_4 = \{\ldots, -11, -6, -1, 4, 9, 14, 19, \ldots\}$

Uspořádání

Binární relace R na množině A je **(částečné a neostré) uspořádání**, jestliže je reflexivní, tranzitivní a antisymetrická.

Binární relace R na množině A je **(částečné) ostré uspořádání**, jestliže je asymetrická a tranzitivní.
(Pozn.: Z toho, že je R asymetrická plyne, že je také ireflexivní a antisymetrická.)

Pro neostrá uspořádání se obvykle používají symboly jako $\leq$ a jemu podobné, pro ostrá uspořádání pak symboly jako $<$ a jemu podobné.

Uspořádání (ať už neostré či ostré) $R \subseteq A \times A$ je **úplné** (nebo také **lineární**), jestliže pro všechna $a, b \in A$ platí buď $(a, b) \in R$, $(b, a) \in R$ nebo $a = b$ (tj. pokud neexistují vzájemně nesrovnatelné prvky).

Uspořádání

Příklady:

- Relace "$\leq$" je úplné (neostré) uspořádání na množině $\mathbb{N}$ ($\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$).
- Relace "$<$" je ostré úplné uspořádání na množině $\mathbb{N}$ ($\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$).
- Relace "$\subseteq$" je částečné (neostré) uspořádání na množině $\mathcal{P}(X)$ (pro libovolnou množinu X).
- Relace "je dělitelem" je částečné (neostré) uspořádání na množině $\mathbb{N}_+$.
- Relace "$=$" je částečné (neostré) uspořádání na množině $\mathbb{N}$.

Ke každému neostrému uspořádání R na množině A existuje odpovídající ostré uspořádání $R' = R - \{(a, a) \mid a \in A\}$.

Naopak ke každému ostrému uspořádání S na množině A existuje odpovídající neostré uspořádání $S' = S \cup \{(a, a) \mid a \in A\}$

Uspořádání

Mějme libovolné neostré uspořádání $\leq$ na množině A.

- Prvek $a \in A$ je **minimální prvek** množiny A, jestliže v A neexistuje menší prvek než a (tj. z $x \leq a$ plyne $x = a$).
- Prvek $a \in A$ je **maximální prvek** množiny A, jestliže v A neexistuje větší prvek než a (tj. z $a \leq x$ plyne $a = x$).
- Prvek $a \in A$ je **nejmenší prvek** množiny A, jestliže je menší než všechny ostatní prvky v A (tj. pro každé $x \in A$ platí $a \leq x$).
- Prvek $a \in A$ je **největší prvek** množiny A, jestliže je větší než všechny ostatní prvky v A (tj. pro každé $x \in A$ platí $x \leq a$).

Uspořádání

- Prvek $a \in A$ je **infimum** množiny B (píšeme $a = \inf B$), jestliže a je největší ze všech prvků, které jsou menší než všechny prvky z B, tj. platí

$$(\forall x \in B)(a \leq x) \ \wedge \ (\forall b)((\forall x \in B)(b \leq x) \Rightarrow b \leq a)$$

- Prvek $a \in A$ je **supremum** množiny B (píšeme $a = \sup B$), jestliže a je nejmenší ze všech prvků, které jsou větší než všechny prvky z B, tj. platí

$$(\forall x \in B)(x \leq a) \ \wedge \ (\forall b)((\forall x \in B)(x \leq b) \Rightarrow a \leq b)$$

Funkce

Funkce f z množiny A do množiny B je binární relace $f \subseteq A \times B$ taková, že pro každé $a \in A$ existuje právě jedno $b \in B$ takové, že $(a, b) \in f$.

Množina A se nazývá **definiční obor** funkce f,

množina B se nazývá **obor hodnot** funkce f.

To, že f je funkce z množiny A do množiny B obvykle zapisujeme jako

$$f : A \to B$$

Místo $(a, b) \in f$ obvykle píšeme $b = f(a)$, neboť volbou prvku a je prvek b jednoznačně určen.

Funkce $f : A \to B$ tedy každému prvku z A přiřazuje právě jeden prvek z B.

Jestliže $b = f(a)$, říkáme, že a je **argumentem** funkce f a že b je **hodnotou** funkce f v bodě a.

Funkce

Výše uvedená definice se týká tzv. **totální** funkce, tj. funkce, jejíž hodnota je definovaná pro každou hodnotu argumentu.

Někdy má smysl uvažovat také tzv. **částečné (parciální) funkce**, tj. funkce, jejichž hodnota není pro některé hodnoty argumentu definována.

Formálně je částečná funkce $f : A \to B$ definována jako relace $f \subseteq A \times B$ taková, že pro každé $a \in A$ existuje nejvýše jedno $b \in B$ takové, že $(a, b) \in f$.

Poznámka: Pokud budeme mluvit o funkci a neuvedeme jinak, budeme mít vždy na mysli funkci totální.

Funkce

Konečná posloupnost (sekvence) délky n je funkce, jejímž definičním oborem je množina $\{0, 1, \ldots, n-1\}$.

Konečnou posloupnost obvykle zapisujeme tak, že vypíšeme její hodnoty:

$$f(0), f(1), \ldots, f(n-1)$$

Nekonečná posloupnost (sekvence) je funkce, jejímž definičním oborem je $\mathbb{N}$.

Nekonečnou posloupnost někdy zapisujeme tak, že uvedeme několik prvních prvků, za kterými následují tři tečky:

$$f(0), f(1), f(2), \ldots$$

Funkce

Jestliže definičním oborem funkce f je kartézský součin, obvykle vynecháváme jeden pár závorek v zápise argumentu funkce f.

Pokud například máme funkci $f : A_1 \times A_2 \times \cdots \times A_n \to B$, pak místo $b = f((a_1, a_2, \ldots, a_n))$ píšeme $b = f(a_1, a_2, \ldots, a_n)$.

Místo o funkci někdy též mluvíme o **operaci**.

n-ární operace je funkce f typu

$$f : A_1 \times A_2 \times \cdots \times A_n \to B$$

V případě, že $n = 2$, mluvíme o **binární** operaci.

- $f : A^n \to A$ – n-ární operace na množině A,
- $f : A \to A$ – unární operace na množině A,
- $f : A \times A \to A$ – binární operace na množině A.

Funkce

Mějme funkci $f : A^n \to A$. Množina $B \subseteq A$ je **uzavřená na operaci** f, jestliže z $a_1, a_2, \ldots, a_n \in B$ plyne, že $f(a_1, a_2, \ldots, a_n) \in B$.

Jestliže $f : A \to B$ je funkce a $b = f(a)$, pak někdy také říkáme, že b je **obrazem** a. Obrazem množiny $A' \subseteq A$ je množina

$$f(A') = \{b \in B \mid b = f(a) \text{ pro nějaké } a \in A'\}.$$

Funkce $f : A \to B$ je:

- **surjektivní** (je surjekcí, je zobrazením na), jestliže $f(A) = B$,
- emphinjektivní (je injekcí, je prostá), jestliže z $a \neq a'$ plyne $f(a) \neq f(a')$,
- **bijektivní** (je bijekcí, je vzájemně jednoznačným zobrazením), jestliže je současně surjektivní i injektivní.

Jestliže funkce f je bijekcí, pak funkce **inverzní** k funkci f, označovaná f^{-1}, je definována takto: $f^{-1}(b) = a$ právě když $f(a) = b$.

Funkce

Předpokládejme nyní funkci $f : A \times A \to A$.

- Funkce f je **asociativní**, jestliže pro libovolné prvky $a, b, c \in A$ platí
$$f(f(a, b), c) = f(a, f(b, c)) .$$
- Funkce f je **komutativní**, jestliže pro libovolné prvky $a, b \in A$ platí
$$f(a, b) = f(b, a) .$$
- Prvek $z \in A$ je **nulovým prvkem** vzhledem k funkci f, jestliže pro libovolné $a \in A$ platí
$$f(z, a) = f(a, z) = z .$$
- Prvek $e \in A$ je **jednotkovým prvkem** vzhledem k funkci f, jestliže pro libovolné $a \in A$ platí
$$f(e, a) = f(a, e) = a .$$

Poznámka: Dá se ukázat, že ke každé funkci existuje nejvýše jeden nulový a nejvýše jeden jednotkový prvek.

Jestliže k funkci f existuje jednotkový prvek e, pak $b \in A$ je **inverzním prvkem** k prvku $a \in A$ právě tehdy, když

$$f(a, b) = f(b, a) = e$$

Funkce

Pro funkce typu $f : A \times A \to A$ je často vhodnější používat infixovou notaci a používat jako název funkce nějaký speciální symbol.

Mějme například funkci

$$\otimes : A \times A \to A$$

Pak místo $\otimes(a, b)$ píšeme $a \otimes b$.

- Asociativita $\otimes$ pak znamená, že pro libovolné $a, b, c \in A$ platí

$$(a \otimes b) \otimes c = a \otimes (b \otimes c)$$

- a komutativita $\otimes$ znamená, že pro libovolné $a, b \in A$ platí

$$a \otimes b = b \otimes a$$

Příklad: Místo $+(x, y)$ píšeme $x + y$.

Logika

Výroková logika

Výroková logika analyzuje způsoby skládání jednoduchých (atomických) výroků do výroků složených pomocí logických spojek.
Výrok je tvrzení, o němž má smysl prohlásit, zda je pravdivé či nepravdivé.

Výroková logika – syntaxe

Výroky zapisujeme pomocí **formulí**, což jsou slova nad určitou **abecedou** vytvořená podle určitých pravidel.

V případě výrokové logiky je abeceda tvořena následujícími symboly:

- Výrokové symboly (atomické výroky): p, q, r, ...
- Logické spojky:
  - $\neg$ – negace
  - $\wedge$ – konjunkce
  - $\vee$ – disjunkce
  - $\supset$ – implikace
  - $\equiv$ – ekvivalence
- Závorky: $($, $)$, $[$, $]$, ...

Výroková logika – syntaxe

Množina všech dobře vytvořených formulí výrokové logiky je definována následovně:

1. Výrokové symboly p, q, r, ... jsou formule.
2. Jestliže φ, ψ jsou formule, pak i $\neg\varphi$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \supset \psi)$ a $(\varphi \equiv \psi)$ jsou formule.
3. Žádné další formule, než ty definované podle předchozích dvou bodů, neexistují.

Poznámka: Jedná se o induktivní definici.

Příklad:

- p, q, r jsou dobře utvořené formule
- $\neg r$, $(q \wedge r)$ jsou dobře utvořené formule
- $\neg(\neg(p \vee (q \supset \neg r)) \wedge q)$ je dobře utvořená formule

Výroková logika – syntaxe

Abychom nemuseli všude psát závorky, používají se následující konvence o vypouštění závorek:

- Vnější pár závorek je možno vypustit.
- Je definována následující priorita logických spojek (od největší po nejmenší): $\neg$, $\wedge$, $\vee$, $\supset$, $\equiv$
- Je možno využít toho, že $\wedge$ a $\vee$ jsou asociativní.

Příklad: Místo $((p \wedge (q \wedge r)) \supset s)$ můžeme psát $p \wedge q \wedge r \supset s$

Výroková logika – syntaxe

Poznámka: V literatuře se často používají pro logické spojky i jiné symboly:

Symbol	Alternativně
$\neg$	$\sim$
$\wedge$	$\&$
$\supset$	$\rightarrow$, $\Rightarrow$
$\equiv$	$\leftrightarrow$, $\Leftrightarrow$

Výroková logika – sémantika

Pravdivostní ohodnocení (valuace) je zobrazení ν, které každému výrokovému symbolu p přiřazuje pravdivostní hodnotu, tj. hodnotu z množiny $\{0, 1\}$.

Poznámka: 0 – nepravda (false), 1 – pravda (true)

Pravdivostní hodnotu, která je přiřazena formuli φ při daném pravdivostním ohodnocení ν označujeme $[\varphi]_\nu$ a definujeme:

- $[p]_\nu = \nu(p)$ pro výrokový symbol p,
- $[\neg\varphi]_\nu = 1$, právě když $[\varphi]_\nu = 0$,
- $[\varphi \wedge \psi]_\nu = 1$, právě když $[\varphi]_\nu = 1$ a $[\psi]_\nu = 1$,
- $[\varphi \vee \psi]_\nu = 1$, právě když $[\varphi]_\nu = 1$ nebo $[\psi]_\nu = 1$,
- $[\varphi \supset \psi]_\nu = 1$, právě když $[\varphi]_\nu = 0$ nebo $[\psi]_\nu = 1$,
- $[\varphi \equiv \psi]_\nu = 1$, právě když $[\varphi]_\nu = [\psi]_\nu$.

Výroková logika – sémantika

Výše popsaný význam logických spojek je možné znázornit pomocí následujících **pravdivostní tabulek**:

φ	$\neg\varphi$
0	1
1	0

φ	ψ	$\varphi \wedge \psi$
0	0	0
0	1	0
1	0	0
1	1	1

φ	ψ	$\varphi \vee \psi$
0	0	0
0	1	1
1	0	1
1	1	1

φ	ψ	$\varphi \supset \psi$
0	0	1
0	1	1
1	0	0
1	1	1

φ	ψ	$\varphi \equiv \psi$
0	0	1
0	1	0
1	0	0
1	1	1

Výroková logika – sémantika

Příklad: Vezměme si pravdivostní ohodnocení ν, kde $\nu(p) = 1$, $\nu(q) = 0$ a $\nu(r) = 1$, a formuli $\neg(\neg(p \vee (q \supset \neg r)) \wedge q)$:

- $[p]_\nu = 1$
- $[q]_\nu = 0$
- $[r]_\nu = 1$
- $[\neg r]_\nu = 0$
- $[q \supset \neg r]_\nu = 1$
- $[p \vee (q \supset \neg r)]_\nu = 1$
- $[\neg(p \vee (q \supset \neg r))]_\nu = 0$
- $[\neg(p \vee (q \supset \neg r)) \wedge q]_\nu = 0$
- $[\neg(\neg(p \vee (q \supset \neg r)) \wedge q)]_\nu = 1$

Výroková logika – sémantika

Neformální význam jednotlivých logických spojek:

- $\neg$ – „není pravda, že“
- $\wedge$ – „a“
- $\vee$ – „nebo“ (nevylučující)
- $\supset$ – „jestliže, pak“, „když, tak“, „je-li, pak“ apod.
- $\equiv$ – „právě tehdy, když“, „tehdy a jen tehdy“ apod.

Výroková logika – sémantika

- Formule φ je **splnitelná**, jestliže existuje pravdivostní ohodnocení ν takové, že $[\varphi]_\nu = 1$.
- Formule φ je **nesplnitelná (kontradikce)**, jestliže pro každé pravdivostní ohodnocení ν je $[\varphi]_\nu = 0$.
- Formule φ je **tautologie (logicky pravdivá)**, jestliže pro každé pravdivostní ohodnocení ν je $[\varphi]_\nu = 1$.

Poznámka: To, že formule φ je tautologie, označujeme zápisem $\models \varphi$.

Výroková logika – sémantika

Ověřit, zda daná formule φ je splnitelná, kontradikce, tautologie apod., můžeme tak, že vyzkoušíme všechna možná pravdivostní ohodnocení výrokových symbolů vyskytujících se ve φ (tzv. „tabulková metoda“):

Například pro formule φ a $\neg\varphi$, kde φ je formule $\neg(p \supset q) \equiv (p \wedge \neg q)$

p	q	$p \supset q$	$\neg(p \subset q)$	$p \wedge \neg q$	φ	$\neg\varphi$
0	0	1	0	0	1	0
0	1	1	0	0	1	0
1	0	0	1	1	1	0
1	1	1	0	0	1	0

Vidíme tedy, že φ je tautologie, $\neg\varphi$ je kontradikce.
Formule $\neg(p \supset q)$ a $p \wedge \neg q$ jsou splnitelné (nejsou to tedy kontradikce), nejsou to však tautologie.

Výroková logika – sémantika

Poznámka: Pokud se ve formuli vyskytuje n různých výrokových symbolů, musíme tabulkovou metodou ověřit celkem 2^n různých pravdivostních ohodnocení.

Výroková logika – tautologie

Všimněme si, že pokud je nějaká formule tautologie, pak i formule, která z ní vznikne nahrazením výrokových symbolů $p_1, p_2, \ldots, p_n$ formulemi $\varphi_1, \varphi_2, \ldots, \varphi_n$ je také tautologií.

Příklady některých důležitých tautologií:

Tautologie s jedním výrokovým symbolem:

$$\begin{aligned}
&\models p \equiv p \\
&\models p \vee \neg p && \text{– zákon vyloučení třetího} \\
&\models \neg(p \wedge \neg p) && \text{– zákon sporu} \\
&\models p \equiv \neg\neg p && \text{– zákon dvojí negace}
\end{aligned}$$

Příklad: Pokud například v zákoně vyloučení třetího nahradíme výrokový symbol p formulí $(p \wedge q) \supset r$, dostaneme tautologii:

$$((p \wedge q) \supset r) \vee \neg((p \wedge q) \supset r)$$

Výroková logika – tautologie

Algebraické zákony pro konjunkci, disjunkci a ekvivalenci:

$\models (p \wedge q) \equiv (q \wedge p)$ – komutativní zákon pro $\wedge$
$\models (p \vee q) \equiv (q \vee p)$ – komutativní zákon pro $\vee$
$\models (p \equiv q) \equiv (q \equiv p)$ – komutativní zákon pro $\equiv$

$\models ((p \wedge q) \wedge r) \equiv (p \wedge (q \wedge r))$ – asociativní zákon pro $\wedge$
$\models ((p \vee q) \vee r) \equiv (p \vee (q \vee r))$ – asociativní zákon pro $\vee$
$\models ((p \equiv q) \equiv r) \equiv (p \equiv (q \equiv r))$ – asociativní zákon pro $\equiv$

$\models ((p \vee q) \wedge r) \equiv ((p \wedge r) \vee (q \wedge r))$ – distributivní zákon
$\models ((p \wedge q) \vee r) \equiv ((p \vee r) \wedge (q \vee r))$ – distributivní zákon

Výroková logika – tautologie

Zákony pro implikaci:

$\models p \supset (q \supset p)$	– zákon simplifikace
$\models (p \wedge \neg p) \supset q$	– zákon Dunse Scota
$\models (p \supset q) \equiv (\neg q \supset \neg p)$	– zákon kontrapozice
$\models (p \supset (q \supset r)) \equiv ((p \wedge q) \supset r)$	– spojování předpokladů
$\models (p \supset (q \supset r)) \equiv (q \supset (p \supset r))$	– na pořadí předpokladů nezáleží
$\models (p \supset q) \supset ((q \supset r) \supset (p \supset r))$	– hypotetický sylogismus
$\models ((p \supset q) \wedge (q \supset r)) \supset (p \supset r)$	– tranzitivita implikace
$\models (p \supset (q \supset r)) \equiv ((p \supset q) \supset (p \supset r))$	– Fregův zákon
$\models (\neg p \supset p) \supset p$	– reductio ad absurdum
$\models ((p \supset q) \wedge (p \supset \neg q)) \supset \neg p$	– reductio ad absurdum

$\models (p \wedge q) \supset p$, $\models (p \wedge q) \supset q$

$\models p \supset (p \vee q)$, $\models q \supset (p \vee q)$

Výroková logika – tautologie

Zákony pro převody:

$$\models (p \equiv q) \equiv (p \supset q) \wedge (q \supset p)$$
$$\models (p \equiv q) \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$$
$$\models (p \equiv q) \equiv (\neg p \vee q) \wedge (\neg q \vee p)$$
$$\models (p \supset q) \equiv (\neg p \vee q)$$
$$\models \neg(p \supset q) \equiv (p \wedge \neg q) \quad \text{– negace implikace}$$
$$\models \neg(p \wedge q) \equiv (\neg p \vee \neg q) \quad \text{– De Morganův zákon}$$
$$\models \neg(p \vee q) \equiv (\neg p \wedge \neg q) \quad \text{– De Morganův zákon}$$

Poznámka: Tyto zákony jsou také návodem jak negovat.

Výroková logika – sémantika

Formule φ a ψ jsou **ekvivalentní**, jestliže pro každé pravdivostní ohodnocení ν platí $[\varphi]_\nu = [\psi]_\nu$.

Skutečnost, že φ a ψ jsou ekvivalentní budeme značit zápisem $\varphi \Leftrightarrow \psi$.

Není těžké si rozmyslet, že platí následující tvrzení.

Tvrzení

Pro libovolné formule φ a ψ platí, že $\varphi \Leftrightarrow \psi$ právě tehdy, když $\models \varphi \equiv \psi$.

Poznámka: Pro ověření, zda jsou formule ekvivalentní můžeme opět použít tabulkovou metodu.

Výroková logika – sémantika

Všimněme si, že relace $\Leftrightarrow$ je ekvivalence:

- Je **reflexivní**: Pro libovolnou formuli φ platí $\varphi \Leftrightarrow \varphi$.
- Je **symetrická**: Z $\varphi \Leftrightarrow \psi$ plyne $\psi \Leftrightarrow \varphi$.
- Je **tranzitivní**: Z $\varphi_1 \Leftrightarrow \varphi_2$ a $\varphi_2 \Leftrightarrow \varphi_3$ plyne $\varphi_1 \Leftrightarrow \varphi_3$.

Navíc platí následující:

Pokud formuli φ' dostaneme z formule φ tak, že ve φ nahradíme nějaký výskyt podformule ψ podformulí ψ' takovou, že $\psi \Leftrightarrow \psi'$, pak platí $\varphi \Leftrightarrow \varphi'$.

To nám umožňuje dokazovat ekvivalelci formulí pomocí **ekvivalentních úprav**, kdy postupně nahrazujeme různé podformule ekvivalentními podformulemi, až z jedné zadané formule dostaneme druhou zadanou formuli.

Výroková logika – logické vyplývání

Formule ψ **logicky vyplývá** z množiny formulí $\varphi_1, \varphi_2, \ldots, \varphi_n$, jestliže při každém ohodnocení, ve kterém platí všechny formule $\varphi_1, \varphi_2, \ldots, \varphi_n$ platí i formule ψ.

To, že formule ψ vyplývá z množiny formulí $\varphi_1, \varphi_2, \ldots, \varphi_n$ zapisujeme

$$\varphi_1, \varphi_2, \ldots, \varphi_n \models \psi$$

Poznámka: Platí, že $\varphi_1, \varphi_2, \ldots, \varphi_n \models \psi$ právě tehdy, když $\models \varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n \supset \psi$.

Predikátová logika 1. řádu – syntaxe

Formule predikátové logiky 1. řádu jsou tvořeny následujícími symboly:

- Proměnné: x, y, z, ...
- Funkční symboly: f, g, ...
- Predikátové symboly: P, Q, R, ...
- Logické spojky: $\neg$, $\wedge$, $\vee$, $\supset$, $\equiv$
- Kvantifikátory:
  - $\exists$ – existenční kvantifikátor
  - $\forall$ – universální (všeobecný) kvantifikátor
- Závorky: $($, $)$, $[$, $]$, ...

Pro každý funkční a predikátový symbol musí být specifikována jeho arita (počet argumentů).

Poznámka: Funkčním symbolům s aritou 0 se říká konstanty.
Budeme je označovat symboly a, b, c, ...

Predikátová logika 1. řádu – syntaxe

Množina všech **termů** je definována následovně:

1. Proměnná (tj. libovolný symbol z množiny x, y, z, ...) je term.
2. Jestliže $t_1, t_2, \ldots, t_n$ jsou termy a f je n-ární funkční symbol (tj. funkční symbol s aritou n), pak $f(t_1, t_2, \ldots, t_n)$ je term.
3. Neexistují žádné další termy.

Příklad: Řekněme, že f je binární funkční symbol a g je unární funkční symbol. Pak níže uvedené výrazy jsou termy:

$$x \qquad f(x, y) \qquad g(f(g(x), f(g(z), y)))$$

Poznámka: Pro konstanty (tj. 0-ární funkční symboly) většinou v zápise termu vynecháváme závorky, tj. například místo $c()$ obvykle píšeme c.

Predikátová logika 1. řádu – syntaxe

Atomické formule jsou definovány následovně:

1. Jestliže $t_1, t_2, \ldots, t_n$ jsou termy a P je n-ární predikátový symbol, pak $P(t_1, t_2, \ldots, t_n)$ je atomická formule.
2. Žádné další atomické formule neexistují.

Příklady atomických formulí, kde f je binární funkční symbol, g je unární funkční symbol, c je konstanta, P je binární predikátový symbol a Q je unární predikátový symbol:

$$P(x, y) \qquad Q(g(c)) \qquad P(f(x, g(y)), c)$$

Predikátová logika 1. řádu – syntaxe

Množina všech dobře utvořených **formulí** je definována následovně:

1. Atomické formule jsou formule.
2. Jestliže φ a ψ jsou formule, pak i $\neg\varphi$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \supset \psi)$ a $(\varphi \equiv \psi)$ jsou formule.
3. Jestliže φ je formule a x je proměnná, pak i $\exists x\varphi$ a $\forall x\varphi$ jsou formule.
4. Žádné další formule neexistují.

Příklady dobře utvořených formulí:

$$P(x,y) \qquad \forall x(Q(x) \vee P(y,x)) \qquad \forall z \exists x \forall y(P(x,y) \supset \neg Q(g(x)))$$

Poznámka: Používáme podobná pravidla o vypouštění závorek jako ve výrokové logice. Kvantifikátory mají vyšší prioritu než všechny logické spojky.

Predikátová logika 1. řádu – syntaxe

Výskyt proměnné x ve formuli φ se nazývá **vázaný**, jestliže se nachází v nějaké podformuli tvaru $\exists x\psi$ nebo $\forall x\psi$.

Výskyt proměnné x ve formuli φ, který není vázaný, se nazývá **volný**. Pokud formule φ obsahuje volný výskyt proměnné x, pak říkáme, že φ obsahuje volnou proměnnou x.

Poznámka: Všimněte si, že formule může současně obsahovat volné i vázané výskyty téže proměnné.

Příklad:

$$\exists x(R(y,z) \wedge \forall y(\neg P(y,x) \vee R(y,z)))$$

První výskyt proměnné y je volný, další dva jsou vázané.

Formule, která neobsahuje žádnou volnou proměnnou, se nazývá **uzavřená formule (sentence)**.

Predikátová logika 1. řádu – sémantika

Interpretace (**interpretační struktura**) $\mathcal{I}$ se skládá z:

- neprázdné množiny U nazývané **universum**,
- z funkce r přiřazující funkce a relace funkčním a predikátovým symbolům:
  - Jestliže f je n-ární funkční symbol, pak $r(f) = f^{\mathcal{I}}$, kde $f^{\mathcal{I}}$ je nějaká n-ární funkce na množině U, tj. funkce typu $f^{\mathcal{I}} : U^n \to U$.
  - Jestliže P je n-ární predikátový symbol, pak $r(P) = P^{\mathcal{I}}$, kde $P^{\mathcal{I}}$ je nějaká n-ární relace na množině U, tj. $P^{\mathcal{I}} \subseteq U^n$.

Poznámka: Všimněte si, že pro konstantu c je $c^{\mathcal{I}}$ nějaký prvek množiny U.

Predikátová logika 1. řádu – sémantika

Řekněme, že f a g jsou binární funkční symboly, h je unární funkční symbol, a je konstanta a P a Q jsou binární predikátové symboly.

Příklad: Interpretace $\mathcal{N}$, kde universem je množina přirozených čísel $\mathbb{N} = \{0, 1, 2, 3, \ldots\}$, a kde funkčním a predikátovým symbolům jsou přiřazeny následující funkce a relace:

- $f^{\mathcal{N}} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ je funkce taková, že $f^{\mathcal{N}}(x, y) = x + y$,
- $g^{\mathcal{N}} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ je funkce taková, že $g^{\mathcal{N}}(x, y) = x \cdot y$,
- $h^{\mathcal{N}} : \mathbb{N} \to \mathbb{N}$ je funkce taková, že $h^{\mathcal{N}}(x) = x + 1$.
- $a^{\mathcal{N}}$ je 0,
- $P^{\mathcal{N}} \subseteq \mathbb{N} \times \mathbb{N}$ je relace taková, že $P^{\mathcal{N}}(x, y)$ právě když $x = y$,
- $Q^{\mathcal{N}} \subseteq \mathbb{N} \times \mathbb{N}$ je relace taková, že $P^{\mathcal{N}}(x, y)$ právě když $x < y$.

Predikátová logika 1. řádu – sémantika

Příklad: Interpretace $\mathcal{A}$, kde universem je množina $A = \{a, b, c\}$, a kde:

- $f^{\mathcal{A}} : A \times A \to A$, $g^{\mathcal{A}} : A \times A \to A$ a $h^{\mathcal{A}} : A \to A$ jsou funkce dané následujícími tabulkami:

$f^{\mathcal{A}}$	a	b	c
a	c	a	b
b	b	b	a
c	c	a	c

$g^{\mathcal{A}}$	a	b	c
a	a	a	b
b	a	b	c
c	c	b	a

$h^{\mathcal{A}}$	
a	b
b	a
c	c

- $a^{\mathcal{A}}$ je prvek b,
- $P^{\mathcal{A}} \subseteq A \times A$ a $Q^{\mathcal{A}} \subseteq A \times A$ jsou relace

$$P^{\mathcal{A}} = \{(a, a), (a, c), (b, a), (c, b)\}$$
$$Q^{\mathcal{A}} = \{(a, b), (a, c), (b, b), (b, c), (c, b), (c, c)\}$$

Predikátová logika 1. řádu – sémantika

Předpokládejme nějakou interpretaci $\mathcal{I}$ s universem U.

Ohodnocení (valuace) proměnných je zobrazení e, které každé proměnné x přiřazuje hodnotu $e(x) \in U$.

Ohodnocení termů e^* indukované ohodnocením proměnných e je induktivně definováno takto:

- $e^*(x) = e(x)$
- $e^*(f(t_1, t_2, \ldots, t_n)) = f^{\mathcal{I}}(e^*(t_1), e^*(t_2), \ldots, e^*(t_n))$, kde $f^{\mathcal{I}}$ je funkce přiřazená v dané interpretaci funkčnímu symbolu f.

Poznámka: Hodnotou termu v interpretaci $\mathcal{I}$ pro valuaci e je tedy vždy nějaký prvek universa U.

Predikátová logika 1. řádu – sémantika

Zápisem $\mathcal{I} \models \varphi[e]$ budeme značit, že formule φ je **pravdivá** v interpretaci $\mathcal{I}$ pro ohodnocení e.

Pravdivost φ v interpretaci $\mathcal{I}$ pro ohodnocení e je definována následovně:

- Pokud P je n-ární predikátový symbol a $t_1, t_2, \ldots, t_n$ jsou termy, tak $\mathcal{I} \models P(t_1, t_2, \ldots, t_n)[e]$ platí právě tehdy, když $(e^*(t_1), e^*(t_2), \ldots, e^*(t_n)) \in P^{\mathcal{I}}$.
- $\mathcal{I} \models \neg\psi[e]$ platí právě tehdy, když neplatí $\mathcal{I} \models \psi[e]$.
- $\mathcal{I} \models \psi_1 \wedge \psi_2[e]$ platí právě tehdy, když platí $\mathcal{I} \models \psi_1[e]$ a $\mathcal{I} \models \psi_2[e]$.
- $\mathcal{I} \models \psi_1 \vee \psi_2[e]$ platí právě tehdy, když platí $\mathcal{I} \models \psi_1[e]$ nebo platí $\mathcal{I} \models \psi_2[e]$.
- $\mathcal{I} \models \psi_1 \supset \psi_2[e]$ platí právě tehdy, když neplatí $\mathcal{I} \models \psi_1[e]$ nebo platí $\mathcal{I} \models \psi_2[e]$.
- $\mathcal{I} \models \psi_1 \equiv \psi_2[e]$ platí právě tehdy, když platí $\mathcal{I} \models \psi_1[e]$ i $\mathcal{I} \models \psi_2[e]$ nebo neplatí $\mathcal{I} \models \psi_1[e]$ ani $\mathcal{I} \models \psi_2[e]$.

Predikátová logika 1. řádu – sémantika

- $\mathcal{I} \models \exists x \psi[e]$ platí právě tehdy, jestliže existuje prvek $i \in U$ takový, že $\mathcal{I} \models \psi[e(x \mapsto i)]$.
- $\mathcal{I} \models \forall x \psi[e]$ platí právě tehdy, jestliže pro každý prvek $i \in U$ platí $\mathcal{I} \models \psi[e(x \mapsto i)]$.

Poznámka: $e(x \mapsto i)$ je ohodnocení stejné jako e až na to, že přiřazuje proměnné x prvek i.

Příklad: Formule

$$\forall x \exists y (Q(x, y) \wedge P(y, g(x, x)))$$

ve dříve uvedené interpretaci $\mathcal{N}$ říká, že ke každému přirozenému číslu x existuje přirozené číslo y takové, že $x < y$ a $y = x \cdot x$.

Predikátová logika 1. řádu – sémantika

- Formule φ je **splnitelná v interpretaci** $\mathcal{I}$, jestliže existuje ohodnocení proměnných e takové, že $\mathcal{I} \models \varphi[e]$.
- Formule φ je **pravdivá v interpretaci** $\mathcal{I}$, značíme $\mathcal{I} \models \varphi$, jestliže každé ohodnocení proměnných e platí $\mathcal{I} \models \varphi[e]$.

Model formule φ je interpretace $\mathcal{I}$, ve které je φ pravdivá.

- Formule φ je **splnitelná**, jestliže existuje interpretace $\mathcal{I}$, ve které je splněna, tj. jestliže existuje interpretace $\mathcal{I}$ a valuace e taková, že $\mathcal{I} \models \varphi[e]$.
- Formule φ je **tautologie (logicky pravdivá)**, značíme $\models \varphi$, jestliže je pravdivá v každé interpretaci.
- Formule φ je **kontradikce**, jestliže nemá model, tedy neexistuje interpretace $\mathcal{I}$, která by formuli φ splňovala.

Predikátová logika 1. řádu – sémantika

Příklad: Zapište v jazyce PL1 následující výrok a najděte jeho model a také interpretaci, ve které není pravdivý:

Množina A je podmnožinou rozdílu množin B a C.

Predikátová logika 1. řádu – sémantika

Příklad: Zapište v jazyce PL1 následující výrok a najděte jeho model a také interpretaci, ve které není pravdivý:

Množina A je podmnožinou rozdílu množin B a C.

Řešení:

$$\forall x(A(x) \supset (B(x) \wedge \neg C(x)))$$

Predikátová logika 1. řádu – sémantika

Příklad: Zapište v jazyce PL1 následující výrok a najděte jeho model a také interpretaci, ve které není pravdivý:

Množina A je podmnožinou rozdílu množin B a C.

Řešení:

$$\forall x(A(x) \supset (B(x) \wedge \neg C(x)))$$

Model:

$U = \{a, b, c\}$
$A = \{a\}$
$B = \{a, b, c\}$
$C = \{c\}$

Predikátová logika 1. řádu – sémantika

Příklad: Zapište v jazyce PL1 následující výrok a najděte jeho model a také interpretaci, ve které není pravdivý:

Množina A je podmnožinou rozdílu množin B a C.

Řešení:

$$\forall x(A(x) \supset (B(x) \wedge \neg C(x)))$$

Model:

$U = \{a, b, c\}$
$A = \{a\}$
$B = \{a, b, c\}$
$C = \{c\}$

Interpretace, kde není pravdivý:

$U = \{a, b, c\}$
$A = \{c\}$
$B = \{a, b, c\}$
$C = \{c\}$

Predikátová logika 1. řádu – logické vyplývání

Je zřejmé, že φ je tautologie, právě když $\neg\varphi$ je kontradikce.

Model množiny formulí $\{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ je taková interpretace $\mathcal{I}$, ve které jsou pravdivé všechny formule $\varphi_1, \varphi_2, \ldots, \varphi_n$.

Formule ψ **logicky vyplývá** z formulí $\varphi_1, \varphi_2, \ldots, \varphi_n$, značíme

$$\varphi_1, \varphi_2, \ldots, \varphi_n \models \psi$$

jestliže ψ je pravdivá v každém modelu množiny formulí $\varphi_1, \varphi_2, \ldots, \varphi_n$.

Predikátová logika 1. řádu – ekvivalence formulí

Formule φ a ψ jsou **(sémanticky) ekvivalentní**, jestliže pro všechny interpretace $\mathcal{I}$ a valuace e mají stejná pravdivostní ohodnocení. Skutečnost, že φ a ψ jsou ekvivalentní zapisujeme $\varphi \Leftrightarrow \psi$.

Poznámka: Všimněte si, že $\varphi \Leftrightarrow \psi$ právě tehdy, když $\varphi \models \psi$ a $\psi \models \varphi$.

Poznámka: Také platí, že $\varphi \Leftrightarrow \psi$ právě tehdy, když $\models \varphi \equiv \psi$.

Podobně jako ve výrokové logice platí, že pokud vezmeme libovolnou tautologii výrokové logiky a nahradíme v ní výrokové symboly $p_1, p_2, \ldots, p_n$ formulemi $\varphi_1, \varphi_2, \ldots, \varphi_n$ PL1, dostaneme tautologii PL1.

Podobně jako ve výrokové logice také nahrazením libovolné podformule ekvivalentní podformulí dostaneme formuli ekvivalentní s původní formulí.

Predikátová logika 1. řádu – ekvivalence formulí

Příklad: Pomocí ekvivalentních úprav dokážeme, že následující formule jsou ekvivalentní

$$\forall x \exists y \left[P\left(y, f(y)\right) \supset Q(x) \right]$$
$$\forall x \left[\forall y P\left(y, f(y)\right) \supset Q(x) \right]$$

Predikátová logika 1. řádu – ekvivalence formulí

Příklad: Pomocí ekvivalentních úprav dokážeme, že následující formule jsou ekvivalentní

$$\forall x \exists y \left[P\left(y, f(y)\right) \supset Q(x)\right]$$
$$\forall x \left[\forall y P\left(y, f(y)\right) \supset Q(x)\right]$$

Řešení:

$$\begin{aligned}
&\forall x \exists y \left(P\left(y, f(y)\right) \supset Q(x)\right) \\
\Leftrightarrow\ &\forall x \exists y (\neg P(y, f(y)) \vee Q(x)) \\
\Leftrightarrow\ &\forall x (\exists y (\neg P(y, f(y))) \vee Q(x)) \\
\Leftrightarrow\ &\forall x (\neg \forall y P(y, f(y)) \vee Q(x)) \\
\Leftrightarrow\ &\forall x (\forall y P(y, f(y)) \supset Q(x))
\end{aligned}$$

Predikátová logika 1. řádu – logické vyplývání

Všechny následující úsudky jsou z hlediska logiky korektní (v podstatě se jedná o totožné úsudky):

Všichni žáci jsou lidé. Někteří žáci jsou pilní. --- Někteří lidé jsou pilní.	$\forall x(Z(x) \supset R(x))$ $\exists x(Z(x) \wedge M(x))$ --- $\exists x(R(x) \wedge M(x))$

Všichni žáci jsou ryby. Někteří žáci jsou mloci. --- Některé ryby jsou mloci.	$Z(x)$ – x je žák $R(x)$ – x je člověk/ryba $M(x)$ – x je pilný/mlok

Pro zdůvodnění toho, že jsou tyto úsudky korektní (že závěr logicky vyplývá z předpokladů) můžeme použít tzv. Vénových diagramů

Predikátová logika 1. řádu – logické vyplývání

Příklad: Zformalizujte pomocí PL1 a rozhodněte, zda je následující úsudek platný (své tvrzení podložte důkazem):

Nikdo s červeným nosem nemůže být premiér.
Všichni Valaši mají červený nos.

Žádný Valach nemůže být premiérem.

Predikátová logika 1. řádu – logické vyplývání

Příklad: Zformalizujte pomocí PL1 a rozhodněte, zda je následující úsudek platný (své tvrzení podložte důkazem):

Nikdo s červeným nosem nemůže být premiér.
Všichni Valaši mají červený nos.

Žádný Valach nemůže být premiérem.

Formalizace:

$$\frac{\forall x(C(x) \supset \neg P(x)) \quad \forall x(V(x) \supset C(x))}{\forall x(V(x) \supset \neg P(x))}$$

$C(x)$ – x má červený nos
$P(x)$ – x může být premiérem
$V(x)$ – x je Valach

Poznámka:
Věta „Nikdo s červeným nosem nemůže být premiér.“ může být formalizována např. i takto: $\neg\exists x(C(x) \wedge P(x))$

Predikátová logika 1. řádu – logické vyplývání

Příklad: Zformalizujte pomocí PL1 a rozhodněte, zda je následující úsudek platný (své tvrzení podložte důkazem):

Nikdo s červeným nosem nemůže být premiér.
Všichni Valaši mají červený nos.

Existuje Valach, který nemůže být premiérem.

Predikátová logika 1. řádu – logické vyplývání

Příklad: Zformalizujte pomocí PL1 a rozhodněte, zda je následující úsudek platný (své tvrzení podložte důkazem):

Nikdo s červeným nosem nemůže být premiér.
Všichni Valaši mají červený nos.

Existuje Valach, který nemůže být premiérem.

Řešení:

$$\frac{\forall x(C(x) \supset \neg P(x)) \quad \forall x(V(x) \supset C(x))}{\exists x(V(x) \wedge \neg P(x))}$$

$C(x)$ – x má červený nos
$P(x)$ – x může být premiérem
$V(x)$ – x je Valach

Např. v interpretaci, kde universum je $U = \{a\}$, a jednotlivým predikátům jsou přiřazeny (unární) relace $C = \emptyset$, $P = \emptyset$, $V = \emptyset$, platí obě premisy, ale neplatí závěr.

Rezoluční metoda

Rezoluční metoda je příkladem syntaktické metody dokazování.

Poznámka: Rezoluční metoda se využívá např. v programovacím jazyce Prolog (logické programování).

My se zaměříme na rezoluční metodu pouze ve výrokové logice.

Rezoluční metoda je založena na dvou jednoduchých principech:

1. Formule φ je tautologie právě když formule $\neg\varphi$ je kontradikce (a naopak).
2. Rezoluční pravidlo odvozování:

$$(p \vee \varphi) \wedge (\neg p \vee \psi) \models \varphi \vee \psi$$

Rezoluční metoda

Rezoluční metoda pracuje s formulemi v tzv. **konjunktivní normální formě (KNF)**.

- **Literál** je výrokový symbol nebo jeho negace, např. p nebo $\neg p$.
- **Klauzule** je disjunkce libovolného počtu literálů, např. $p \vee \neg q \vee \neg r$.
- Formule v **konjunktivní normální formě (KNF)** je konjunkce libovolného počtu klauzulí, např.

$$(\neg p \vee r) \wedge (p \vee q \vee \neg r) \wedge (\neg q) \wedge (r \vee \neg s)$$

Rezoluční metoda

Každou formuli výrokové logiky můžeme pomocí ekvivalentních úprav převést do KNF, například takto:

- Zbavíme se logické spojky $\equiv$ (např. využitím $\varphi \equiv \psi \Leftrightarrow (\varphi \supset \psi) \wedge (\psi \supset \varphi)$).
- Zbavíme se logické spojky $\supset$ (např. využitím $\varphi \supset \psi \Leftrightarrow \neg\varphi \vee \psi$).
- Zbavíme se všech negací z vyjímkou těch, které jsou aplikovány přímo na výrokový symbol (s použitím De Morganových zákonů a zákona dvojité negace).
- Formuli upravíme do požadovaného tvaru použitím distributivních zákonů pro $\wedge$ a $\vee$.

Rezoluční metoda

Problém dokázat, že platí $\varphi_1, \varphi_2, \ldots, \varphi_n \models \psi$ převedeme na problém dokázat, že $\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n \wedge \neg\psi$ je kontradikce.

To, že je to korektní postup je zaručeno následující sérií vzájemně ekvivalentních tvrzení:

- $\varphi_1, \varphi_2, \ldots, \varphi_n \models \psi$ je tautologie
- $\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n \supset \psi$ je tautologie
- $\neg\varphi_1 \vee \neg\varphi_2 \vee \cdots \vee \neg\varphi_n \vee \psi$ je tautologie
- $\neg(\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n \wedge \psi)$ je tautologie
- $\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n \wedge \neg\psi$ je kontradikce

Formuli převedeme do KNF a zapíšeme si její jednotlivé klauzule.

Rezoluční metoda

Při libovolném ohodnocení, při kterém je celá formule pravdivá, musí být pravdivé všechny její klauzule.

Pokud se ve formuli nachází nějaké dvě klauzule tvaru

$$p \vee L_1 \vee L_2 \vee \cdots L_m \qquad \neg p \vee L'_1 \vee L'_2 \vee \cdots L'_n$$

tak při libovolném ohodnocení, při kterém je celá formule pravdivá, musí být pravdivá (kvůli rezolučnímu pravidlu) i následující klauzule

$$L_1 \vee L_2 \vee \cdots L_m \vee L'_1 \vee L'_2 \vee \cdots L'_n$$

kterou tím pádem můžeme k formuli přidat aniž bychom tím ovlivnili splnitelnost celé formule.

Poznámka: Vzhledem k asociativitě a komutativitě disjunkce nezáleží na pořadí literálů v klauzuli.

Rezoluční metoda

Speciálně v případě kdy klauzule jsou tvaru p a $\neg p$ je výsledkem použití rezolučního pravidla, tzv. **prázdná klauzule**, kterou budeme označovat $\square$, která nemůže být při žádné ohodnocení pravdivá, a která představuje spor.

Celý postup tedy vypadá tak, že uplatňujeme rezoluční pravidlo tak dlouho, dokud nějaké nové klauzule přibývají nebo dokud neodvodíme spor.

- Pokud odvodíme spor, byla formule kontradikcí a původní úsudek byl logicky platný.
- Pokud spor neodvodíme a nemůžeme pomocí rezolučního pravidla už žádnou další klauzuli přidat, formule nebyla kontradikcí a původní úsudek nebyl logicky platný.

Poznámka: Klauzule, které jsou již ve formuli obsaženy znovu nepřidáváme. V každé klauzuli také vypouštíme opakující se literály (necháváme každý literál jedenkrát).

Rezoluční metoda

Chceme ověřit platnost následujícího úsudku:

Není pravda, že Jana je ve škole a Petr není doma.
Jana není ve škole nebo je všední den nebo prší.
Jestliže je všední den, pak Petr není doma.

Jestliže je Jana ve škole, pak prší.

Jednotlivá tvrzení nejprve zformalizujeme pomocí výrokové logiky:

$\neg(J \wedge \neg P)$
$\neg J \vee D \vee R$
$D \supset \neg P$

$J \supset R$

J – Jana je škole
P – Petr je doma
D – je všední den
R – prší

Rezoluční metoda

$$\frac{\begin{array}{l}\neg(J \wedge \neg P)\\ \neg J \vee D \vee R\\ D \supset \neg P\end{array}}{J \supset R}$$

Jednotlivé předpoklady převedeme do KNF:

- $\neg(J \wedge \neg P) \Leftrightarrow \neg J \vee P$
- $J \vee D \vee R$
- $D \supset \neg P \Leftrightarrow \neg D \vee \neg P$

Závěr znegujeme a převedeme do KNF:

- $\neg(J \supset R) \Leftrightarrow J \wedge \neg R$

Rezoluční metoda

Sepíšeme si jednotlivé klauzule:

1. $\neg J \vee P$ – předpoklad 1
2. $\neg J \vee D \vee R$ – předpoklad 2
3. $\neg D \vee \neg P$ – přespoklad 3
4. J – 1. klauzule znegovaného závěru
5. $\neg R$ – 2. klauzule znegovaného závěru

Rezoluční metoda

Sepíšeme si jednotlivé klauzule:

1.	$\neg J \vee P$	– předpoklad 1
2.	$\neg J \vee D \vee R$	– předpoklad 2
3.	$\neg D \vee \neg P$	– přespoklad 3
4.	J	– 1. klauzule znegovaného závěru
5.	$\neg R$	– 2. klauzule znegovaného závěru
6.	P	– rezoluce: 1,4

Rezoluční metoda

Sepíšeme si jednotlivé klauzule:

1.	$\neg J \vee P$	– předpoklad 1
2.	$\neg J \vee D \vee R$	– předpoklad 2
3.	$\neg D \vee \neg P$	– přespoklad 3
4.	J	– 1. klauzule znegovaného závěru
5.	$\neg R$	– 2. klauzule znegovaného závěru
6.	P	– rezoluce: 1,4
7.	$D \vee R$	– rezoluce: 2,4

Rezoluční metoda

Sepíšeme si jednotlivé klauzule:

1.	$\neg J \vee P$	– předpoklad 1
2.	$\neg J \vee D \vee R$	– předpoklad 2
3.	$\neg D \vee \neg P$	– přespoklad 3
4.	J	– 1. klauzule znegovaného závěru
5.	$\neg R$	– 2. klauzule znegovaného závěru
6.	P	– rezoluce: 1,4
7.	$D \vee R$	– rezoluce: 2,4
8.	$\neg D$	– rezoluce: 3,6

Rezoluční metoda

Sepíšeme si jednotlivé klauzule:

1.	$\neg J \vee P$	– předpoklad 1
2.	$\neg J \vee D \vee R$	– předpoklad 2
3.	$\neg D \vee \neg P$	– přespoklad 3
4.	J	– 1. klauzule znegovaného závěru
5.	$\neg R$	– 2. klauzule znegovaného závěru
6.	P	– rezoluce: 1,4
7.	$D \vee R$	– rezoluce: 2,4
8.	$\neg D$	– rezoluce: 3,6
9.	R	– rezoluce: 7,8

Rezoluční metoda

Sepíšeme si jednotlivé klauzule:

1.	$\neg J \vee P$	– předpoklad 1
2.	$\neg J \vee D \vee R$	– předpoklad 2
3.	$\neg D \vee \neg P$	– přespoklad 3
4.	J	– 1. klauzule znegovaného závěru
5.	$\neg R$	– 2. klauzule znegovaného závěru
6.	P	– rezoluce: 1,4
7.	$D \vee R$	– rezoluce: 2,4
8.	$\neg D$	– rezoluce: 3,6
9.	R	– rezoluce: 7,8
10.	$\square$	– rezoluce: 5,9