

1 Prolog: fakta, pravidla, proměnné, funkce. Interpretace a model. Dotazy a jejich vyhodnocení rezoluční metodou, rozhodnutelnost. Řez, negace, předpoklad uzavřeného světa. (A4B33FLP)

1.1 Prolog

- deklarativní programovací jazyk - popisuje pravidla, kterým má řešení vyhovovat, ale neříká nic o postupu řešení, to se u prologu hledá rezoluční metodou predikátové logiky
- základem prologu je databáze klauzulí
- znalostní databáze se dále dá dělit na fakta a pravidla

1.1.1 fakta

- nejjednodušší klauzule, které popisují vlastnosti objektů pomocí predikátů
 - $\text{Man}(X)$. “X je muž”

1.1.2 pravidla

- složitější klauzule, které se skládají z faktů a zapisují se pomocí implikace
- pravidlo dává možnost odvozovat nová fakta
 - $\text{son}(A,B) \text{ :- } \text{parent}(B,A), \text{man}(A)$. “Pokud je B rodič A a zároveň A je muž, pak A je z pohledu B syn.”

1.1.3 proměnné

- značí se velkým písmem nebo podtržítkem jako prvním znakem
- vyskytují se v klauzulích jako jejich účastníci
- rozsah platnosti je pouze v klauzuli

- hodnotu získává pomocí unifikace a její hodnota je po přiřazení neměnná do chvíle kdy nedojde k backtrackingu a pravidlo přiřazení je odvoláno

1.1.4 funkce

- podobný způsob jako u pravidel
- blok postupně vyhodnocovaných faktů
- pevná arita

1.2 Interpretace a model

- interpretace - postupné přiřazování hodnot jednotlivým proměnným rekurzivně všem predikátům i těm nově vzniklým
- model
 - konkrétní přiřazení hodnot proměnným, pro které je zjištěno, že je množina pravdivá
 - modelů může být víc nebo také žádný

1.3 Dotazování

- jednorázové tvrzení - způsob zjišťování pravdivosti různých faktů
- zadává se do příkazové řádky
- odpovědi na jednotlivé dotazy může být více
- vyhodnocování se provádí postupně shora
- výsledkem může být přiřazení hodnoty proměnné či prosté tvrzení true/false

1.3.1 rezoluce

- postupně dochází k unifikaci jednotlivých proměnných
- klasický resoluční algoritmus

1.4 řez

- speciální vestavěný predikát, značený jako “!”
- pomocí řezu zakážeme při vyhodnocování využití backtrackingu
- hodí se například ve chvíli, kdy je dostatečné vyhodnotit pouze první výsledek a další ostatní jsou nepotřebné

- pokud neuspějí predikáty uvnitř řezu, tak je predikát opuštěn jako neúspěšný

1.4.1 zelený řez

- tento řez ovlivňuje pouze efektivitu programu
 - $\text{max2}(X,Y,X) \text{ :- } X > Y, !. \text{max2}(X,Y,Y) \text{ :- } Y \geq X.$
- bez řezu by tento příklad byl funkční, ale po nalezení maxima by se neefektivně zabýval dalším hledáním

1.4.2 červený řez

- tímto řezem je ovlivněna funkčnost programu
 - $\text{max1}(X,Y,X) \text{ :- } X > Y, !. \text{max1}(X,Y,Y).$
- v tomto případě by bez řezu došlo chybně k označení maxima i Y

1.5 negace

- predikát, který uspěje v případě, že predikát uvnitř neuspěje a opačně
 - $\text{not}(X).$

1.6 uzavřenost světa

- konvence o faktech světa, které nejsou specifikovány
- všechna fakta databáze jsou pravdivé
- všechna nedefinovaná fakta se považují za nepravdu