

# Logika

Dana Nejedlová

Katedra informatiky

Ekonomická fakulta

Technická univerzita v Liberci

# Úloha logiky v umělé inteligenci

- převést fakta na formalizované výroky, se kterými se dá automatizovaně operovat tak, že se z počáteční množiny výroků (axiomů) postupně vyvozují nová fakta (axiomy) až se mezi nimi objeví dokazovaný teorém
- Prvním známým formálním systémem takové činnosti byly Aristotelovy sylogismy pro dedukci.
  - Například: Kapr je ryba. Všechny ryby žijí ve vodě. A proto kapr žije ve vodě.
- V současnosti se automatizované systémy pro logická dokazování využívají
  - pro verifikaci softwaru a hardwaru,
  - jako pomoc matematikům při provádění složitých důkazů,
  - v expertních systémech pro libovolnou problematiku.

# Monotónnost logiky

- Výchozím předpokladem úspěšnosti uplatňování metod logického usuzování je monotónnost logiky.
- Monotónnost znamená, že přidáním nového axiomu mezi ty stávající se nezmenší množství nových tvrzení, které lze dokázat. Jinak řečeno, mezi axiomy, ze kterých se má dokázat teorém, nesmí být spor.
- Příkladem nemonotónního uvažování je budování znalostní báze, při kterém se později narazí na výjimky. Například dítě se nejdříve dozví, že všichni ptáci létají a teprve potom se dozví o tučňácích.
- Systémy uplatňující nemonotónní logiku by měly lépe zpracovávat informace z reálného světa, ale nejsou příliš rozvinuté, protože složitost jejich algoritmů bude vždy exponenciální (množství axiomů je exponentem konstanty).

# Výroková logika

## (Propositional Calculus)

- Symboly reprezentují celé výroky.
- Výrok
  - je každá oznamovací věta, o jejíž pravdivosti lze jednoznačně rozhodnout,
  - má pravdivostní hodnotu pravda nebo nepravda,
  - může být spojován s jinými výroky logickými spojkami neboli operátory.
    - Dále nedělitelný výrok se nazývá **atomický výrok**.
- Účelem výrokové logiky je testování validity výroků typu logický úsudek.
  - Úsudek (**inference**) je logicky správný (**is sound**), když ze splnění všech předpokladů plyne splnění závěru, což se pozná tak, že na každém řádku **pravdivostní tabulky** (**truth table**, neboli tabulky pravdivostních hodnot), na němž mají všechny předpoklady hodnotu 1, má i závěr hodnotu 1.
  - Přitom má zvláštní význam logická spojka zvaná **implikace**. 4

# Pravdivostní tabulka logických operátorů

| Výrok 1 | Výrok 2 | Negace   | Konjunkce    | Disjunkce  | Implikace         | Ekvivalence           | XOR      |
|---------|---------|----------|--------------|------------|-------------------|-----------------------|----------|
| A       | B       | $\neg A$ | $A \wedge B$ | $A \vee B$ | $A \Rightarrow B$ | $A \Leftrightarrow B$ | $\oplus$ |
| 0       | 0       | 1        | 0            | 0          | 1                 | 1                     | 0        |
| 0       | 1       | 1        | 0            | 1          | 1                 | 0                     | 1        |
| 1       | 0       | 0        | 0            | 1          | 0                 | 0                     | 1        |
| 1       | 1       | 0        | 1            | 1          | 1                 | 1                     | 0        |

- 0 = nepravda, false
- 1 = pravda, true
- Ekvivalence:
  - A platí, když a jenom když platí B.

# Implikace

- $A \Rightarrow B$ ,  $A$  implikuje  $B$ 
  - $A$  je **premisa** neboli **antecedent**,  $B$  je **konkluze** neboli **konsekvent**.
- Intuitivně bývá vnímána jako „Když platí  $A$ , tak platí  $B$ “.
  - To je však chyba.
- Správně má být vykládána jako „Když platí  $A$ , tak platí  $B$ . V opačném případě (když  $A$  neplatí) se o  $B$  nedá nic tvrdit“.
- Programovací jazyky pro ni nemají operátor.
  - Místo toho se vyjadřuje jako  $\neg A \vee B$ .
  - Důkaz, že je výrok  $\neg A \vee B$  ekvivalentní s výrokem  $A \Rightarrow B$  je možné provést pomocí pravdivostní tabulky.
  - Když má výrok (zde ekvivalence) hodnotu pravda (neboli výrok platí) pro všechny možné kombinace vstupních hodnot  $A$ ,  $B$ , tak je dokázán.
- Výroky, které jsou za všech okolností pravdivé, se nazývají **tautologie**.
- Tautologie obsahující implikaci se využívají jako **inferenční pravidla**.

| $A$ | $B$ | $\neg A$ | $(\neg A) \vee B$ | $A \Rightarrow B$ | $(\neg A \vee B) \Leftrightarrow (A \Rightarrow B)$ |
|-----|-----|----------|-------------------|-------------------|-----------------------------------------------------|
| 0   | 0   | 1        | 1                 | 1                 | 1                                                   |
| 0   | 1   | 1        | 1                 | 1                 | 1                                                   |
| 1   | 0   | 0        | 0                 | 0                 | 1                                                   |
| 1   | 1   | 0        | 1                 | 1                 | 1                                                   |

# Inferenční pravidla

| Inferenční pravidlo | Konjunkce axiomů<br>Operátorem je čárka. | Zobecněná forma                                              | Využití                           |
|---------------------|------------------------------------------|--------------------------------------------------------------|-----------------------------------|
|                     | Konkluze implikace                       |                                                              |                                   |
| Modus ponens        | $A \Rightarrow B, A$                     | $A \wedge B \wedge C \Rightarrow D, A, B, C$                 | Expertní systémy                  |
|                     | $B$                                      | $D$                                                          |                                   |
|                     | $A \Rightarrow B, A$                     | $A \vee B \vee C \Rightarrow D, A \vee B \vee C$             |                                   |
|                     | $B$                                      | $D$                                                          |                                   |
| Modus tollens       | $A \Rightarrow B, \neg B$                | $A \wedge B \wedge C \Rightarrow D, \neg D$                  |                                   |
|                     | $\neg A$                                 | $\neg(A \wedge B \wedge C) = \neg A \vee \neg B \vee \neg C$ |                                   |
|                     | $A \Rightarrow B, \neg B$                | $A \vee B \vee C \Rightarrow D, \neg D$                      |                                   |
|                     | $\neg A$                                 | $\neg(A \vee B \vee C) = \neg A \wedge \neg B \wedge \neg C$ |                                   |
| Rezoluce            | $A \vee B, \neg B \vee C$                | $A \vee B \vee C, \neg A \vee D \vee E$                      | Systémy pro dokazování<br>teorémů |
|                     | $A \vee C$                               | $B \vee C \vee D \vee E$                                     |                                   |

# Inferenční pravidla

- Přidávají nová fakta (axiomy) do báze axiomů, ze které se vyvozují závěry (další axiomy).
- Lze je dokázat pravdivostními tabulkami.
  - Například důkaz rezoluce
    - spočívá v tom, že na všech řádcích je v posledním sloupečku pravda.

| A | B | C | $A \vee B$ | $\neg B \vee C$ | $D = ((A \vee B) \wedge (\neg B \vee C))$ | $A \vee C$ | $D \Rightarrow (A \vee C)$ |
|---|---|---|------------|-----------------|-------------------------------------------|------------|----------------------------|
| 0 | 0 | 0 | 0          | 1               | 0                                         | 0          | 1                          |
| 0 | 0 | 1 | 0          | 1               | 0                                         | 1          | 1                          |
| 0 | 1 | 0 | 1          | 0               | 0                                         | 0          | 1                          |
| 0 | 1 | 1 | 1          | 1               | 1                                         | 1          | 1                          |
| 1 | 0 | 0 | 1          | 1               | 1                                         | 1          | 1                          |
| 1 | 0 | 1 | 1          | 1               | 1                                         | 1          | 1                          |
| 1 | 1 | 0 | 1          | 0               | 0                                         | 1          | 1                          |
| 1 | 1 | 1 | 1          | 1               | 1                                         | 1          | 1                          |



# Rezoluce versus Modus ponens

- Rezoluce
  - Je kompletní, což znamená, že pokud fakt logicky vyplývá z ostatních faktů (pomocí kterých jej chceme dokázat), tak je pomocí rezoluce dokazatelný.
  - Pokud dokazovaný fakt logicky z ostatních faktů nevyplývá, tak není zaručeno zastavení algoritmu hledajícího důkaz.
    - To se nazývá **nerozhodnutelnost** (**semidecidability**) inferenčních algoritmů v predikátové logice 1. řádu.
    - Prakticky se řeší nastavením časového limitu, po kterém algoritmus skončí, a člověk potom může upravit zadání.
- Modus ponens
  - Není kompletní, což se projevuje tak, že ne každý fakt lze převést do formy umožňující jeho uplatnění.
  - Je nerozhodnutelný stejně jako rezoluce.
  - Forma umožňující jeho uplatnění se nazývá **Hornova klauzule** (**Horn clause, definite clause**).
  - Báze faktů (znalostí, axiomů) ve formě Hornových klauzulí umožňuje provést důkaz v polynomiálním čase (pokud to, co dokazujeme, skutečně vyplývá z ostatních faktů).
  - Používá jej jazyk Prolog.

# Univerzální metoda inference

- Pomocí kompletních pravdivostních tabulek lze dokázat vše, ale je to  $2^n$  složité, kde  $n$  je počet dále nerozložitelných (atomických) výroků (axiomů).
  - Když je v úloze 100 atomických výroků, tabulka bude mít  $2^{100}$  řádků.
  - V desítkové soustavě je  $2^{100}$  rovné číslu s počtem číslic  $1 + 100 \cdot \log_{10} 2 = 1 + 100 \cdot 0,3 = 31$ .
  - 1267650600228229401496703205376 řádků
- Proto se v praxi využívají inferenční systémy s vybranými inferenčními pravidly, které určité typy úloh řeší efektivně, ale neumí řešit vše.

# Konverze mezi Hornovou klauzulí a formou vhodnou pro rezoluci

- Hornova klauzule
  - vhodná pro použití zobecněného Modus ponens
  - $A \wedge B \wedge C \Rightarrow D =$
  - $= \neg(A \wedge B \wedge C) \vee D = \neg A \vee \neg B \vee \neg C \vee D$
  - Pokud je Hornova klauzule převedena do disjunkce literálů, tak v ní musí být maximálně jedna kladná atomická formule (hlava, zde  $D$ ).
    - Pokud tam je právě jedna, tak se nazývá **definite clause**.
    - Například formule  $A \wedge B \wedge C \Rightarrow D \vee E$  znamená, že, když platí  $A \wedge B \wedge C$ , tak se stane  $D$  nebo  $E$ , což není definitivní (jisté).
- Disjunkce literálů (je vhodná pro rezoluci)
  - $A \vee B \vee C \vee D =$
  - $= \neg(A \vee B \vee C) \Rightarrow D = (\neg A \wedge \neg B \wedge \neg C) \Rightarrow D$
  - Hlavou Hornovy klauzule nemusí být právě  $D$ .

# Příklad logicky správného úsudku

- $A$ : Klesá tlak.
- $B$ : Zkazí se počasí.
- $C$ : Když klesá tlak, zkazí se počasí.  $A \Rightarrow B$
- Dokažte, že platí  $B$ , když platí  $A$  a  $C$ .
- $A$  a  $C$  jsou předpoklady (axiomy).
- $B$  je závěr (dokazovaný teorém).
- $B$  je dokázáno,
  - když pro všechny řádky pravdivostní tabulky, kde jsou oba předpoklady pravdivé, platí, že i závěr je pravdivý, nebo
  - když je v pravdivostní tabulce sloupek s implikací, která má jako antecedent konjunkci všech axiomů a jako konkluzi dokazovaný závěr, a pro všechny kombinace hodnot všech atomických výroků je výsledek v tomto sloupci pravda.

| $A$      | $B$      | $C = (A \Rightarrow B)$ | $A \wedge C$ | $(A \wedge C) \Rightarrow B$ |
|----------|----------|-------------------------|--------------|------------------------------|
| 0        | 0        | 1                       | 0            | <b>1</b>                     |
| 0        | 1        | 1                       | 0            | <b>1</b>                     |
| 1        | 0        | 0                       | 0            | <b>1</b>                     |
| <b>1</b> | <b>1</b> | <b>1</b>                | 1            | <b>1</b>                     |

# Příklad logicky nesprávného úsudku

- $A$ : Klesá tlak.
- $B$ : Zkazí se počasí.
- $C$ : Když klesá tlak, zkazí se počasí.  $A \Rightarrow B$  (axiom)
- $D$ : Dnes neklesá tlak. (axiom)
- $E$ : Dnes se nezkazí počasí. (závěr, to jest dokazovaný teorém)
- Dokažte, že platí  $E$ , když platí  $C$  a  $D$ .
- $E$  nevyplývá z  $C$  a  $D$ , protože
  - na řádcích pravdivostní tabulky, kde jsou oba předpoklady (axiomy) pravdivé, není vždy závěr je pravdivý, a
  - sloupek s implikací, která má jako antecedent konjunkci všech axiomů a jako konkluzi dokazovaný závěr, nemá pro všechny kombinace hodnot všech atomických výroků výsledek pravda.

| $A$ | $B$ | $C = (A \Rightarrow B)$ | $D = \neg A$ | $E = \neg B$ | $C \wedge D$ | $(C \wedge D) \Rightarrow E$ |
|-----|-----|-------------------------|--------------|--------------|--------------|------------------------------|
| 0   | 0   | 1                       | 1            | 1            | 1            | 1                            |
| 0   | 1   | 1                       | 1            | 0            | 1            | 0                            |
| 1   | 0   | 0                       | 0            | 1            | 0            | 1                            |
| 1   | 1   | 1                       | 0            | 0            | 0            | 1                            |

# Predikátová logika 1. řádu

## (First-Order Predicate Calculus)

- Logika prvního řádu se tak jmenuje proto, že kvantifikuje objekty (entity prvního řádu pojmenovatelné podstatnými jmény) ale ne vztahy mezi těmito objekty nebo funkce těchto objektů.
- Oproti výrokové logice má navíc **kvantifikátory**.
  - existenční  $\exists$ 
    - Existuje pták, který létá.
  - univerzální  $\forall$ 
    - Všichni ptáci létají, neboli, když je někdo pták, tak létá.
- Logika vyššího řádu (higher-order logic) dovoluje kvantifikovat také vztahy a funkce.
  - Pomocí logiky vyššího řádu můžeme například říct, že dva objekty jsou stejné, když a právě když všechny jejich vlastnosti jsou stejné.
  - Logika vyššího řádu má větší vyjadřovací schopnost, ale pravděpodobně nelze vyvinout efektivní metody pro její používání.
- Predikát vyjadřuje vlastnost objektu nebo vztah mezi objekty.
- Predikáty tvoří **klauzule (clause)**.
  - Klauzule je disjunkce literálů.
  - **Literál** je atomická formule, což je atomický výrok vzniklý, když se za proměnnou dosadí konstanta.

# Inference v predikátové logice 1. řádu

- Za proměnné jsou dosazovány konstanty, čímž je jazyk predikátové logiky 1. řádu konvertován do jazyka výrokové logiky.
- Na výsledné výroky jsou uplatňována inferenční pravidla.
- Příklad:
  - Každý, kdo se učí a má štěstí, složí zkoušku.
    - $\forall X (učiSe(X) \wedge máŠtěstí(X) \Rightarrow složíZkoušku(X))$
  - Petr se učí. Petr má štěstí.
    - $učiSe(petr)$ .  $máŠtěstí(petr)$ .
    - Tyto axiomy jsou v konjunkci.
  - Složí Petr zkoušku?
    - $složíZkoušku(petr)$ .
- Řešení:
  - $učiSe(petr) \wedge máŠtěstí(petr) \Rightarrow složíZkoušku(petr)$ 
    - Za proměnnou  $X$  byla dosazena konstanta  $petr$ .
    - V bázi znalostí (axiomů) jsou všechny literály antecedentu, a tak lze pravidlem modus ponens vyvodit, že Petr složí zkoušku.

# Úlohy

- Úloha č. 1
  - Lze ji reprezentovat pomocí Hornových klauzulí.
  - Fakt, který se dokazuje, je nedokazatelný.
  - Když někdo studuje nebo má štěstí, tak složí zkoušky.
    - $(\text{studuje} \vee \text{štěstí}) \Rightarrow \text{složí} = (\text{studuje} \Rightarrow \text{složí}) \wedge (\text{štěstí} \Rightarrow \text{složí})$
  - $((\text{studuje} \Rightarrow \text{složí}) \wedge (\text{štěstí} \Rightarrow \text{složí}) \wedge \text{složí} \wedge \neg \text{studuje}) \Rightarrow \text{štěstí}$
- Úloha č. 2
  - Nelze ji reprezentovat pomocí Hornových klauzulí.
  - Fakt, který se dokazuje, je dokazatelný pomocí rezoluce.
  - Zkoušky složí jen ten, kdo studuje nebo má štěstí, neboli, když někdo složí zkoušky, znamená to, že studuje nebo má štěstí.
  - $((\text{složí} \Rightarrow (\text{studuje} \vee \text{štěstí})) \wedge \text{složí} \wedge \neg \text{studuje}) \Rightarrow \text{štěstí}$
- Z obou úloh chceme vyvodit zda, když Petr složil zkoušku a nestudoval, tak měl štěstí.
- $\text{složí} = A, \text{studuje} = B, \text{štěstí} = C$



# Úloha č. 1 řešená pomocí Hornových klauzulí

- 1.  $B \Rightarrow A$  (axiom)
- 2.  $C \Rightarrow A$  (axiom)
- 3.  $\neg B$  (axiom) nelze rezolvovat s 1. nebo 2.
- 4.  $A$  (axiom) nelze rezolvovat s 1. nebo 2.
- 5.  $C$  (závěr, to jest dokazovaný teorém) nelze odvodit
- Všechny hodnoty v posledním sloupci pravdivostní tabulky nemají hodnotu pravda.
- Zkoušku je možné složit i z jiných důvodů, než že někdo studuje nebo má štěstí.
  - Výsledek implikace je pravda, i když neplatí její antecedent.

| A        | B        | C        | $B \vee C$ | $D = (B \vee C \Rightarrow A)$ | $\neg B$ | $A \wedge D \wedge \neg B$ | $(A \wedge D \wedge \neg B) \Rightarrow C$ |
|----------|----------|----------|------------|--------------------------------|----------|----------------------------|--------------------------------------------|
| 0        | 0        | 0        | 0          | 1                              | 1        | 0                          | 1                                          |
| 0        | 0        | 1        | 1          | 0                              | 1        | 0                          | 1                                          |
| 0        | 1        | 0        | 1          | 0                              | 0        | 0                          | 1                                          |
| 0        | 1        | 1        | 1          | 0                              | 0        | 0                          | 1                                          |
| <b>1</b> | <b>0</b> | <b>0</b> | <b>0</b>   | <b>1</b>                       | <b>1</b> | <b>1</b>                   | <b>0</b>                                   |
| 1        | 0        | 1        | 1          | 1                              | 1        | 1                          | 1                                          |
| 1        | 1        | 0        | 1          | 1                              | 0        | 0                          | 1                                          |
| 1        | 1        | 1        | 1          | 1                              | 0        | 0                          | 1                                          |

# Úloha č. 2 řešená pomocí rezoluce

- 1.  $A \Rightarrow (B \vee C)$  (axiom)
- 2.  $\neg B$  (axiom)
- 3.  $A$  (axiom)
- 3.  $C$  (závěr, to jest dokazovaný teorém)
- Výrok 1. zkonvertovaný do disjunkce literálů  $\neg A \vee B \vee C$  nemá maximálně jednu kladnou atomickou formuli a proto nelze převést do formy Hornových klauzulí.
- $\neg A \vee B \vee C$  rezolvujeme s  $\neg B$  a dostaneme axiom  $\neg A \vee C$ .
- $\neg A \vee C$  rezolvujeme s  $A$  a dostaneme axiom  $C$ , který jsme chtěli dokázat.
- Je možný i postup začínající tím, že  $\neg A \vee B \vee C$  rezolvujeme s  $A$  a dostaneme axiom  $B \vee C$ , který potom rezolvujeme s  $\neg B$  a dostaneme také axiom  $C$ .
- **Postup řešení lze reprezentovat grafem typu strom.**
- Všechny hodnoty v posledním sloupci pravdivostní tabulky mají hodnotu pravda.

| A | B | C | $B \vee C$ | $D = (A \Rightarrow B \vee C)$ | $\neg B$ | $A \wedge D \wedge \neg B$ | $(A \wedge D \wedge \neg B) \Rightarrow C$ |
|---|---|---|------------|--------------------------------|----------|----------------------------|--------------------------------------------|
| 0 | 0 | 0 | 0          | 1                              | 1        | 0                          | <b>1</b>                                   |
| 0 | 0 | 1 | 1          | 1                              | 1        | 0                          | <b>1</b>                                   |
| 0 | 1 | 0 | 1          | 1                              | 0        | 0                          | <b>1</b>                                   |
| 0 | 1 | 1 | 1          | 1                              | 0        | 0                          | <b>1</b>                                   |
| 1 | 0 | 0 | 0          | 0                              | 1        | 0                          | <b>1</b>                                   |
| 1 | 0 | 1 | 1          | 1                              | 1        | 1                          | <b>1</b>                                   |
| 1 | 1 | 0 | 1          | 1                              | 0        | 0                          | <b>1</b>                                   |
| 1 | 1 | 1 | 1          | 1                              | 0        | 0                          | <b>1</b>                                   |

# Deklarativní programovací jazyky

- Prolog patří mezi takzvané deklarativní programovací jazyky.
- Deklarativnost spočívá v tom, že je nejdříve vytvořen univerzální program pro řešení problému (inferenční mechanismus). V případě Prologu je to program pro rezolvování, který je druhem překladače pro uživatelem zadanou úlohu.
- Uživatel už potom pouze popíše problém způsobem kompatibilním s oním univerzálním programem, ale nestanovuje algoritmus jeho řešení.
  - V případě Prologu deklaruje
    - fakta
      - mensi(jan,zdenek).
      - mensi(jan,petr).
      - mensi(petr,zdenek).
    - pravidla
      - nejmensi(X) :- mensi(X,Y) and mensi(Y,Z).
    - otázky
      - ?nejmensi(Who).
  - Inferenční mechanismus Prologu odpoví na otázky:
    - nejmensi(jan) yes
- Vzhledem k tomu, že algoritmus překladače Prologu, což je vlastní inferenční mechanismus, je fixní, tak v praxi záleží na pořadí, v jakém jsou fakta a implikace zadány.

# Co je třeba umět do testu

- Převod věty z přirozeného jazyka do jazyka predikátové logiky 1. řádu
- Úprava výroků do formy klauzulí
- Provedení důkazu z axiomů a inferenčního pravidla

# Převod věty z přirozeného jazyka do jazyka predikátové logiky 1. řádu (1)

- Ne všichni jsou zároveň bohatí a šťastní.
- Žádný člověk není zároveň bohatý a šťastný.
- Chytrá lama nejí rostlinu, která je pro ni jedovatá.
- Existuje lama, která má ráda všechny rostliny, které nejsou trnité.
- Hedvika má ráda pouze rostliny, které nejsou trnité.
- Hedvika má ráda všechny rostliny, které nejsou trnité.
- Pouze Evženovi kamarádi znají Evženovo tajemství.
- Všichni Evženovi kamarádi znají Evženovo tajemství.
- Nikdo si nevážší studovaného podvodníka.
- Nikdo si nenajme právníka, který nevyhrál ani jeden spor.
- Nikdo si nekoupí dům, pokud tento dům není zkolaudovaný.
- Existují domy, které si nikdo nekoupí, i když jsou zkolaudované.
- Každý člověk přijme práci, pokud je dobře placená.
- Existují práce, které někteří lidé nepřijmou, i když jsou dobře placené.
- Všichni mají rádi napínavé příběhy. = Každý má rád napínavý příběh.
- Dobré prase všechno spase.
- Brána má smysl, jen když se neotevře všem.
- Evžen jí všechno, co vaří maminka, ale jen něco z toho, co vaří závodka.
- Ten, kdo vyřeší alespoň jeden příklad / všechny příklady, je chytrý.

# Převod věty z přirozeného jazyka do jazyka predikátové logiky 1. řádu (2)

- Ne každé auto se spalovacím motorem má manuální převodovku.
- Každé auto se spalovacím motorem potřebuje benzin nebo naftu.
- Každé auto, které jezdí po silnicích, má homologaci.
- Ne každé auto na této výstavě má homologaci.
- Nikdo si nekoupí auto bez homologace.
- Ne každá součástka auta, které se vyrobilo v ČR, byla vyrobena v ČR.
- Jana umí jezdit jen tam, kde není náledí.
- Jana umí jezdit všude tam, kde není náledí.
- Jana umí v řízení auta vše až na smyk.

# Úprava výroků do formy klauzulí

- Když nějaký předmět studuje Jan, tak jej nestuduje Petr, a naopak.
- $A \oplus B = \neg(A \Leftrightarrow B) = \neg((A \Rightarrow B) \wedge (B \Rightarrow A)) =$
- $= \neg((\neg A \vee B) \wedge (\neg B \vee A)) =$
- $= \neg(\neg A \vee B) \vee \neg(\neg B \vee A) =$
- $= (A \wedge \neg B) \vee (B \wedge \neg A) =$
- $= ((A \wedge \neg B) \vee B) \wedge ((A \wedge \neg B) \vee \neg A) =$
- $= (A \vee B) \wedge (\neg B \vee B) \wedge (A \vee \neg A) \wedge (\neg B \vee \neg A) =$
- $= (A \vee B) \wedge (\neg B \vee \neg A)$
- $(\neg B \vee B) \wedge (A \vee \neg A)$  jsou tautologie a ty se škrtají.

# Provedení důkazu z axiomů a inferenčního pravidla

- 1.  $P \Rightarrow Q$                        $\neg P \vee Q$
- 2.  $\neg P \Rightarrow R$      $P \vee R$  je nepřevoditelný do Hornovy formy.
- 3.  $Q \Rightarrow S$                        $\neg Q \vee S$
- 4.  $R \Rightarrow S$                        $\neg R \vee S$
- 5.  $S$  (teorém, který máme dokázat z předchozích axiomů)  
přidáme mezi axiomy jako jeho negaci  $\neg S$ , což se nazývá  
**metoda rezolučního zamítnutí** (proof by refutation).
- 6. Resolventa výroků 1. a 2. je  $Q \vee R$ .
- 7. Resolventa výroků 6. a 3. je  $R \vee S$ .
- 8. Resolventa výroků 7. a 4. je  $S$ .
- 9. Resolventa výroků 8.  $S$  a 5.  $\neg S$  je prázdná klauzule značící  
rozpor.
- Je-li negovaný teorém v rozporu s axiomy, znamená to, že je  
dokázán.
- Je možné rezolvovat v různém pořadí – **strom možností**.



# Provedení důkazu z axiomů a inferenčního pravidla

- 1.  $A \vee B \vee C$
- 2.  $\neg A \vee \neg B \vee D$
- 3.  $C \vee D$
- Je 3. teorém vzhledem k axiomům 1. a 2.?
- Resolventa výroků 1. a 2. je  $B \vee C \vee \neg B \vee D$ .
- Vzhledem k tomu, že obsahuje atomy  $B$  a  $\neg B$ , tak je to tautologie a ta se vyřazuje z množiny vygenerovaných axiomů.
- Víc axiomů vygenerovat nelze, takže teorém 3. z axiomů 1. a 2. nevyplývá.
  - Viz řádek z pravdivostní tabulky, který to ukazuje:

| A | B | C | $\neg A$ | $\neg B$ | D | $C \vee D$ |
|---|---|---|----------|----------|---|------------|
| 1 | 0 | 0 | 0        | 1        | 0 | 0          |