Základy logiky

26. 9. 2023

Logika je naukou, která se zabývá studiem lidského uvažování.

Mezi základní úlohy logiky patří nalézání *metod správného usuzování*, tedy postupů, které dovolují přecházet od poznatků, jejichž pravdivost byla ověřena, k poznatkům novým, vyplývajícím a také pravdivým.

Dva způsoby poznávání skutečnosti:

1. přímý (empirický)
2. nepřímý (teoretický)

Metody teoretického poznávání – *vyplývání* – zkoumá logika. Jsou založeny na odvozování nových poznatků z poznatků již získaných.

Zjištěné poznatky jsou zaznamenávány vhodným jazykem, u nějž bývá stejně důležitá syntax i sémantika. Jazyky logických počtů (*kalkulů*) umožňují přesně a úsporně vyjadřovat poznatky i formalizovat usuzování.

Ze symbolů abecedy jazyka vytváříme podle syntaktických pravidel *slova*. Správně vytvořená slova nazýváme *formule*. Mezi symboly jazyka patří úzus označování proměnných (je zcela speciální) a dále symboly pro označení logických funkcí.

Nové poznatky lze odvodit dvěma způsoby:

I. dedukcí

II. indukcí

Dedukce je proces usuzování, v němž se od předpokladů (premis) dochází k závěru z těchto premis vyplývajícího. Usuzování je vždy jisté (= prokázaná pravdivost), jde o základní postup při dokazování.

Důkazem je konečná posloupnost formulí vedoucí k závěru.

Způsoby dokazování:

1. Formule tvořící výchozí soubor jako pravdivé definujeme => *axiomy*
2. Pravdivost formulí dokážeme pomocí pravidel správného usuzování z tzv. *premis* (předpokladů)

Při použití souboru axiomů musí tento soubor být úplný a beze-sporný. Formule, které takto dokazujeme, nazýváme *teorémy* (*věty*).

Indukce spočívá v odvozování obecného poznatku z řady poznatků speciálních. Induktivní metody jsou spojovány s empirickým poznáváním.

Úplná indukce × neúplná indukce

Př.: Mějme tvrzení: Všichni sourozenci pana XY mají krevní skupinu A_2.

Důkaz: Odebereme panu XY i (všem) jeho sourozencům krevní vzorek, provedeme rozbor a jsou-li všechny vzorky skupiny A_2, pak je tvrzení dokázáno.

Výroková logika

Nejjednodušší logikou je **výroková logika**.

Tvrzení, o kterých má smysl rozhodovat, zda jsou pravdivá, nazýváme *výroky*.

Výrok, který nelze dále rozložit na výroky jednodušší, je tzv. *elementární výrok*.

Př.: *„Tři plus čtyři“* – není výrok, nemá smysl určovat jeho pravdivost
„Tři plus čtyři je sedm“ – elementární výrok (nelze již rozložit)
„Tři plus čtyři je sedm a *jedna plus pět je šest“* – složený výrok

Výroková logika se nezabývá vnitřní strukturou elementárních výroků, zajímá se pouze o to, zda výroky nabývají jednu ze dvou možných *pravdivostních hodnot* – **pravda**, **nepravda** (true, false nebo 1 a 0).

Elementární výroky lze proto nahradit libovolně zvolenými symboly => *výrokovými proměnnými*. Z těchto symbolů můžeme dále tzv. *logickými spojkami* a v případě nutnosti též závorkami vytvářet *složené výroky*.

Logickými spojkami ($\neg$, $\wedge$, $\vee$, $\rightarrow$, $\Leftrightarrow$) budeme zapisovat logické funkce **negace**, **logického součinu** (*konjunkce*), **logického součtu** (*disjunkce*), **podmíněného soudu** (*implikace*) a **logické rovnosti** (*ekvivalence*). Toto jsou základní funkce, jimiž lze popsat jakýkoli výrok.

Jazyk výrokové logiky

Zavedením výrokových proměnných, logických spojek a závorek jsme vymezili symboly používané *jazykem výrokové logiky*. Každá posloupnost těchto symbolů však nepatří do jazyka výrokové logiky (např. $A \wedge \rightarrow B$ atd.). Je třeba dodržovat syntax jazyka.

Korektně vytvořené výrazy jazyka výrokové logiky budeme nazývat *výrokové formule*.

Jejich syntax vymezuje následující definice:

Označme $V = \{ \neg, \wedge, \vee, \rightarrow, \Leftrightarrow, (,), A, B, C, \ldots\}$ abecedu jazyka výrokové logiky, kde $A, B, C, \ldots$ označují výrokové proměnné.

A platí:

1. Každá výroková proměnná je výrokovou formulí.
2. Jestliže $\mathcal{A}$ a $\mathcal{B}$ jsou výrokové formule, pak také $(\neg\mathcal{A})$, $(\mathcal{A} \wedge \mathcal{B})$, $(\mathcal{A} \vee \mathcal{B})$, $(\mathcal{A} \rightarrow \mathcal{B})$ a $(\mathcal{A} \Leftrightarrow \mathcal{B})$ jsou výrokové formule.
3. Žádné jiné výrokové formule než podle bodů 1 a 2 neexistují.

Jazykem výrokové logiky nad abecedou V potom nazýváme množinu všech výrokových formulí správně vytvořených ze symbolů abecedy V .

Existují výrokové formule, které jsou *identicky pravdivé*, tj. pravdivé pro jakékoli pravdivostní hodnoty přiřazené proměnným vyskytujícím se ve výrokové formuli. Takovéto formule nazýváme **výrokové tautologie**.

Příkladem výrokových tautologií jsou výrokové formule

$$((\neg(\mathcal{A} \wedge \mathcal{B})) \;\Leftrightarrow\; ((\neg\mathcal{A} \vee \neg\mathcal{B}))),$$

$$((\neg(\mathcal{A} \vee \mathcal{B})) \;\Leftrightarrow\; ((\neg\mathcal{A} \wedge \neg\mathcal{B}))),$$

které jsou známy jako de Morganovy zákony pro úpravu výrokových formulí.

Formule, která je identicky nepravdivá, se nazývá **kontradikce**. Dále platí, že je-li formule $\mathcal{A}$ kontradikce, potom $\neg\mathcal{A}$ je tautologie.

Jestliže formule $\mathcal{A} \Leftrightarrow \mathcal{B}$ je výrokovou tautologií, říkáme, že formule $\mathcal{A}$ a $\mathcal{B}$ jsou *logicky ekvivalentní*.

Libovolnou výrokovou formuli můžeme posoudit (vyhodnotit) *pravdivostní tabulkou*. To je vlastně metoda úplné indukce.

Př.: Vyhodnocení složeného výroku pravdivostní tabulkou – budiž dán složený výrok: $((A \wedge (\neg B)) \rightarrow C) \rightarrow (A \vee C)$

A	B	C	$A \wedge (\neg B)$	$(A \wedge (\neg B)) \rightarrow C$	$A \vee C$	$((A \wedge (\neg B)) \rightarrow C) \rightarrow (A \vee C)$
0	0	0	0	1	0	0
0	0	1	0	1	1	1
0	1	0	0	1	0	0
0	1	1	0	1	1	1
1	0	0	1	0	1	1
1	0	1	1	1	1	1
1	1	0	0	1	1	1
1	1	1	0	1	1	1

Řekneme, že výroková formule $\mathcal{B}$ vyplývá z formule $\mathcal{A}$, když formule $\mathcal{B}$ je pravdivá vždy, když je pravdivá formule $\mathcal{A}$.

Jestliže formule $\mathcal{B}$ vyplývá z formule $\mathcal{A}$ (zapíšeme $\mathcal{A} \models \mathcal{B}$), pak formule $\mathcal{A} \rightarrow \mathcal{B}$ je výrokovou tautologií.

Formuli $\mathcal{A}$ nazýváme *antecendent* nebo *premisa* a formuli $\mathcal{B}$ *konsekvent* nebo také *konkluse (závěr)*.

Z definice logického vyplývání a z vlastností implikace můžeme odvodit dvě formulace věty významné pro dokazování:

1. Formule $\mathcal{B}$ logicky vyplývá z formulí $\mathcal{A}_1$, $\mathcal{A}_2$, …, $\mathcal{A}_k$ tehdy a jen tehdy, když formule $(\mathcal{A}_1 \wedge \mathcal{A}_2 \wedge \ldots \wedge \mathcal{A}_k) \rightarrow \mathcal{B}$ je výrokovou tautologií.
2. Formule $\mathcal{B}$ logicky vyplývá z formulí $\mathcal{A}_1$, $\mathcal{A}_2$, …, $\mathcal{A}_k$ tehdy a jen tehdy, když formule $(\mathcal{A}_1 \wedge \mathcal{A}_2 \wedge \ldots \wedge \mathcal{A}_k \wedge \neg\mathcal{B})$ je výrokovou kontradikcí.

Z logického vyplývání lze také odvodit další významné pravidlo správného usuzování, a to *pravidlo odloučení* neboli pravidlo **modus ponens**, které říká, že jsou-li formule $\mathcal{A}$ a $\mathcal{A} \rightarrow \mathcal{B}$ výrokovými tautologiemi, pak i formule $\mathcal{B}$ je výrokovou tautologií.

Pravidlo modus ponens nám dovoluje odvozovat dedukcí. Zvolíme některé výrokové formule za axiómy jazyka výrokové logiky. Pokud vybereme jako axiómy výrokové tautologie, umíme pravidlem modus ponens odvozovat další výrokové tautologie.

Jedním z takových možných systémů axiomů je následující soubor formulí:

$$(\mathcal{A} \rightarrow (\mathcal{B} \rightarrow \mathcal{A})) \tag{6.1}$$

$$((\mathcal{A} \rightarrow (\mathcal{B} \rightarrow \mathcal{C})) \rightarrow ((\mathcal{A} \rightarrow \mathcal{B}) \rightarrow (\mathcal{A} \rightarrow \mathcal{C}))) \tag{6.2}$$

$$((\neg\mathcal{B} \rightarrow \neg\mathcal{A}) \rightarrow ((\neg\mathcal{B} \rightarrow \mathcal{A}) \rightarrow \mathcal{B})) \tag{6.3}$$

Tyto axiomy jsou tautologiemi a obsahují pouze dvě základní logické funkce – *negaci a implikaci*. Zbývající tři lze pomocí nich definovat takto:

$\mathcal{A} \wedge \mathcal{B}$ jako $(\neg(\mathcal{A} \rightarrow \neg\mathcal{B}))$

$\mathcal{A} \vee \mathcal{B}$ jako $((\neg\mathcal{A}) \rightarrow \mathcal{B})$

$\mathcal{A} \Leftrightarrow \mathcal{B}$ jako $((\mathcal{A} \rightarrow \mathcal{B}) \wedge (\mathcal{B} \rightarrow \mathcal{A}))$

Formule získané dedukcí z axiomů nazveme *formálně dokazatelné*.

Každá formálně dokazatelná formule jazyka výrokové logiky je výrokovou tautologií, protože axiomy (6.1) až (6.3) jsou výrokové tautologie a pravidlo modus ponens vede od tautologií opět k tautologiím.

Kromě toho lze dokázat, že uvedený soubor axiomů (6.1) až (6.3) je *úplný* v širokém smyslu či podle Gödela, tj. každá formule, která je tautologií, je z něj formálně dokazatelná.

Poznámka: Jsou i jiné možnosti, jak zvolit axiomy výrokové logiky. Obecně lze pak jeden systém axiomů odvodit z druhého, protože všechny axiomy jsou tautologie a každá tautologie je formálně dokazatelná z nějakého axiomu.

Formule, které zachycují další poznatky a byly získány odvozováním, nazýváme **(logicky) pravdivé formule**.

V odvození pravidla modus ponens jsme vycházeli z faktu, že $\mathcal{A}$ a $\mathcal{A} \to \mathcal{B}$ jsou tautologie; podobnou úvahou se lze přesvědčit, že platí i slabší tvrzení: jestliže formule $\mathcal{A}$ a $\mathcal{A} \to \mathcal{B}$ jsou pravdivé pro nějaké přiřazení pravdivostních hodnot výrokovým proměnným, které se v nich vyskytují, je pro stejné přiřazení i formule $\mathcal{B}$ pravdivá.

Tím získáme pravidlo modus ponens platné pro logicky pravdivé formule i pro výrokové tautologie.

Pravidla pro úpravy logických výrazů
(přehled ekvivalentních formulí)

1. $\mathcal{F} \Leftrightarrow \mathcal{G} \equiv (\mathcal{F} \rightarrow \mathcal{G}) \wedge (\mathcal{G} \rightarrow \mathcal{F})$
2. $\mathcal{F} \rightarrow \mathcal{G} \equiv \neg \mathcal{F} \vee \mathcal{G}$
3. $\mathcal{F} \vee \mathcal{G} \equiv \mathcal{G} \vee \mathcal{F}$

   $\mathcal{F} \wedge \mathcal{G} \equiv \mathcal{G} \wedge \mathcal{F}$
4. $(\mathcal{F} \vee \mathcal{G}) \vee \mathcal{H} \equiv \mathcal{F} \vee (\mathcal{G} \vee \mathcal{H})$

   $(\mathcal{F} \wedge \mathcal{G}) \wedge \mathcal{H} \equiv \mathcal{F} \wedge (\mathcal{G} \wedge \mathcal{H})$
5. $\mathcal{F} \vee (\mathcal{G} \wedge \mathcal{H}) \equiv (\mathcal{F} \vee \mathcal{G}) \wedge (\mathcal{F} \vee \mathcal{H})$

   $\mathcal{F} \wedge (\mathcal{G} \vee \mathcal{H}) \equiv (\mathcal{F} \wedge \mathcal{G}) \vee (\mathcal{F} \wedge \mathcal{H})$

Pravidla pro úpravy logických výrazů – pokračování

(přehled ekvivalentních formulí)

6. $\mathcal{F} \lor \square \equiv \mathcal{F}$

   $\mathcal{F} \land \blacksquare \equiv \mathcal{F}$

7. $\mathcal{F} \lor \blacksquare \equiv \blacksquare$

   $\mathcal{F} \land \square \equiv \square$

8. $\mathcal{F} \lor \neg \mathcal{F} \equiv \blacksquare$

   $\mathcal{F} \land \neg \mathcal{F} \equiv \square$

9. $\neg(\neg \mathcal{F}) \equiv \mathcal{F}$

10. $\neg(\mathcal{F} \lor \mathcal{G}) \equiv \neg \mathcal{F} \land \neg \mathcal{G}$

    $\neg(\mathcal{F} \land \mathcal{G}) \equiv \neg \mathcal{F} \lor \neg \mathcal{G}$

Teorie a modely jazyka výrokové logiky

Zvolíme nějaký soubor axiomů výrokové logiky, např. axiomy (6.1) až (6.3). Přidáme-li k těmto axiomům některé další pravdivé formule (*mimologické axiomy*), můžeme dedukcí odvozovat další formule (*poznatky*). Ty budou pravdivé ve všech množinách, ve kterých jsou pravdivé dané mimologické axiomy.

Označme *T* nějakou množinu formulí jazyka výrokové logiky; tyto formule zahrneme do jazyka jako mimologické axiomy. Množinu *T* pak budeme nazývat **teorií jazyka výrokové logiky**.

Důkazem formule $\mathcal{A}$ z teorie T je konečná posloupnost formulí taková, že jejím posledním členem je formule $\mathcal{A}$ a každá z předchozích formulí je buď axiom výrokové logiky, patří do T nebo se získá z předchozích formulí odvozením (aplikací pravidla *modus ponens*).

Potom řekneme, že formule $\mathcal{A}$ je *formálně dokazatelná* z teorie T, resp. $\mathcal{A}$ je *teorém* v T, a zapíšeme ji $T \vdash \mathcal{A}$. Pokud $T = \emptyset$, píšeme $\vdash \mathcal{A}$ a formule $\mathcal{A}$ je *formálně dokazatelnou formulí jazyka výrokové logiky* (a výrokovou tautologií).

Teorie T je *sporná*, jestliže lze najít nějakou formuli $\mathcal{A}$ takovou, že $T \vdash \mathcal{A}$ a $T \vdash \neg\mathcal{A}$. V opačném případě se teorie nazývá *bezesporná*.

Důležitou vlastnost odvozování popisuje tzv. **věta o dedukci**:

Jestliže $T \cup (\mathcal{A}) \vdash \mathcal{B}$, pak $T \vdash (\mathcal{A} \rightarrow \mathcal{B})$, kde $\mathcal{A}$ a $\mathcal{B}$ jsou formule výrokové logiky.

Důkaz věty o dedukci vychází z dané soustavy axiomů.

Větu ilustrujme následujícím příkladem:

Př.: *Odvoditelnost libovolné formule $\mathcal{B}$ ze sporné teorie $T = \{ \mathcal{A}, \neg\mathcal{A} \}$*

1.	$\mathcal{A}$	MA1
2.	$\neg\mathcal{A}$	MA2
3.	$\mathcal{A} \rightarrow (\neg\mathcal{B} \rightarrow \mathcal{A})$	axiom (6.1) – $\neg\mathcal{B}$ za $\mathcal{B}$
4.	$\neg\mathcal{A} \rightarrow (\neg\mathcal{B} \rightarrow \neg\mathcal{A})$	axiom (6.1) – $\neg\mathcal{A}$ za $\mathcal{A}$
5.	$\neg\mathcal{B} \rightarrow \mathcal{A}$	MP 1, 3
6.	$\neg\mathcal{B} \rightarrow \neg\mathcal{A}$	MP 2, 4
7.	$(\neg\mathcal{B} \rightarrow \neg\mathcal{A}) \rightarrow ((\neg\mathcal{B} \rightarrow \mathcal{A}) \rightarrow \mathcal{B})$	axiom (6.3)
8.	$(\neg\mathcal{B} \rightarrow \mathcal{A}) \rightarrow \mathcal{B}$	MP 6, 7
9.	$\mathcal{B}$	MP 5, 8

Závěr: O formuli $\mathcal{B}$ nebyly učiněny žádné předpoklady, lze tedy ze sporných axiomů $\mathcal{A}, \neg\mathcal{A}$ odvodit libovolnou formuli $\mathcal{B}$.

Zkoumáme-li sémantiku jazyka výrokové logiky, můžeme abstrahovat od konkrétních poznatků a nahradit je jejich *pravdivostními hodnotami*. Přiřazení pravdivostních hodnot všem symbolům abecedy jazyka výrokové logiky nazveme **modelem tohoto jazyka**.

Stejný model potom zastupuje všechny soubory elementárních výroků, které přiřazují jednotlivým výrokovým proměnným tutéž pravdivostní hodnotu.

Označme M model jazyka výrokové logiky a $\mathcal{A}$ formuli tohoto jazyka. Pokud model M vede k vyhodnocení formule s pravdivostní hodnotou *true*, řekneme, že formule $\mathcal{A}$ je pravdivá v modelu M a toto zapíšeme $M \models \mathcal{A}$.

Formule $\mathcal{A}$, která je pravdivá ve všech možných modelech, je *výrokovou tautologií*, formule $\mathcal{A}$, která není pravdivá ve všech modelech, je *výrokovou kontradikcí*.

Označme T teorii jazyka výrokové logiky a M model tohoto jazyka. Když $M \models \mathcal{A}$ pro všechny formule $\mathcal{A} \in T$, potom řekneme, že M je modelem teorie T. Jestliže v každém modelu M teorie T je formule $\mathcal{A}$ pravdivá, potom formule $\mathcal{A}$ *logicky vyplývá* z teorie T, což zapíšeme $T \models \mathcal{A}$.

Závěr: Lze ukázat, že $T \vdash \mathcal{A}$ právě tehdy, když $T \models \mathcal{A}$,

tj. formule $\mathcal{A}$ je *formálně dokazatelná* (též *odvoditelná*) z T právě tehdy, když tato formule z T *logicky vyplývá*.

Př.: $\underset{(T_1)}{A \rightarrow B}, \; \underset{(T_2)}{B \rightarrow C} \;\vdash\; A \rightarrow C$

1) Axiom 6.2 $\quad \big(\boldsymbol{A} \rightarrow (\boldsymbol{B} \rightarrow \boldsymbol{C})\big) \rightarrow ((\boldsymbol{A} \rightarrow \boldsymbol{B}) \rightarrow (\boldsymbol{A} \rightarrow \boldsymbol{C}))$

2) Axiom 6.1 $\quad (\boldsymbol{B} \rightarrow \boldsymbol{C}) \rightarrow (\boldsymbol{A} \rightarrow (\boldsymbol{B} \rightarrow \boldsymbol{C}))$

3) Mimologický axiom T_2 $\quad \boldsymbol{B} \rightarrow \boldsymbol{C}$

4) Modus ponens (3) a (2) $\quad \boldsymbol{A} \rightarrow (\boldsymbol{B} \rightarrow \boldsymbol{C})$

5) Modus ponens (4) a (1) $\quad (\boldsymbol{A} \rightarrow \boldsymbol{B}) \rightarrow (\boldsymbol{A} \rightarrow \boldsymbol{C})$

6) Mimologický axiom T_1 $\quad \boldsymbol{A} \rightarrow \boldsymbol{B}$

7) Modus ponens (6) a (5) $\quad \boldsymbol{A} \rightarrow \boldsymbol{C}$

Predikátová logika

Predikátová logika je logický formalismus sloužící především k symbolickému zápisu logického odvozování v matematické logice.

Poznatky zapsané symbolikou predikátové logiky – *jazykem predikátové logiky* – vyjádříme formulemi jazyka predikátové logiky.

Interpretací symbolů, které ve formulích vystupují, získáme výroky, jimž lze přiřadit pravdivostní hodnotu.

Jazyk predikátové logiky obsahuje spojky, proměnné, funkční symboly, predikátové symboly a symboly kvantifikátorů.

Přirozený jazyk je velmi bohatý a často nejednoznačný, což je také jeden z důvodů formalizace jazyka logiky, aby byl vyjadřovací prostředek unifikovaný a zcela jednoznačný.

Př.: *Logické a významové paradoxy*

Paradox lháře:

Někdo říká: „Lžu."

- Pokud v tomto okamžiku lže, pak není pravda, co říká, tedy nelže.
- Jestliže v tomto okamžiku nelže, pak je pravda, co říká, tzn., že lže, takže by v obou případech musel lhát a nelhat současně, což zřejmě není možné.

Jazyk predikátové logiky prvního řádu

Jazyk predikátové logiky poskytuje vyjadřovací prostředky nejen pro vyjádření *stavu úlohy*, ale i k popisu *pravidel* a *znalostí*. Je v tomto směru výrazně bohatější než jazyk výrokové logiky.

Jazyk výrokové logiky zkoumá pouze, zda jsou výroky pravdivé či nepravdivé. V jazyce predikátové logiky máme oproti tomu k dispozici symboly pro pojmenování *popisovaných objektů* a také symboly, které pojmenovávají *vlastnosti objektů* a *vztahy* mezi nimi.

Pro zápis vlastností objektů a relací mezi nimi zavádíme tzv. **predikáty** – symboly pro vyjádření obecně n-árních relací.

Pro popis vlastností používáme jednomístné (*unární*) predikáty, pro popis relací predikáty vícemístné (*binární, n-ární*).

Pro množstevní vyjádření používáme *kvantifikaci objektů* a pro její symbolický zápis tzv. **kvantifikátorů**.

Pro všeobecnou kvantifikaci (platící pro všechny objekty, vlastnosti, relace, atd.) používáme symbol *obecného kvantifikátoru* $\forall$, pro vyjádření, že existuje alespoň jeden objekt, jedna vlastnost,… pak *existenční kvantifikátor* $\exists$.

Kvantifikujeme-li v popisech našich poznatků *pouze objekty*, potom hovoříme o použití **predikátové logiky prvního řádu**, kvantifikujeme-li i vlastnosti a relace, pak hovoříme o použití predikátové logiky druhého, resp. vyšších řádů.

Dále se budeme zabývat pouze predikátovou logikou prvního řádu, protože pouze ta je formálně dokazatelná (predikátové logiky vyšších řádů jsou pouze částečně dokazatelné či nedokazatelné).

Syntax jazyka predikátové logiky prvního řádu popíšeme *abecedou symbolů* a *postupem*, jak jsou symboly skládány, aby tvořily *slova* jazyka. Tato slova budeme nazývat *formulemi* jazyka predikátové logiky prvního řádu.

Kromě základních symbolů abecedy budeme používat závorky a někdy též predikátové symboly nebo individuové konstanty tvořené více písmeny tak, aby byl zřejmý jejich význam.

Př.: *Sestavení formulí predikátové logiky*

- Někdo má hudební sluch (S) a někdo nemá hudební sluch.

$$(\exists x\ S(x)) \land (\exists x\ \neg S(x))$$

Abecedu jazyka tvoří:

1. *individuové proměnné* – značíme malými písmeny x, y, z, …
2. *individuové konstanty* – zapisujeme velkými písmeny A, B, C, …
3. *funkční symboly* – značíme malými písmeny f, g, h, …, každý funkční symbol má stanoven počet argumentů (četnost), které přijímá
4. *predikátové symboly* - označujeme velkými písmeny P, Q, R, …, každý predikátový symbol má stanoven počet argumentů (místnost), které přijímá
5. *symboly logických spojek* $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\Leftrightarrow$
6. *symboly kvantifikátorů* $\forall$, $\exists$

Ze symbolů tvoříme *výrazy* jazyka predikátové logiky prvního řádu:

a) *Term* je
- individuová konstanta nebo proměnná
- výraz tvaru $f(t_1, t_2, \ldots, t_n)$, kde f je n-četný funkční symbol a $t_1, t_2, \ldots, t_n$ jsou termy

b) *Atomická formule* je výraz tvaru $P(t_1, t_2, \ldots, t_m)$, kde P je m-místný predikátový symbol a $t_1, t_2, \ldots, t_m$ jsou termy

c) *Formule* je
- atomická formule
- jeden z výrazů $\neg\mathcal{A}$, $\mathcal{A} \wedge \mathcal{B}$, $\mathcal{A} \vee \mathcal{B}$, $\mathcal{A} \rightarrow \mathcal{B}$, $\mathcal{A} \Leftrightarrow \mathcal{B}$, $(\forall x)\mathcal{A}$, $(\exists x)\mathcal{A}$, kde $\mathcal{A}$, $\mathcal{B}$ jsou formule a x je individuová proměnná

d) *Individuová proměnná* se nazývá **vázaná**, když se vyskytuje v oblasti působnosti obecného nebo existenčního kvantifikátoru. Individuová proměnná, která není vázaná, se nazývá **volná**

e) *Uzavřená formule* je formule, ve které se nevyskytují volné proměnné

f) *Literál* je atomická formule nebo její negace

g) *Disjunkci literálů* nazýváme **klauzulí**, někdy též *disjunktem*

Interpretace formulí predikátové logiky 1. řádu

Vyjádření sémantiky formule – přiřazení významu výrazům – použijeme relační strukturu:

Označíme $\mathcal{M}$ relační strukturu, která je tvořena:

- nosičem D
- relacemi $R_i^n \subset D^n$
- operacemi $O_i^n : D^n \rightarrow D$

Symboly jazyka predikátové logiky 1. řádu interpretujeme takto:

a) Každé individuové konstantě A přiřadíme prvek $A_{\mathcal{M}}$ nosiče D, $A_{\mathcal{M}} \in D$.

b) Každému funkčnímu symbolu f_i přiřadíme operaci O_i stejné četnosti. Termům, které neobsahují proměnné, odpovídají elementy nosiče, které získáme provedením příslušné operace O_i s elementy nosiče přiřazenými podle bodu a).

c) Každému predikátovému symbolu P_i o místnosti k přiřadíme k−ární relaci R_i.

Potom relační strukturu $\mathcal{M} = (D, O, R)$ nazveme **interpretační strukturou**.

Poznámka: Někdy volíme symboly jazyka tak, aby vyjadřovaly určitou standardní interpretaci. Například jako individuové konstanty zavedeme obvyklá jména lidí – *KAREL*, *JOSEF*, …, jako predikátové konstanty pojmenování vztahů mezi nimi – *BRATR*(*x, y*), *SYN*(*otec, matka, potomek*) atd.

Pravdivostní hodnoty přiřazujeme při interpretaci formulí takto:

A) Uzavřená atomická formule $P(t_1, t_2, \dots, t_n)$ je pravdivá v interpretační struktuře $\mathcal{M}$, jestliže predikátovému symbolu P je přiřazena relace R a prvky nosiče D odpovídající termům $t_1, t_2, \dots, t_n$ jsou v relaci R.

Atomická formule P je pravdivá v interpretační struktuře $\mathcal{M}$, jestliže ji v této relační struktuře všechny interpretace I splňují. Zapíšeme:

$$\mathcal{M} \models P$$

B) Uzavřená formule, tvořená atomickými formulemi, logickými spojkami a kvantifikátory, má pravdivostní hodnotu určenou pravdivostními hodnotami atomických formulí, interpretací logických spojek podle obecně známých pravidel a interpretací kvantifikátorů.

Kvantifikovaná formule $(\forall x)P(x)$ znamená *konjunkci* formulí $P(x)$ přes všechna možná přiřazení symbolu x prvkům nosiče D

Kvantifikovaná formule $(\exists x)P(x)$ znamená *disjunkci* formulí $P(x)$ přes všechna možná přiřazení symbolu x prvkům nosiče D

Podobně jako ve výrokové logice platí:

- Formule B logicky vyplývá z formule A, když formule B je pravdivá ve všech interpretačních strukturách, ve kterých je pravdivá formule A.
- Formule A, která je logicky pravdivá v každé interpretační struktuře $\mathcal{M}$, se nazývá *tautologie* .
- Formule A a B nazýváme logicky ekvivalentní, jestliže A logicky vyplývá z B a B logicky vyplývá z A .

Axiomy predikátové logiky 1. řádu

Platí dříve uvedené axiomy výrokové logiky (6.1) až (6.3):

- $(\mathcal{A} \rightarrow (\mathcal{B} \rightarrow \mathcal{A}))$ (6.1)
- $((\mathcal{A} \rightarrow (\mathcal{B} \rightarrow \mathcal{C})) \rightarrow ((\mathcal{A} \rightarrow \mathcal{B}) \rightarrow (\mathcal{A} \rightarrow \mathcal{C})))$ (6.2)
- $((\neg\mathcal{B} \rightarrow \neg\mathcal{A}) \rightarrow ((\neg\mathcal{B} \rightarrow \mathcal{A}) \rightarrow \mathcal{B}))$ (6.3)

Dále platí:

- $(\forall x)\, \mathcal{A}(x) \rightarrow \mathcal{A}(t)$ (6.4)

  pokud term t neobsahuje proměnné vázané v $\mathcal{A}$.
- $(\forall x)(\mathcal{A} \rightarrow \mathcal{B}) \rightarrow (\mathcal{A} \rightarrow (\forall x)\, \mathcal{B})$ (6.5)

  když formule $\mathcal{A}$ neobsahuje volnou proměnnou x .

Odvozovacími pravidly jsou:

- Pravidlo modus ponens:
  $(\mathcal{A} \wedge (\mathcal{A} \rightarrow \mathcal{B})) \rightarrow \mathcal{B}$

- Pravidlo generalizace:
  $\mathcal{A} \rightarrow (\forall x)\, \mathcal{A}$ ($(\forall x)\, \mathcal{A}$ vyplývá z $\mathcal{A}$)

Definice: Formule $\mathcal{A}$ predikátové logiky 1. řádu je formálně dokazatelná z axiomů predikátové logiky 1. řádu, jestliže existuje důkaz využívající uvedená dvě odvozovací pravidla.

Odvozování v predikátové logice 1. řádu

Podobně jako v případě výrokové logiky můžeme k axiomům přidat další formule – *mimologické axiomy*. Množinu mimologických axiomů T pak nazveme *teorií* jazyka predikátové logiky 1. řádu.

Dedukcí, tj. použitím odvozovacích pravidel, můžeme z teorie T dokázat další formule. Jestliže formule $\mathcal{A}$ je *formálně dokazatelná* z teorie T (je teorémem v T), píšeme:

$$T \vdash \mathcal{A}$$

Nechť $\mathcal{M}$ je interpretační struktura. Jestliže všechny axiomy teorie T jsou v $\mathcal{M}$ pravdivé, říkáme, že $\mathcal{M}$ je *modelem* teorie T a píšeme:

$$\mathcal{M} \models T$$

Řekneme, že formule $\mathcal{A}$ je pravdivá v teorii T, jestliže je pravdivá v každém jejím modelu. Píšeme:

$$T \models \mathcal{A}$$

Lze dokázat (podle Gödela), že formule jazyka predikátové logiky je formálně dokazatelná z teorie T právě tehdy, když je pravdivá v T:

$$T \vdash \mathcal{A} \quad \Leftrightarrow \quad T \models \mathcal{A}$$

Shrnutí definic

1. Je-li $\mathcal{A}$ správně vytvořená formule, která je pravdivá v interpretaci $\mathcal{M}$, potom $\mathcal{M}$ je **modelem** $\mathcal{A}$.
2. **Logicky pravdivá formule** (tautologie) je formule, která je pravdivá ve všech svých interpretacích.
3. Formule je **splnitelná**, jestliže existuje **alespoň jedna** interpretace, ve které je pravdivá.
4. Formule **není logicky pravdivá**, jestliže není pravdivá alespoň v jedné své interpretaci.
5. Formule je **nesplnitelná** (kontradikce), jestliže není pravdivá v žádné své interpretaci.

6. **Literál** je atomická formule (atom) nebo negace atomu.
7. **Disjunktivní normální forma** (DNF):

$$A = A_1 \vee \dots \vee A_n\,, \quad n \geq 1$$

   A_i … konjunkce literálů
8. **Konjunktivní normální forma** (KNF):

$$A = A_1 \wedge \dots \wedge A_n\,, \quad n \geq 1$$

   A_i … disjunkce literálů
9. Disjunkce literálů $L_1 \vee \dots \vee L_n$ se nazývá **klauzule**.
10. Klauzule obsahující 1 literál je **jednotková** klauzule.
11. Klauzule neobsahující žádný literál je **prázdná** klauzule.
12. Klauzule obsahující komplementární pár je **tautologie**.

Rezoluční metoda a dokazování teorémů

Dokazování formulí jazyka výrokové logiky či jazyka predikátové logiky 1. řádu z axiomů příslušného logického kalkulu výše uvedenými odvozovacími pravidly je sice korektní, ale z hlediska výpočetní složitosti neefektivní.

V roce 1965 přišel *J. A. Robinson* s postupem odvozování, který vede k návrhu algoritmů použitelných pro automatické dokazování formulí výrokové a predikátové logiky. Tento postup byl později nazván **rezoluční metodou**, resp. **rezolučním principem**.

Důkaz, že vyšetřovaný teorém logicky vyplývá z axiomů (z dané teorie T), je při použití rezoluční metody založen na tvrzení, že vyplývá-li teorém z dané teorie, pak neexistuje model teorie T, v němž by byla pravdivá negace dokazovaného teorému. Axiomy z teorie T a negace dokazovaného teorému tak musí vést ke ***sporu***.

Rezoluční metoda ve výrokové logice

Aplikace rezoluční metody na formule výrokové logiky vyžaduje, aby formule teorie T byly ve tvaru *klauzulí*, tj. *disjunktů literálů*. Literálem v jazyce výrokové logiky nazveme výrokové proměnné nebo jejich negace.

Jestliže formule teorie T nejsou klauzulemi, nahradíme je logicky ekvivalentními formulemi, které jsou klauzulemi, nebo postupujeme podle následujícího algoritmu: Není-li nějaká formule $\mathcal{A}$ z teorie T klauzulí, upravíme ji na tvar

$$C_1 \wedge C_2 \wedge \ldots \wedge C_n,$$

kde C_i jsou klauzule.

Z teorie T vynecháme formuli $\mathcal{A}$ a přidáme klauzule $\mathcal{C}_1$, $\mathcal{C}_2$, …, $\mathcal{C}_n$. Nově vzniklou teorii označíme T' a bude mít tvar

$$T' = (T - \{\mathcal{A}\}) \cup \{ \mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n \}$$

Každý model teorie T' je také modelem teorie T a naopak. V modelu, v němž je formule $\mathcal{A}$ pravdivá, je pravdivá také formule $\mathcal{C}_1 \wedge \mathcal{C}_2 \wedge \ldots \wedge \mathcal{C}_n$, a proto musejí být v tomto modelu pravdivé také klauzule $\mathcal{C}_1$, $\mathcal{C}_2$, …, $\mathcal{C}_n$.

V dalším mějme dány dvě klauzule ve tvaru

$$A \vee L_1 \vee L_2 \vee \ldots \vee L_n \quad \text{a} \quad \neg A \vee M_1 \vee M_2 \vee \ldots \vee M_m,$$

kde A, L_i, M_j jsou literály, $L_i \neq M_j$ pro $i = 1, \ldots, n$, $j = 1, \ldots, m$. Tyto klauzule nazveme *rodičovské*.

Rodičovské klauzule určené k rezoluci se vyznačují tím, že obsahují tzv. *pár komplementárních literálů* – jedna z rodičovských klauzulí obsahuje literál A, druhá pak jeho negaci $\neg A$. Rezolucí odvodíme z těchto dvou rodičovských klauzulí novou klauzuli, kterou nazveme jejich **rezolventou**. Rezolventa vznikne disjunkcí rodičovských klauzulí s vynecháním páru komplementárních literálů, čili

$$L_1 \vee L_2 \vee \ldots \vee L_n \vee M_1 \vee M_2 \vee \ldots \vee M_m.$$

Jestliže k množině klauzulí (teorii T, resp. T') neexistuje žádný model, v němž by negace teorému byla pravdivá, lze odvodit postupným opakováním rezoluční metody *prázdnou klauzuli* (budeme ji označovat $\square$), která *není pravdivá a představuje spor*.

Př.: Mějme formuli $T = \{ \neg A \vee B, \neg B \vee C, \neg C \vee D, \neg D \vee \neg E \}$, máme dokázat, že i formule $\neg A \vee \neg E$ je logicky pravdivá:

Rezoluční metodou dokážeme, že spojení T s formulí $\neg(\neg A \vee \neg E)$ vede ke sporu.

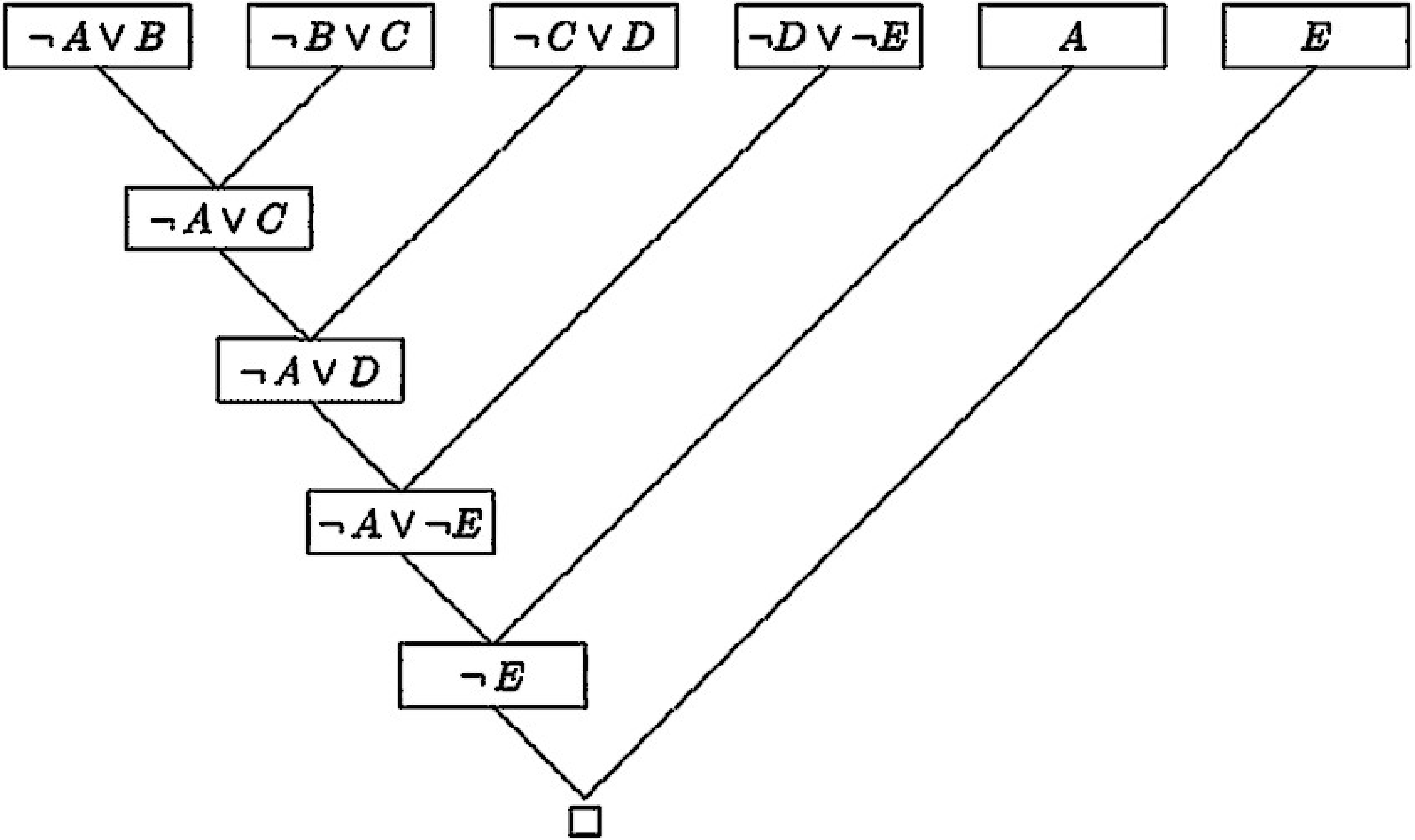
- $T' = \{ \neg A \vee B, \neg B \vee C, \neg C \vee D, \neg D \vee \neg E, \neg(\neg A \vee \neg E) \}$

  rozšíření o dokazovanou formuli

- $T' = \{ \neg A \vee B, \neg B \vee C, \neg C \vee D, \neg D \vee \neg E, A, E \}$

  po úpravě

Postup rezoluční metody znázorníme derivačním stromem:


$\neg A \vee B$
$\neg B \vee C$
$\neg C \vee D$
$\neg D \vee \neg E$
A
E
$\neg A \vee C$
$\neg A \vee D$
$\neg A \vee \neg E$
$\neg E$
$\square$

Rezoluční metoda v predikátové logice 1. řádu

V jazyce predikátové logiky je hledání páru komplementárních literálů obtížnější, protože modely tohoto jazyka jsou složitější. Je třeba porovnávat odpovídající si termy a hledat vhodné *substituce* v klauzulích.

Např. literály $P(x, f(A))$ a $\neg P(B, z)$ se stanou komplementárním párem, pokud proměnná x je rovna konstantě B a proměnná z termu $f(A)$.

Nalezení vhodné substituce s je důležitým krokem rezolučního dokazování formulí jazyka predikátové logiky 1. řádu. Spočívá v nahrazení všech výskytů proměnné x_i termem t_i. Substituci zapíšeme

$$s = \{ t_1/x_1, t_2/x_2, \ldots, t_n/x_n \}.$$

Term t_i, kterým substituujeme, nesmí obsahovat proměnnou x_i, kterou nahrazuje.

Pro zjednodušení zápisu zapíšeme opět klauzule jako množiny literálů, např. klauzuli

$C = L_1 \lor L_2 \lor \ldots \lor L_n$ zapíšeme $C = \{ L_1, L_2, \ldots, L_n \}$.

Definice: Literál L' nazveme **instancí** literálu L, jestliže jej získáme nějakou substitucí s v literálu L. Pak platí $L' = L\,s$.

Příklad: Literály $L' = Q(x, A, f(C))$ a $L'' = Q(B, A, f(C))$ jsou instancemi literálu $L = Q(x, A, f(y))$, v němž jsme postupně aplikovali $s_1 = \{ C/y \}$ a $s_2 = \{ B/x \}$ takové, že platí:

$$L' = L\,s_1 \quad \text{a} \quad L'' = L'\,s_2 = L\,s_1\,s_2$$

Definice: Instance, která neobsahuje proměnné, se nazývá **základní instance**.

Definice: Substituce s se nazývá **unifikátor** klauzule $C = \{ L_1, L_2, \dots, L_n \}$, jestliže platí:

$$L_1\,s = L_2\,s = \dots = L_n\,s$$

Příklad: Substituce $s = \{ B/x, A/z, C/y, f(C)/w \}$ je unifikátorem klauzule $C = \{ Q(x, A, f(y)), Q(B, z, w) \}$. Protože oba literály mají po provedení substituce tvar $Q(B, A, f(C))$, stávají se základními instancemi.

Jednodušším unifikátorem je substituce $g = \{ B/x, A/z, f(y)/w \}$. Její aplikací získáme instance $Q(B, A, f(y))$, které nejsou základní, díky čemuž lze v následujících krocích aplikovat další substituce.

Definice: Unifikátor g klauzule C se nazývá **nejobecnější unifikátor** této klauzule, jestliže pro libovolný jiný unifikátor s klauzule C platí, že:

$C\,s$ je instancí klauzule $C\,g$.

Mějme nyní klauzule C_1, C_2 s navzájem různými proměnnými:

$$C_1 = \{ L_i \} \ a \ C_2 = \{ M_j \} \quad i = 1, 2, \dots, n \quad j = 1, 2, \dots, m$$

Dále mějme klauzule C_1' a C_2' takové, že

$$C_1' = \{ L_i' \} \subset C_1 \quad C_2' = \{ M_j' \} \subset C_2$$

a mějme g jako nejobecnější unifikátor klauzule $C_1' \cup \{\neg \mathrm{M}_\mathrm{j}'\}$.

Potom klauzule

$$(C_1 - C_1')\, g \ \cup \ (C_2 - C_2')\, g$$

je **rezolventou** klauzulí C_1 a C_2 .

Př.: *Pozorováním vymezené oblasti (zde zoologie) jsme zjistili následující poznatky:*

1. Každá ryba má žábry.
2. Savci nemají žábry.
3. Někteří savci dovedou plavat.

Úkolem je dokázat tvrzení (závěr), že:

Někteří živočichové dovedou plavat a přitom nejsou ryby.

V jazyce predikátové logiky 1. řádu vyjádříme tyto poznatky následovně:

- individuová proměnná *x* bude reprezentovat živočichy
- predikátový symbol *RYBA* bude přiřazen vlastnosti „být rybou“
- predikátový symbol *MÁ_ŽÁBRY* označuje živočichy se žábrami
- predikát *SAVEC(x)* představuje, že živočich *x* je savec
- predikát *PLAVE(x)* vyjadřuje, že živočich *x* umí plavat

Atomická formule *RYBA(x)* bude pravdivá tehdy, pokud označuje rybu, formule *PLAVE(x)* bude pravdivá, pokud živočich dovede plavat.

Formule predikátové logiky 1. řádu zapíšeme:

1. $(\forall x)(RYBA(x) \rightarrow MÁ\_ŽÁBRY(x))$
2. $(\forall x)(SAVEC(x) \rightarrow \neg MÁ\_ŽÁBRY(x))$
3. $(\exists x)(SAVEC(x) \wedge PLAVE(x))$

Úkolem je dokázat platnost teorému

4. $(\exists x)(PLAVE(x) \wedge \neg RYBA(x))$

Tvrzení teorému je logicky pravdivé tehdy a jen tehdy, pokud formule

$$[(\forall x)(RYBA(x) \rightarrow MÁ\_ŽÁBRY(x)) \wedge (\forall x)(SAVEC(x) \rightarrow \neg MÁ\_ŽÁBRY(x)) \wedge \wedge (\exists x)(SAVEC(x) \wedge PLAVE(x))] \rightarrow (\exists x)(PLAVE(x) \wedge \neg RYBA(x))$$

je logicky pravdivá, resp. formule

$$[(\forall x)(RYBA(x) \rightarrow MÁ\_ŽÁBRY(x)) \wedge (\forall x)(SAVEC(x) \rightarrow \neg MÁ\_ŽÁBRY(x)) \wedge \wedge (\exists x)(SAVEC(x) \wedge PLAVE(x))] \wedge \neg (\exists x)(PLAVE(x) \wedge \neg RYBA(x))$$

je nesplnitelná pro jakoukoli hodnotu proměnné x .

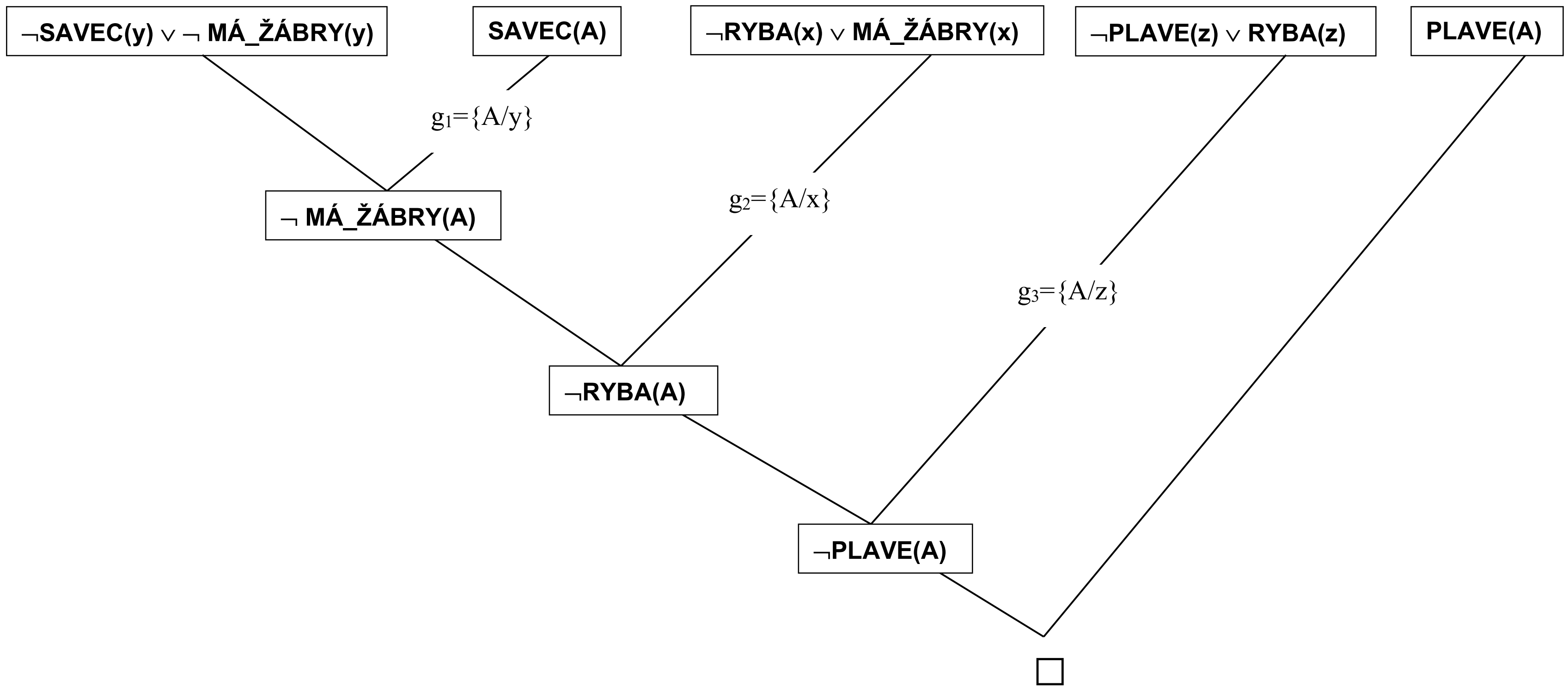
Převedením formulí do klauzulárního tvaru dostaneme

1. $\neg$ *RYBA(x)* $\vee$ *MÁ_ŽÁBRY(x)*
2. $\neg$ *SAVEC(y)* $\vee$ $\neg$ *MÁ_ŽÁBRY(y)*
3. *SAVEC(A)*
   PLAVE(A)
4. $\neg$ *PLAVE(z)* $\vee$ *RYBA(z)*

resp.

1. {$\neg$ *RYBA(x), MÁ_ŽÁBRY(x)*}
2. {$\neg$ *SAVEC(y),* $\neg$ *MÁ_ŽÁBRY(y)*}
3. {*SAVEC(A)*}
   {*PLAVE(A)*}
4. {$\neg$ *PLAVE(z), RYBA(z)*} .

Důkaz odvodíme opakovaným použitím rezoluční metody na klauzule *1* až *4* a rozšiřováním množiny klauzulí o rezolventy – viz následující obrázek:



PRENEXNÍ NORMÁLNÍ FORMA FORMULÍ

Určování logické pravdivosti formulí → postupy unifikovány → na základě speciálního typu formulí predikátové logiky se určuje jejich *logická pravdivost*, resp. *nesplnitelnost*.

Definice: Formule *A* se nachází v *prenexní normální formě*, pokud má tvar

$$(Q_1 x_1)\ldots(Q_n x_n)\ M,$$

kde $(Q_i x_i)$, $i=1,\ldots,n$ je *prefix formule* $(\forall x_i)$ nebo $(\exists x_i)$,

M je tzv. *matice* formule, která neobsahuje *žádné kvantifikátory*.

Tvrzení: Pomocí ekvivalentních úprav se dá každá správně vytvořená formule predikátové logiky *převést do PNF*.

Pravidla pro převod formule do PNF:

(a) $(Qx)F(x) \vee G = (Qx)[F(x) \vee G]$

(b) $(Qx)F(x) \wedge G = (Qx)[F(x) \wedge G]$

- dosah kvantifikátorů se dá rozšířit, pokud nedojde ke změně volné proměnné na vázanou

(c) $\neg ((\forall x)F(x)) = (\exists x)(\neg F(x))$

(d) $\neg ((\exists x)F(x)) = (\forall x)(\neg F(x))$

Pravidla pro vytknutí kvantifikátorů z matice do prefixu:

(e) $(\forall x)F(x) \wedge (\forall x)H(x) = (\forall x)(F(x) \wedge H(x))$

(f) $(\exists x)F(x) \vee (\exists x)H(x) = (\exists x)(F(x) \vee H(x))$

(g) $(Q_1x)F(x) \vee (Q_2x)H(x) = (Q_1x)(Q_2z)(F(x) \vee H(z))$

(h) $(Q_3x)F(x) \wedge (Q_4x)H(x) = (Q_3x)(Q_4z)(F(x) \wedge H(z))$

Algoritmus převodu správně vytvořených formulí do PNF

1) Eliminace kvantifikátorů $(\forall x)$ nebo $(\exists x)$, v jejichž dosahu se nevyskytuje kvantifikovaná proměnná.
2) Odstranění spojek $\rightarrow$ a $\leftrightarrow$ (pravidla *1*, *2*).
3) Opakované použití pravidla *9* a pravidel *(a)* až *(d)* pro přesun všech negací těsně k atomům.
4) Provedení všech potřebných přejmenování proměnných.
5) Aplikace pravidel pro vytknutí kvantifikátorů z matice formule do jejího prefixu – pravidla *(e)* až *(h)*.

Poznámka: Je-li matice M formule A v konjunktivní formě, pak formu nazýváme konjunktivní prenexní normální formou. Obdobně hovoříme o disjunktivní prenexní normální formě.

Př.: Převeďte do PNF formuli

$$(\forall x)[(\forall y)(P(x) \vee (\forall z)Q(z,y)) \rightarrow \neg (\forall y)R(x,y)]$$

Řešení:

$$(\forall x)[(\forall y)(P(x) \vee (\forall z)Q(z,y)) \rightarrow \neg (\forall y)R(x,y)] \quad \Rightarrow_{(1,2)}$$

$$(\forall x)[\neg ((\forall y)(P(x) \vee (\forall z)Q(z,y))) \vee \neg (\forall y)R(x,y)] \quad \Rightarrow_{(3)}$$

$$(\forall x)[(\exists y)(\neg P(x) \wedge (\exists z) \neg Q(z,y)) \vee (\exists y) \neg R(x,y)] \quad \Rightarrow_{(4)}$$

$$(\forall x)[\neg P(x) \wedge (\exists y)(\exists z) \neg Q(z,y) \vee (\exists y_1) \neg R(x,y_1)] \quad \Rightarrow_{(5)}$$

$$(\forall x)(\exists z)(\exists y)(\exists y_1)[(\neg P(x) \wedge \neg Q(z,y)) \vee \neg R(x,y_1)] \quad \Rightarrow$$

$$(\forall x)(\exists z)(\exists y)(\exists y_1)[(\neg P(x) \vee \neg R(x,y_1)) \wedge (\neg Q(z,y) \vee \neg R(x,y_1))]$$

Převod formulí predikátové logiky 1. řádu do klauzulárního tvaru

Převod obecné formule *F* jazyka predikátové logiky prvního řádu, např.

$$(\forall x)\{P(x) \rightarrow \{(\forall y)[P(y) \vee Q(x,y)] \;\; \wedge \;\; \neg(\forall y)[P(y) \rightarrow Q(y,x)]\}\}$$

na *klauzulární tvar* provedeme v deseti krocích:

1. Není-li formule *F* uzavřená, tj. obsahuje-li volné proměnné, vytvoříme existenční uzávěr formule *F* tak, že všechny volné proměnné kvantifikujeme existenčním kvantifikátorem. Uzávěr *F* zapíšeme $(\exists x)\ldots(\exists z)\,(F)$, kde symboly $x, \ldots, z$ reprezentují všechny volné proměnné ve formuli *F*.

2. Ve formuli *F* se vyskytující ekvivalence a implikace nahradíme ekvivalentními formulemi:

$$(\forall x)\{P(x) \rightarrow \{(\forall y)[P(y) \vee Q(x,y)] \ \wedge \ \neg(\forall y)[P(y) \rightarrow Q(y,x)]\}\}$$

$$(\forall x)\{\neg P(x) \vee \{(\forall y)[P(y) \vee Q(x,y)] \ \wedge \neg(\forall y)[\neg P(y) \vee Q(y,x)]\}\}$$

3. Upravíme formuli tak, aby se negace vztahovaly jen na atomy, tj. převedeme negace „dovnitř“ jednotlivých výrazů následovně:

$$\neg(\exists x)P(x) \quad \Leftrightarrow \quad (\forall x)(\neg P(x))$$

$$\neg(\forall x)P(x) \quad \Leftrightarrow \quad (\exists x)(\neg P(x))$$

Dále použijeme De Morganova pravidla a získáme tvar:

$$(\forall x)\{\neg P(x) \vee \{(\forall y)[P(y) \vee Q(x,y)] \ \wedge \ (\exists y)[P(y) \wedge \neg Q(y,x)]\}\}$$

4. Přejmenujeme kvantifikované proměnné tak, aby se v rámci platnosti kvantifikátoru lišily (odstranění víceznačnosti):

$$(\forall x)\{\neg P(x) \vee \{(\forall y)[P(y) \vee Q(x,y)] \;\wedge\; (\exists y)[P(y) \wedge \neg Q(y,x)]\}\}$$

$$(\forall x)\{\neg P(x) \vee \{(\forall y)[P(y) \vee Q(x,y)] \;\wedge\; (\exists z)[P(z) \wedge \neg Q(z,x)]\}\}$$

5. Vyloučíme existenční kvantifikátory: Výraz $(\forall x)(\exists z)P(z)$ znamená, že pro každé *x* existuje *z* takové, že platí *P*(*z*) − to vyjádříme *Skolemovou funkcí* *z* = *f*(*x*). Výraz $(\exists z)P(z)$ nahradíme Skolemovou funkcí bez argumentů − *z* = *A*, kde *A* je individuová konstanta, tedy $(\exists z)P(z)$ nahradíme $P(A)$.

$$(\forall x)\{\neg P(x) \vee \{(\forall y)[P(y) \vee Q(x,y)] \;\wedge\; [P(f(x)) \wedge \neg Q(f(x),x)]\}\}$$

6. Všeobecné kvantifikátory přesuneme na začátek formule, do *prefixu formule*. Zbytek formule nazýváme *maticí formule*. Tento tvar nazýváme *prenexní normální forma* formule:

$$(\forall x)\{\neg P(x) \vee \{(\forall y)[P(y) \vee Q(x,y)] \;\wedge\; [P(f(x)) \wedge \neg Q(f(x),x)]\}\}$$

$$(\forall x)(\forall y)\{\neg P(x) \vee \{[P(y) \vee Q(x,y)] \;\wedge\; [P(f(x)) \wedge \neg Q(f(x),x)]\}\}$$

7. Všechny proměnné formule jsou všeobecně kvantifikovány, na pořadí kvantifikátorů nezáleží. Formule je tak splněna pro všechny hodnoty, takže pro jednoduchost zápisu můžeme prefix vynechat:

$$\neg P(x) \vee \{[P(y) \vee Q(x,y)] \;\wedge\; [P(f(x)) \wedge \neg Q(f(x),x)]\}$$

8. Matici formule převedeme na konjunkci klauzulí použitím ekvivalentních formulí:

$$\neg P(x) \vee \{[P(y) \vee Q(x,y)] \;\wedge\; [P(f(x)) \wedge \neg Q(f(x),x)]\}$$

$$[\neg P(x) \vee P(y) \vee Q(x,y)] \wedge \{\neg P(x) \vee [P(f(x)) \wedge \neg Q(f(x),x)]\}$$

$$[\neg P(x) \vee P(y) \vee Q(x,y)] \wedge [\neg P(x) \vee P(f(x))] \wedge [\neg P(x) \vee \neg Q(f(x),x)]$$

9. Výrazy v hranatých závorkách jsou nyní klauzulemi. Konjunkci klauzulí lze nahradit množinou klauzulí:

$$\left\{\begin{array}{c} [\neg P(x) \vee P(y) \vee Q(x,y)], \\ \big[\neg P(x) \vee P\big(f(x)\big)\big], \\ [\neg P(x) \vee \neg Q(f(x),x)] \end{array}\right\}$$

Klauzule lze dále vyjádřit jako množiny literálů:

$$\{\neg P(x), P(y), Q(x, y)\},$$
$$\{\neg P(x), P\big(f(x)\big)\},$$
$$\{\neg P(x), \neg Q(f(x), x)\}$$

10. Proměnné přejmenujeme tak, aby každá proměnná byla obsažena nejvýše v jedné klauzuli:

$$\{\neg P(a), P(b), Q(a, b)\},$$
$$\{\neg P(c), P\big(f(c)\big)\},$$
$$\{\neg P(d), \neg Q(f(d), d)\}$$

Původní logickou formuli jsme tak převedli na množinu klauzulí neobsahujících stejné proměnné, zapsaných jako množiny literálů.

ZÁVĚR

Rezoluční metoda umožňuje dokazování teorémů z dané výchozí teorie, není však obecným předpisem k řešení zadaného problému, protože např. jednoznačně neurčuje, které klauzule v daném kroku vybrat pro rezoluci jako rodičovské. Generování všech možných rezolucí vede ke kombinatorické explozi, a proto je třeba zvolit vhodnou řídicí strategii.

Nejjednodušší možností je volba *prohledávání do šířky*, to je ale značně neefektivní.

Jiné strategie např. požadují, aby jako rodičovská byla přednostně vybrána klauzule, kterou tvoří jediný literál, neboť rezolventa bude obsahovat méně literálů než druhá z rodičovských klauzulí, dále lze nalézt strategii, která požaduje, aby alespoň jedna z rodičovských klauzulí byla negací odvozeného teorému nebo byla z této klauzule v některém z předchozích kroků odvozena.

Mezi efektivnější postupy patří strategie, které *minimalizují prohledávaný derivační strom*.