

Zadanie č. 4
Dopredný produkčný systém

Ondrej Harnúšek
ID: 79545

Zadanie

Úlohou je vytvoriť jednoduchý dopredný produkčný systém, ktorý na základe odvodzovacieho pravidla modus ponens (pravidlo odlúčenia) odvodzuje zo známych faktov a produkčných pravidiel nové fakty. Produkčný systém ako program nepozná konkrétne pravidlá ani fakty! Pozná len formalizmus, v tomto prípade štruktúru pravidiel a faktov a spôsob ich spracovania. Pozná akcie pridaj, vymaž, a správa, ktoré sa môžu vykonávať, lebo tie patria do opisu formalizmu.

Použitý algoritmus

Hlavná metóda sa nachádza v triede `Runner`. Trieda slúži na načítanie vstupných údajov zo súboru a vytvorenie bázy pravidiel a faktov. Následne sa vytvorí objekt triedy `Inference`, v ktorej sa vykonáva samotný cyklus odvodzovania.

```
public static LinkedList<Rule> rules_base;
public static LinkedList<Expression> facts_base;
/**
 * Algoritmus inferencie dopredneho produkneho systemu
 */
public Inference()
    for(int i=0; i<LIMIT; i++)
        LinkedList<Binding> allInstances = new LinkedList<Binding>();
        allInstances = generateAllInstances();
        filterInstances(allInstances);
        if(allInstances.isEmpty()) return;
        allInstances.get(0).executeActions();
```

Trieda obsahuje bázu vstupných pravidiel a faktov. Algoritmus pozostáva z cyklu v ktorom postupne:

1. vytvorí zoznam aplikovateľných inšancií naviazaním všetkých pravidiel na všetky fakty.
2. odfiltruje inšancie, ktoré určí metóda `needFilter()`. Sú to tie, ktoré chcú vykonať výhradne akcie, ktoré pridávajú fakty už v báze obsiahnuté, vymazávajú fakty v báze neobsiahnuté a akcie typu správa.
Taktiež sa odfiltrujú duplicitné inšancie.
3. ak sa už žiadne aplikovateľné inšancie v zozname nenachádzajú cyklus odvodzovania končí
4. vyberie sa prvá inšancia zo zoznamu a vykonajú sa jej akcie.

Reprezentáciu údajov problému

Pri načítaní vstupných údajov zo súboru sa štruktúrovaný text namapuje na objekty rôznych tried.

Základnou triedou je `Expression`, ktorá obsahuje výraz - zoznam slov a vykonáva operácie nad ním. Trieda reprezentuje fakt, elementárnu podmienku pravidla alebo elementárnu akciu.

Ďalšou triedou je `Action`, ktorá predpisuje triedam `AddAction`, `DeleteAction` a `MessageAction` metódu `execute()`. Túto metódu implementujú jednotlivé triedy podľa svojho druhu. Prvé dve pracujú nad bazou faktov a tretia vypisuje správy do súboru.

Trieda `Rule` vykonáva operácie nad pravidlom. Obsahuje meno, zoznam elementárnych podmienok a elementárnych pravidiel. Obsahuje metódu `setBindings()`, ktorá rekurzívne na každé pravidlo naviaže fakty a vytvorí aplikovateľné inšancie.

Prvé volanie metódy posiela: $i=0$, čo je index v liste elementárnych podmienok. Ďalej prázdny list naviazaní a prázdnu HashMapu premenných.

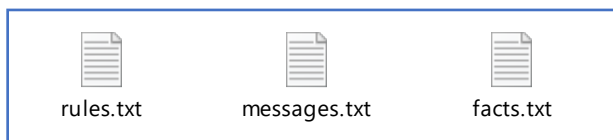
```
/**
 * Rekurzivne prechadza elementarne podmienky a nacita premenne
 */
private void setBindings(int i, LinkedList<Binding> binds, HashMap<String, String>
vars) {
    if(i >= conditions.size()) {//prešiel úspešne cez podmienky
        binds.add(new Binding(this, vars));
        return;
    }
    for(Expression fact : Inference.facts_base) {
        HashMap<String, String> myVars = new HashMap<String,String>(vars);
        if(setVariables(conditions.get(i), fact, myVars)) {
            setBindings(i+1, binds, myVars);
        }
    }
}
```

Metóda `setVariables()` postupne prechádza slovami elementárnej podmienky a slovami faktu. Porovnáva ich a premenné ukladá do HashMapy. Taktiež kontroluje platnosť špeciálnej podmienky $<> ?X ?Y$. Ak sa úspešne naviažu, zavolá s inkrementovaným indexom.

Poslednou triedou je `Binding`, ktorá reprezentuje aplikovateľnú inštanciu. Obsahuje pravidlo a HashMapu premenných, ktoré sú na pravidlo naviazané.

Vstupné údaje

Program podporuje prácu na rôznych doménami. DOMAIN = {"family","vehicle","blasphemy"}; Vstupné údaje – pravidlá a fakty, sa nachádzajú v priečinku pomenovanom podľa domény. Výstupné správy sa ukladajú do súboru. Do konzoly sa vypisujú interné údaje programu- aktuálne fakty a inštalácie.



Program prijíma ako vstupný parameter index domény nad ktorou pracuje.

Zhodnotenie riešenia

Program podporuje akcie typu *pridaj, vymaž a správa*. Program je možné jednoducho, pomocou polymorfizmu rozšíriť o ďalšie typy akcií. Podporuje tiež špeciálnu podmienku $<> ?X ?Y$. Program bol použitý na doméne *rodinné vzťahy, fiaty* a na vlastnej, ktorá je pribalená k projektu. Riešenie je overené vzorovými výstupmi programu na stránke predmetu.