Úvod do TI - logika
Výroková logika - pokračování (3.přednáška)

Marie Duží
marie.duzi@vsb.cz

Normální formy formulí výrokové logiky

- Každé formuli výrokové logiky přísluší právě jedna pravdivostní funkce, zobrazení $\{p, q, r\ldots\} \rightarrow \{0, 1\}$ (pravdivostní tabulka).

- Naopak však jedné takové funkci odpovídá ***nekonečně mnoho formulí***, které jsou navzájem *ekvivalentní.*

Definice: Formule A, B jsou ***ekvivalentní***, značíme $A \Leftrightarrow B$, mají-li přesně stejné modely, tj. vyjadřují stejnou pravdivostní funkci. Jinými slovy, **$A \Leftrightarrow B$ iff $A \models B$ a $B \models A$**.

Příklad: $p \supset q \Leftrightarrow \neg p \vee q \Leftrightarrow \neg(p \wedge \neg q) \Leftrightarrow$
$(p \supset q) \wedge (p \vee \neg p) \Leftrightarrow (p \supset q) \vee (p \wedge \neg p) \Leftrightarrow \ldots$

Pozn.: Nesmíme plést ekvivalenci formulí $A \Leftrightarrow B$ s formulí tvaru ekvivalence $A \equiv B$.
Platí však, že $A \Leftrightarrow B$ právě když formule $A \equiv B$ je tautologie.

Př.: $(p \equiv q) \Leftrightarrow [(p \supset q) \wedge (q \supset p)]$ iff
$\models [(p \equiv q) \equiv ((p \supset q) \wedge (q \supset p))]$

Normální formy formulí výrokové logiky

$(p \equiv q) \Leftrightarrow [(p \supset q) \wedge (q \supset p)] \Leftrightarrow$

$\mathbf{[(\neg p \vee q) \wedge (\neg q \vee p)]} \Leftrightarrow$

$\mathbf{[(p \wedge q) \vee (\neg p \wedge \neg q)]} \Leftrightarrow \ldots.$

- Je užitečné stanovit nějaký ***normální tvar*** formule – tj. vybrat mezi těmito nekonečně mnoha ekvivalentními formulemi jeden nebo dva *kanonické normální tvary*. ***Třída ekvivalentních formulí je pak reprezentována*** touto vybranou formulí v normálním tvaru.

- V našem příkladu jsou v normálním tvaru formule na druhém a třetím řádku.

Normální formy formulí výrokové logiky

Literál je výrokový symbol nebo jeho negace. Př.: $\mathbf{p}, \neg\mathbf{q}, \mathbf{r}, \ldots$

Elementární konjunkce (**EK)** je konjunkce literálů. Př.: $\mathbf{p} \wedge \neg\mathbf{q}, \mathbf{r} \wedge \neg\mathbf{r}, \ldots$

Elementární disjunkce (**ED)** je disjunkce literálů. Př.: $\mathbf{p} \vee \neg\mathbf{q}, \mathbf{r} \vee \neg\mathbf{r}, \ldots$

Úplná elementární konjunkce (**ÚEK)** dané množiny výrokových symbolů je elementární konjunkce, ve které se každý symbol z dané množiny vyskytuje právě jednou (buďto prostě nebo negovaný): Př.: $\mathbf{p} \wedge \neg\mathbf{q}$

Úplná elementární disjunkce (**ÚED**) dané množiny výrokových symbolů je elementární disjunkce, ve které se každý symbol z dané množiny vyskytuje právě jednou (buďto prostě nebo negovaný). Př.: $\mathbf{p} \vee \neg\mathbf{q}$

Disjunktivní normální forma (**DNF**) dané formule je formule ekvivalentní s danou formulí a mající tvar disjunkce elementárních konjunkcí. *Příklad*: DNF$(\mathbf{p} \equiv \mathbf{p})$: $(\mathbf{p} \wedge \mathbf{p}) \vee (\neg\mathbf{p} \wedge \neg\mathbf{p}), \mathbf{p} \vee \neg\mathbf{p}$

Konjunktivní normální forma (**KNF**) dané formule je formule ekvivalentní s danou formulí a mající tvar konjunkce elementárních disjunkcí. *Příklad*: KNF$(\mathbf{p} \equiv \mathbf{p})$: $(\neg\mathbf{p} \vee \mathbf{p}) \wedge (\neg\mathbf{p} \vee \mathbf{p})$

Úplná disjunktivní normální forma (**UDNF**) dané formule je formule ekvivalentní s danou formulí a mající tvar disjunkce úplných elementárních konjunkcí. *Příklad*: UDNF$(\mathbf{p} \equiv \mathbf{q})$: $(\mathbf{p} \wedge \mathbf{q}) \vee (\neg\mathbf{p} \wedge \neg\mathbf{q})$

Úplná konjunktivní normální forma (**UKNF**) dané formule je formule ekvivalentní s danou formulí a mající tvar konjunkce úplných elementárních disjunkcí. *Příklad*: UKNF$(\mathbf{p} \equiv \mathbf{q})$: $(\neg\mathbf{p} \vee \mathbf{q}) \wedge (\neg\mathbf{q} \vee \mathbf{p})$

Normální formy formulí výrokové logiky

Jak nalézt kanonický tvar (tj. UDNF, UKNF) formule?

- **UDNF**: disjunkce = 1, když alespoň jedna UEK = 1, tj. všechny literály v této UEK = 1.
- **UKNF**: konjunkce = 0, když alespoň jedna UED = 0, tj. všechny literály v této UED = 0.

Proto: UDNF (UKNF) sestrojíme z pravdivostní funkce tak, že si všímáme řádků, kde je hodnota 1 (0) a *„zajišťujeme správnou hodnotu literálů“* – 1 (0).

Nalezení UDNF, UKNF - tabulkou

Nalézt UDNF, UKNF formule: $\neg(p \supset q)$

p	q	$\neg(p \supset q)$	UEK	UED
1	1	0		$\neg p \vee \neg q$
1	0	1	$p \wedge \neg q$	
0	1	0		$p \vee \neg q$
0	0	0		$p \vee q$

UDNF: $\mathbf{p \wedge \neg q}$

UKNF: $\mathbf{(\neg p \vee \neg q) \wedge (p \vee \neg q) \wedge (p \vee q)}$

Nalezení UDNF, UKNF - úpravami

Metoda ekvivalentních úprav:

$\neg(\mathbf{p} \supset \mathbf{q}) \Leftrightarrow \neg(\neg\mathbf{p} \vee \mathbf{q}) \Leftrightarrow (\mathbf{p} \wedge \neg\mathbf{q})$ UDNF $\Leftrightarrow$
$[\mathbf{p} \vee (\mathbf{q} \wedge \neg\mathbf{q})] \wedge [\neg\mathbf{q} \vee (\mathbf{p} \wedge \neg\mathbf{p})] \Leftrightarrow$
$\Leftrightarrow (\mathbf{p} \vee \mathbf{q}) \wedge (\mathbf{p} \vee \neg\mathbf{q}) \wedge (\neg\mathbf{p} \vee \neg\mathbf{q})$ UKNF

Pozn.: Využíváme zde tautologie výrokové logiky, viz předchozí presentace.

Ve druhém řádku využíváme toho, že disjunkce libovolné formule A s kontradikcí (F) je ekvivalentní A: $A \vee F \Leftrightarrow A$

Ve třetím řádku jsou použity distributivní zákony.

Každá formule, která není kontradikce, má UDNF a
Každá formule, která není tautologie, má UKNF

Opačná úloha: k UDNF, UKNF nalézt jednodušší „původní“ formuli

- Alchymista je zavřen ve vězení a dostane 5 motáků s výroky:
  - p: Podaří se ti přeměna olova ve zlato
  - q: 1.4. bude tvůj švagr jmenován prokurátorem
  - r: Po 1.4. bude soud.

První moták zní: $\mathbf{p \wedge q \wedge r}$

Druhý moták zní: $\mathbf{p \wedge q \wedge \neg r}$

Třetí moták zní: $\mathbf{\neg p \wedge \neg q \wedge r}$

Čtvrtý moták zní: $\mathbf{\neg p \wedge \neg q \wedge \neg r}$

Pátý moták zní: Alespoň jeden z předchozích motáků je pravdivý.

  - Otázka: Co se vlastně nebohý alchymista dověděl?

- *Řešení*: $\mathbf{(p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge \neg q \wedge \neg r)}$. Máme tedy nalézt formuli, k níž je tato UDNF ekvivalentní. Za pomoci distributivních zákonů dostaneme:

  $\mathbf{(p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge \neg q \wedge \neg r) \Leftrightarrow}$

  $\mathbf{(p \wedge q) \wedge (r \vee \neg r) \vee (\neg p \wedge \neg q) \wedge (r \vee \neg r) \Leftrightarrow (p \wedge q) \vee (\neg p \wedge \neg q) \Leftrightarrow (p \equiv q)}$

- *Odpověď*: Podaří se ti přeměna olova ve zlato tehdy a jen tehdy, když bude 1.4. tvůj švagr jmenován prokurátorem.

Otázka: kolik binárních pravdivostních funkcí (a tedy logických spojek) existuje?

p	*q*	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
			$\wedge$						$\vee$	$\downarrow$	$\equiv$				$\supset$	$\uparrow$	
										NOR						**NAND**	

Kolik nejméně a které spojky potřebujeme?

- Dle věty o normálních tvarech stačí: **¬, ∨, ∧ (funkcionálně úplná soustava)**
- Ostatní vytvoříme skládáním funkcí

Následující soustavy pravdivostních funkcí jsou funkcionálně úplné:

1. pravdivostní funkce příslušející spojkám **{¬, ∧, ∨}**,
2. pravdivostní funkce příslušející spojkám **{¬, ∧}** nebo **{¬, ∨}**,
3. pravdivostní funkce příslušející spojkám **{¬, ⊃},**
4. pravdivostní funkce příslušející spojce **{↑}** nebo **{↓}**.

Tedy k vyjádření libovolné pravdivostní funkce, tj. libovolné formule ekvivalentním způsobem stačí jedna spojka!

Buď Schefferova NAND ↑ nebo Pierceova NOR ↓.

Kolik nejméně a které spojky potřebujeme?

- Soustava $\{\neg, \wedge, \vee\}$ stačí dle vět o normálních formách.
- Převod na soustavu $\{\neg, \wedge\}$ nebo $\{\neg, \vee\}$:

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

- Převod na soustavu $\{\neg, \supset\}$:

$$\mathbf{A \vee B \Leftrightarrow \neg A \supset B,}$$
$$\mathbf{A \wedge B \Leftrightarrow \neg(A \supset \neg B)}$$

- Převod na soustavu $\{\uparrow\}$ nebo $\{\downarrow\}$:

$\mathbf{\neg A \Leftrightarrow A{\uparrow}A,\ A{\wedge}B \Leftrightarrow (A{\uparrow}B){\uparrow}(A{\uparrow}B),}$ kde $\uparrow$ značí NAND,
$\mathbf{\neg A \Leftrightarrow A{\downarrow}A,\ A{\vee}B \Leftrightarrow (A{\downarrow}B){\downarrow}(A{\downarrow}B),}$ kde $\downarrow$ značí NOR.

Sémantická tabla: metoda tvorby DNF, KNF

- Disjunktivní tablo: strom jehož listy jsou konjunkce literálů.

- Konjunktivní tablo: strom jehož listy jsou disjunkce literálů.

- $\models$ A (tautologie) $\Rightarrow$ všechny listy ***konjunktivního*** *tabla* musí být uzavřené, tj. obsahovat opačný pár literálů: p, $\neg$p: **(p $\vee$ $\neg$p)** – ***tautologie***

- A $\models$ (kontradikce) $\Rightarrow$ všechny listy ***disjunktivního*** *tabla* musí být uzavřené, tj. obsahovat opačný pár literálů: p, $\neg$p: **(p $\wedge$ $\neg$p)** – ***kontradikce***

Postup tvorby sémantického tabla

1. Formuli upravíme tak, aby negace byla všude „uvnitř“, tj. u jednotlivých proměnných (tedy provedeme ekvivalentní negace).

2. Tam, kde je nějaká podformule ve tvaru implikace nebo ekvivalence, převedeme na disjunkci resp. konjunkci s využitím logických zákonů:

$$(p \supset q) \Leftrightarrow (\neg p \vee q),$$
$$(p \equiv q) \Leftrightarrow (\neg p \vee q) \wedge (\neg q \vee p) \Leftrightarrow (p \wedge q) \vee (\neg p \wedge \neg q)$$

3. Konstruujeme tablo uplatňováním distributivního zákona.

Disjunktivní normální forma

- Větvení znamená disjunkci, čárka znamená konjunkci.
- Postupujeme zleva a jakmile narazíme na disjunkci, rozvětvíme. Levý disjunkt do levé větve + zbytek formule oddělený čárkami, pravý disjunkt do pravé větve + zbytek.
- Větvíme tak dlouho, až žádná podformule neobsahuje disjunkci a strom má v listech pouze seznamy literálů oddělené čárkami, které značí *elementární konjunkce*.

Konjunktivní normální forma:
duální k disjunktivní

- Větvení znamená konjunkci, čárka znamená disjunkci.

- Postupujeme zleva a jakmile narazíme na konjunkci, rozvětvíme. Levý konjunkt do levé větve + zbytek formule oddělený čárkami, pravý do pravé + zbytek.

- Větvíme tak dlouho, až není nikde konjunkce a strom má v listech pouze seznamy literálů oddělené čárkami, které značí *elementární disjunkce*.

Důkaz, že formule je a) tautologie, b) kontradikce

a) **Důkaz,** že formule **F je kontradikce:**

Zkonstruujeme ***disjunktivní*** **tablo**. Pokud se všechny větve uzavřely, tj. každý list sémantického stromu tvoří elementární konjunkce s dvojicí literálů s opačným znaménkem (např. p, ¬p, což znamená p ∧ ¬p), je formule F kontradikce.

b) **Důkaz,** že formule **F je tautologie:**

Zkonstruujeme ***konjunktivní*** **tablo**. Pokud se všechny větve uzavřely, tj. každý list sémantického stromu tvoří elementární disjunkce s dvojicí literálů s opačným znaménkem (např. p, ¬p, což znamená p ∨ ¬p), je formule F tautologie.

Důkaz, co vyplývá z dané formule

- Disjunktivní sémantické tablo, které se neuzavřelo, doplníme takovou formulí (konjunkcí literálů), aby se *všechny* větve uzavřely.
- *Negace* toho, co jsme doplnili, vyplývá**.** Např. doplníme-li **p, ¬q**, pak jsme doplnili $\mathbf{p\wedge\neg q}$, tedy vyplývá $\mathbf{\neg p\vee q}$, tj. $\mathbf{p\supset q}$.
- Duálním způsobem můžeme použít konjunktivní normální formu.

Úplné normální formy.

- Z tabla se dá snadno zdůvodnit, proč tautologie nemá úplnou konjunktivní normální formu (UKNF) a proč kontradikce nemá úplnou disjunktivní normální formu (UDNF):

- Je-li formule tautologie, pak se všechny větve konjunktivního tabla uzavřou, tj. v každé ED je dvojice literálů s opačným znaménkem (p, ¬p), resp. p ∨ ¬p, což je vyloučeno v UED. Tedy neexistuje UKNF.

- Analogicky pro UDNF.

UDNF tautologie
o jedné (T1), dvou (T2) a třech (T3) proměnných

T1: $\mathbf{p \vee \neg p}$

T2: $\mathbf{(p \wedge q) \vee (p \wedge \neg q) \vee (\neg p \wedge q) \vee (\neg p \wedge \neg q)}$

T3: $\mathbf{(p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge r) \vee}$
$\mathbf{(p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \vee}$
$\mathbf{(\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge \neg q \wedge \neg r)}$

V každé možné valuaci musí být alespoň jedna UEK pravdivá.

UDNF tautologie

Není-li formule tautologie, pak musí v UDNF nějaká z těchto UEK chybět.

Proto, konstruujeme-li **UDNF z tabulky,** vyjdeme z těch řádků pravdivostní funkce, ve kterých je **hodnota = 1** (hodnoty nepravda v disjunkci nehrají roli, neboť A $\vee$ F $\Leftrightarrow$ A) a konstruujeme UEK tak, aby byla při této valuaci pravdivá.

Zcela analogicky pro **UKNF – vyjdeme z 0.**

Dokažte, že formule F je tautologie:

F: $[(p \supset (q \lor r)) \land (\neg s \supset \neg q) \land (t \lor \neg r)] \supset (p \supset (s \lor t))$

Nepřímý *důkaz* (sporem). Pomocí ***disjunktivní*** normální formy dokážeme, že formule $\neg F$ je ***kontradikce:***

$$(p \supset (q \lor r)) \land (\neg s \supset \neg q) \land (t \lor \neg r) \land p \land \neg s \land \neg t \Leftrightarrow$$

$$(\neg p \lor q \lor r) \land (s \lor \neg q) \land (t \lor \neg r) \land p \land \neg s \land \neg t$$

Konstrukce disjunktivního tabla (větvení – disjunkce, čárka – konjunkce):

Dokažte, že formule F je tautologie:

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

$\neg p, (s \vee \neg q), (t \vee \neg r), p, \neg s, \neg t$ $\quad$ $q, (s \vee \neg q), (t \vee \neg r), p, \neg s, \neg t$

$+$ $\quad$ $r, (s \vee \neg q), (t \vee \neg r), p, \neg s, \neg t$

$q, s, (t \vee \neg r), p, \neg s, \neg t$ $\quad$ $q, \neg q, (t \vee \neg r), p, \neg s, \neg t$

$+$ $\quad$ $+$

$r, s, (t \vee \neg r), p, \neg s, \neg t$ $\quad$ $r, \neg q, (t \vee \neg r), p, \neg s, \neg t$

$+$

$r, \neg q, t, p, \neg s, \neg t$ $\quad$ $r, \neg q, \neg r, p, \neg s, \neg t$

$+$ $\quad$ $+$

Co vyplývá z formule G ?

G: **$(p \supset (q \vee r)) \wedge (\neg s \supset \neg q) \wedge (t \vee \neg r)$**

Řešení: Kdybychom udělali sémantické tablo formule G, pak ve všech listech předchozího sémantické stromu budou chybět literály **p, $\neg s$, $\neg t$**.

Doplníme-li je tedy do všech větví, všechny větve se uzavřou. Tedy formule ($G \wedge p \wedge \neg s \wedge \neg t$) je kontradikce.

Proto: $\models G \supset \neg(p \wedge \neg s \wedge \neg t)$.

Tedy: $G \models \neg(p \wedge \neg s \wedge \neg t)$,

$G \models (\neg p \vee s \vee t)$,

$G \models (p \supset (s \vee t))$

Není divu, vždyť formule F (sl. 22) je tautologie !

Dokažte, že formule F je tautologie:

F: $[(p \supset (q \vee r)) \wedge (\neg s \supset \neg q) \wedge (t \vee \neg r)] \supset (p \supset (s \vee t))$

Přímý *důkaz:* Pomocí ***konjunktivní*** normální formy dokážeme, že formule F je ***tautologie:***

Větvení – konjunkce, čárka – disjunkce.

$$F \Leftrightarrow (p \wedge \neg q \wedge \neg r) \vee (\neg s \wedge q) \vee (\neg t \wedge r) \vee \neg p \vee s \vee t$$

Atd., analogicky (duálně) k nepřímému důkazu disjunktivní normální formou.

Rezoluční metoda ve výrokové logice

- Sémantické tablo není výhodné z praktických důvodů.
- Chceme-li dokázat, že $\mathbf{P_1,\ldots,P_n} \models \mathbf{Z}$, stačí dokázat, že $\models \mathbf{(P_1 \land \ldots \land P_n) \supset Z}$, tj. $\mathbf{P_1 \land \ldots \land P_n \land \neg Z}$ je kontradikce.
- Ale, pro důkaz sémantickým tablem potřebujeme formuli v ***disjunktivní*** normální formě. To znamená, že musíme převádět tuto formuli, která je téměř v konjunktivní formě, za použití distributivního zákona do disjunktivní formy.
- Použití sémantického tabla vede často k mnoha distributivním krokům.
- Jednodušší prakticky bude dokázat přímo spornost, tj. ***nesplnitelnost,*** formule $P_1 \land \ldots \land P_n \land \neg Z$.

Rezoluční pravidlo dokazování ve výrokové logice

Nechť *l* je literál. Z formule (A ∨ *l*) ∧ (B ∨ ¬*l*) odvoď (A ∨ B). Zapisujeme:

$$\frac{(A \lor l) \land (B \lor \neg l)}{(A \lor B)}$$

Toto pravidlo není přechodem k ekvivalentní formuli, ale zachovává **splnitelnost.**

Důkaz: Nechť je formule **(A ∨ *l*) ∧ (B ∨ ¬*l*)** splnitelná, tedy pravdivá při nějaké valuaci *v*. Pak při této valuaci musí být pravdivé oba disjunkty (tzv. ***klausule***):

(A ∨ *l*) a (B ∨ ¬*l*).

Nechť je dále ***v*(*l*) = 0**. Pak ***w*(A) = 1** a tedy ***w*(A ∨ B) = 1**.
Nechť je naopak ***v*(*l*) = 1**. Pak ***w*(¬*l*) = 0** a musí být ***w*(B) = 1**, a tedy ***w*(A ∨ B) = 1**.

V obou případech je tedy formule **A ∨ B** ***pravdivá v modelu v*** původní formule, a tedy ***splnitelná***.

Rezoluční pravidlo dokazování ve výrokové logice

Uvědomme si, že důkaz byl proveden pro jakýkoli model, tj. valuaci *v*.

Jinými slovy platí, že pravidlo ***zachovává také pravdivost***:

$$\mathbf{(A \lor \mathit{l}) \land (B \lor \neg \mathit{l}) \models (A \lor B).}$$

To nám poskytuje návod, jak řešit úlohu, **co vyplývá z** dané formule, resp. množiny formulí.

Navíc, platí že ***konjunktivní*** **rozšíření** formule o ***rezolventu*** **(A ∨ B)** nemění pravdivostní funkci formule:

$$\mathbf{(A \lor \mathit{l}) \land (B \lor \neg \mathit{l}) \Leftrightarrow (A \lor \mathit{l}) \land (B \lor \neg \mathit{l}) \land (A \lor B)}$$

Klauzulární forma

- Konjunktivní normální forma formule se v rezoluční metodě nazývá ***klauzulární forma.*** Jednotlivé konjunkty (tj. elementární disjunkce) se nazývají ***klauzule.***

- Příklad: *Převod formule do klauzulární formy*:

¬[((p ⊃ q) ∧ (r ∨ ¬q) ∧ ¬r) ⊃ ¬p] ⇔
((p ⊃ q) ∧ (r ∨ ¬q) ∧ ¬r) ∧ p ⇔
(¬p ∨ q) ∧ (r ∨ ¬q) ∧ ¬r ∧ p

- Formule má čtyři klauzule.

  Důkaz, že je to kontradikce provedeme tak, že postupně přidáváme rezolventy, až dojdeme ke sporné klauzuli (¬p ∧ p), značíme: $\square$

- (¬p ∨ q) ∧ (r ∨ ¬q) ∧ **(¬p ∨ r)** ∧ ¬r ∧ **¬p** ∧ p

Důkazy rezoluční metodou

- **R(F) – *konjunktivní* rozšíření formule F o všechny rezolventy.**
- **$R_0(F)$ = F, $R_i(F) = R(R_{i-1}(F))$ – rezoluční uzávěr formule F. Platí, že $R_i(F) \Leftrightarrow F$**
- Důkaz, že **A je kontradikce** (nesplnitelná): existuje *n* takové, že $R_n(A)$ obsahuje prázdnou klausuli:
- Důkaz, že **A je tautologie**: **¬A** je kontradikce.
- Důkaz, že **množina formulí $\{A_1,\ldots,A_n\}$ je nesplnitelná**: dokážeme, že formule $A_1 \wedge \ldots \wedge A_n$ je kontradikce.
- Odvodit, co vyplývá z $\{A_1,\ldots,A_n\}$ znamená odvodit všechny rezolventy.
- Důkaz správnosti úsudku $A_1,\ldots,A_n \models Z$
  - **Přímý**: postupným vytvářením rezolvent odvodíme, že z $A_1,\ldots,A_n$ vyplývá Z
  - **Nepřímý**: dokážeme, že $A_1 \wedge \ldots \wedge A_n \supset Z$ je tautologie, neboli $A_1 \wedge \ldots \wedge A_n \wedge \neg Z$ je kontradikce, neboli množina $\{A_1, \ldots, A_n, \neg Z\}$ je nesplnitelná.

Příklady, postup

- Jednotlivé klausule napíšeme pod sebe a odvozujeme rezolventy (které vyplývají). Při nepřímém důkazu se snažíme dojít k prázdné klauzuli.

- Důkaz nesplnitelnosti formule:

$$(\neg \mathbf{q} \supset \mathbf{p}) \wedge (\mathbf{p} \vee \mathbf{r}) \wedge (\mathbf{q} \supset \neg \mathbf{r}) \wedge \neg \mathbf{p}$$

1. $q \vee p$
2. $p \vee r$
3. $\neg q \vee \neg r$
4. $\neg p$
5. $p \vee \neg r$ rezoluce 1, 3
6. p rezoluce 2, 5
7. spor 4, 6

Dotaz: Co vyplývá z formule $(\neg \mathbf{q} \supset \mathbf{p}) \wedge (\mathbf{p} \vee \mathbf{r}) \wedge (\mathbf{q} \supset \neg \mathbf{r})$?
Formule **p**.

Příklady, postup

- **Přímý** důkaz platnosti úsudku:

$p \supset q \vee r, \neg s \supset \neg q, t \vee \neg r \models p \supset (s \vee t)$

1. $\neg p \vee q \vee r$
2. $s \vee \neg q$
3. $t \vee \neg r$
4. $\neg p \vee s \vee r$ rezoluce 1, 2
5. $\neg p \vee s \vee t$ rezoluce 3, 4
6. $\neg p \vee s \vee t \Leftrightarrow p \supset (s \vee t)$ **Q.E.D.**

Příklady

- **Nepřímý** důkaz platnosti úsudku:

$$\mathbf{p \supset q \lor r, \neg s \supset \neg q, t \lor \neg r \models p \supset (s \lor t)}$$

1. $\neg p \lor q \lor r$
2. $s \lor \neg q$
3. $t \lor \neg r$
4. $\neg p \lor s \lor r$ rezoluce 1, 2
5. $\neg p \lor s \lor t$ rezoluce 3, 4
6. p
7. $\neg s$
8. $\neg t$
9. $\neg p \lor s \lor t, p, \neg s, \neg t \models s \lor t, \neg s, \neg t \models t, \neg t \models$ ■

Negovaný
Závěr
(řádky 6.,7.,8.)

Logické programování

- Strategie generování rezolvent.
- Rezoluční uzávěr – generujeme ***všechny*** rezolventy – strategie generování do ***šířky***
  - Může být implementačně neefektivní – kombinatorická exploze rezolvent
- Proto v logickém programování se používá strategie generování ***do hloubky***: není úplná – může uvíznout v nekonečné větvi, i když řešení existuje.

Příklad neúplnosti strategie ***do hloubky***

$$d \wedge e \wedge (b \supset a) \wedge (e \supset b) \wedge (d \supset a) \models a \ (?)$$

- Logický program (strategie do hloubky, řízená cílem):

1. D. *fakt*
2. E. *fakt*
3. A :– B. *pravidlo (A pokud B)*
4. B :– E. *pravidlo (B pokud E)*
5. A :– D. *pravidlo (A pokud D)*
6. ?A *dotaz, cíl*

– snaží se splnit cíl A, prohledává program ***shora dolů*** a hledá klauzuli s A vlevo – najde klauzuli 3. – generuje nový cíl B – najde klauzuli 4. – nový cíl E – je splněn klauzulí 2. – ***ANO.***

– ***nebo*** *generuje* ***proces navracení: znovu ?A*** – klauzule 5 – cíl D – klauzule
1. – ***ANO.***

Příklad neúplnosti strategie ***do hloubky, zleva***

Malá úprava programu:

1. D.
2. E **:– B.**
3. A :– B.
4. B :– E.
5. A :– D.
6. ?A

A

3. 5.

B D

4. 1.

E

2.

B

4.

...atd.

Řešení nenajde, i když existuje.

Výroková logika – shrnutí úloh

1. Důkaz, že daná formule je tautologie / kontradikce, nesplnitelná.
2. Důkaz, že daná množina formulí je nesplnitelná.
3. Důkaz (přímý / nepřímý), že úsudek je platný.
4. Odvození, co vyplývá z daných předpokladů.

- Umíme řešit:
  - Sémanticky, tj. tabulkou nebo sporem.
  - Úpravami formulí a použitím základních pravidel odvozování (modus ponens, transpozice, eliminace disjunkce).
  - Pomocí normálních forem: konjunktivní a disjunktivní, tj. pomocí sémantického tabla.
  - **Rezoluční metodou**: klazulární forma (konjuktivní normální forma).