

Datalog

Datalog

- Podrobnejšie teoretické základy na prednáškach
- Program v datalogu je množina pravidiel (implikácií), napr.
 $\text{zlozene_cislo}(Z) \leftarrow \text{krat}(X, Y, Z), \text{int}(X), \text{int}(Y), \text{not } X = 1, \text{not } Y = 1.$
 $\text{prvocislo}(Z) \leftarrow \text{int}(Z), \text{not } Z = 1, \text{not } \text{zlozene_cislo}(Z).$
- Syntax:
 <hlava>: <atom>
 <hlava> :- <telo>
 $\text{<telo>: <atom> | \backslash+ <atom> | <telo>, <atom>}$

Na vyhodnocovanie datalogových dotazov budeme používať Prolog, preto na ich zápis využijeme syntax Prologu:

- $\backslash+$ je negácia, $:-$ je “implikácia”

Datalog

- príklad datalogovského pravidla:

`res(N,J) :- emp(_,N,J,_,_,S,_), S>=2000.`

- na ľavej strane vždy len jeden pozitívny atóm
- premenné začínajú veľkým písmenom
- konštanty malými písmenami
- každá premenná sa v tele pravidla uvedená aspoň v jednom *pozitívnom EDB kontexte*
- `_` znamená anonymnú premennú
- na vyhodnocovanie aritmetických výrazov slúži operátor *is*:
 - napr. `X is 2+3`, nie `X = 2+3`
(symbol = by bol interpretovaný ako unifikácia termov a nedôjde k žiadnej aritmetickej operácii).

Práca s datalogom: SWI-Prolog

- tri možnosti:
 - na serveri *cvika*, pripojiť sa cez ssh na `cvika.dcs.fmph.uniba.sk`
(prihlasovacie meno / heslo by malo byť ako v AISe)
 - v Linuxe v M217, alebo na vlastnom počítači, kde nainštalujete SWI-Prolog
 - online na <https://swish.swi-prolog.org/>
- odporúčame otvoriť si 3 okná
 - v jednom editujete súbor s dotazmi, napr. **`vim queries_emp.pl`**
 - v druhom okne máte spustené prostredie prologu: **`swipl -s queries_emp.pl`**
 - v treťom okne máte databázu (zoznam faktov)

Práca s datalogom

- po zapísaní dotazu do súboru ho treba uložiť na disk (vim: "ESC" a ":w").
- potom novú verziu skompilovať: **make.** (aj s tou bodkou)
 - nezabudnite skontrolovať, či kompilátor hlási chyby a prípadne ich opraviť
- výpočet dotazov:
?- q(job(J)).

Predikát "q(_)" slúži na pekné formátovanie výstupu a elimináciu zdanlivých duplikátov

Databáza EMP

```
% emp(Employee number, Name, Job, Superior, Employed from, Salary, Department Number)
emp(11, king, president, null, 1990, 5000, 10).
emp(12, blake, manager, 11, 1992, 2850, 30).
emp(13, clark, manager, 11, 1993, 1500, 10).
emp(14, jones, manager, 11, 1995, 2975, 20).
emp(15, martin, salesman, 12, 1998, 2650, 30).
emp(16, allen, salesman, 12, 1998, 1900, 30).
emp(17, turner, salesman, 12, 1995, 1500, 30).
emp(18, james, clerk, 15, 2000, 950, 30).
emp(19, ward, salesman, 12, 1996, 1750, 30).
emp(20, ford, analyst, 13, 1997, 3000, 20).
emp(21, smith, clerk, 12, 1993, 800, 20).
emp(22, scott, analyst, 13, 1996, 3000, 20).
emp(23, adams, clerk, 11, 1999, 1100, 20).
emp(24, miller, clerk, 14, 2000, 1300, 10).

% dept(Department Number, Name, Location)
dept(10, accounting, newyork).
dept(20, research, dallas).
dept(30, sales, chicago).
dept(40, operations, boston).
```

Datalog a negácia

- Dvojice J, D, že pozícia J nie je v departmente D
 `jobDept(J, D) :- emp(_,_,J,_,_,_,D).`
 `nemaJob(D) :- emp(_,_,J,_,_,_,_), emp(_,_,_,_,_,_,D), \+ jobDept(J,D).`

Prečo nestačí napísať toto?

```
nemaJob(D, J) :-  
    emp(_,_,J,_,_,_,_,_),  
    emp(_,_,_,_,_,_,_,D),  
    \+emp(_,_,J,_,_,_,_,D)
```

Datalog a všeobecný kvatifikátor

- je nutné prepísať všeobecný kvantifikátor ako negáciu existenčného, čiže v pomocnom pravidle popíšeme protipríklad a negovaním pomocného predikátu povieme, že protipríklad neexistuje
- „Departments containing all job positions“

allJobs(D) :- dept(D,_,_,_), \+ nemaJob(D).

nemaJob(D) :- emp(_,_,J,_,_,_,_), emp(_,_,_,_,_,_,D), \+ jobDept(J,D).

jobDept(J, D) :- emp(_,_,J,_,_,_,D).