

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4

4.Zadanie – Jednoduchý produkčný systém

Študijný program: Informatika
Ročník: 2
Cvičenie: Ut 14:00 -1.30b
Cvičiaci: Ing. Matúš Pikuliak
Predmet: Umelá inteligencia
Akademický rok: 2019/2020

Reprezentácia znalostí

Znalosti si reprezentujem pomocou listu stringov s názvom **pracovnaPamat**. Mená osôb, ktoré budem navazovať v podmienkach pravidiel mám uložené v samostatnom liste stringov s názvom **osoby**. Načítane pravidiel mám uložené v liste **nacitanePravidla**, ktorého každý prvok je objektom classy **Pravidlo**, ktorá má tieto atribúty:

- **cisloPravidla** – podľa toho v akom poradí bolo načítané pravidlo z textového súboru **pravidla.txt** dostane číslo od 1 po n. Tento atribút je v classe potrebný z dôvodu usporadúvania prvok v min halde **ovplyvniaPracovnuPamat**, ktorej význam je opísaný v špecifických vlastnostiach.
- **nazov** – ak používateľ zvolí krokovanie riešenie, tak sa pri každom vykonaní akcií jedného pravidla vypíše názov pravidla z dôvodu kontrolovania chodu programu.
- **podmienky** – v tomto atribúte sú uložené nenaviazané podmienky pravidla
- **akcie** – v atribúte sú uložené nenaviazané akcie pravidla

Vytvorené aplikovateľné inštancie pravidla sa ukladajú do listu **instanciePravidla**, prvok tohto listu je tiež objektom classy **Pravidlo**, avšak v atribúte **podmienky** sa už nachádzajú podmienky, ktoré sú naviazané a platné. Taktiež v atribúte **akcie** sú už uložené naviazané akcie pravidla, ktoré môžu ovplyvniť pracovnú pamäť.

Aplikovateľné inštancie pravidla, ktorých akcie ovplyvnia pracovnú pamäť ukladám do min haldy **ovplyvniaPracovnuPamat**, ktorej prvky sú tiež objekty classy **Pravidlo**, ale v atribúte **akcie** je uložená minimálne jedna naviazaná akcia, ktorá ovplyvní pracovnú pamäť.

Algoritmus

Pri spustení programu sa zo súboru **fakty.txt** načíta báza znalostí do listu **pracovnaPamat** a zároveň sa do listu **osoby** načítajú mená všetkých osôb, ktoré sa nachádzajú v báze znalostí a budú sa navazovať na podmienky pravidiel. Následne sa z **pravidla.txt** načítajú do **nacitanePravidla** všetky potrebné údaje o pravidlách, s ktorými bude jednoduchý produkčný systém pracovať.

Potom sa spustí cyklus, v ktorom sa budú vytvárať aplikovateľné inštancie pravidiel pomocou rekurzívnej funkcie **vytvorPodmienkyPravidla**. Funkcia dokáže naviazať ľubovoľne veľké množstvo neznámych v zloženej podmienke pravidla, ale aj elementárnej. Taktiež zvláda naviazať rôzny počet elementárnych podmienok, z ktorých je tvorená jedna zložená podmienka.

Počas chodu funkcia kontroluje pravdivosť elementárnej podmienky, ktorá bola naviazaná. Ak je elementárna podmienka nepravdivá, tak ďalej nepokračuje v navazovaní vytvárateľnej aplikovateľnej inštancie pravidla. V prípade, že je elementárna podmienka pravdivá, tak pokračuje v navazovaní ďalšej elementárnej podmienky, z ktorej sa skladá zložená podmienka pravidla.

Ak sa naviazali všetky elementárne podmienky v zloženej podmienke pravidla, tak to svedčí o tom, že podmienka vytvorenej aplikovateľnej inštancie pravidla je pravdivá a teda sa začnú navazovať akcie pre vytvorenú inštanciu.

To aké meno v akcií má byť dosadené za konkrétne písmenko viem určiť pomocou listu **pridaneOsoby**, ktorý som si postupne vytváral pri navazovaní podmienky. Každý prvok listu je objektom classy **PridanaOsoba**, ktorá má atribúty **meno** a **písmenko**. V atribúte **písmenko** je uložené písmenko, za ktoré sa v podmienke nahrádzala hodnota atribútu **meno**.

Meno: Jakub Hlavačka
AIS ID: 96876

Po naviazaní mien na akcie vytvorených aplikovateľných inštancií pravidiel idem zisťovať, či aspoň jedna akcia danej inštalácie ovplyvní pracovnú pamäť. Ak áno, tak ju vhodím do min haldy **ovplyvniaPracovnuPamat**.

Keď prejdem všetky vytvorené aplikovateľné inštalácie pravidiel a vyhodím tie, ktoré neovplyvnia pracovnú pamäť, tak z haldy vyberiem prvý prvok a jeho akcie vykonám nad listom **pracovnaPamat**. V prípade, že je halda prázdna, tak skončí program. Ak halda nie je prázdna, tak ju vyprázdním a vykoná sa ďalšia iterácia cyklu.

Na konci programu sa výsledná **pracovnaPamat** zapíše do **fakty_vystup.txt**.

Špecifické vlastnosti

Vzhľadom na to, že výber inštalácie, ktorej akcie sa budú aplikovať nemá byť ovplyvnený nejakou heuristikou alebo iným pravidlom, tak som zvolil výber podľa atribútu `cisloPravidla` v `classe Pravidlo`. Pretože keď nie je špecifikovaný výber selekcie, tak sa bežne vykonávajú akcie pravidla, ktoré sa načítalo skôr z bazy pravidiel. O to aby boli pravidlá usporiadané podľa ich čísla sa stará min halda **ovplyvniaPracovnuPamat**.

Štruktúrovanie súboru fakty.txt

Každý fakt je potrebné zapisovať do samostatného riadku, pričom začiatkové písmenká pri menách osôb je potrebné písať s veľkým písmenom, inak ich systém nebude vedieť rozoznať. Taktiež je potrebné aby sa za každým aj za posledným faktom nachádzal znak nového riadku.

Potrebné písať bez diakritiky.

Štruktúrovanie súboru pravidla.txt

Každé pravidlo je štruktúrované na tri riadky, pričom v prvom riadku sa nachádza jeho názov, za ktorým nasleduje dvojbodka a znak nového riadku.

V ďalšom riadku sa nachádza zložená podmienka, ktorá začína veľkými písmenami AK, za ktorými nasleduje jedna medzera. Celá podmienka je uzátvorkovaná v obyčajných zátvorkách, pričom každá elementárna podmienka má svoje vlastné uzátvorkovanie. Podmienka končí znakom nového riadku.

V poslednom riadku pravidla sú akcie, ktoré sa vykonávajú v prípade, že je podmienka pravdivá a ich činnosť ovplyvní pracovnú pamäť. Akcie začínajú písmenami POTOM, za ktorými je jedna medzera. Všetky akcie sú uzátvorkované v jeden veľkej obyčajnej zátvorke, pričom každá akcia má svoje vlastné zátvorky.

Typ akcie (pridaj, vymaz, sprava) je potrebné špecifikovať pred jej obsahom malými písmenkami oddelenými od obsahu akcie jednou medzerou.

Následne nasleduje prázdny riadok a za ním je buď reprezentované ďalšie pravidlo, alebo koniec súboru.

Potrebné písať bez diakritiky.

Zhodnotenie činnosti vytvoreného systému

Systém som testoval nad bazou známych faktov, ktoré sú dostupné na stránke predmetu. Následne som z nich odvodil dva zložitejšie príklady, ktoré tiež fungovali, pričom som ich výstup kontroloval s výstupom jednoduchého produkčného systému na stránke. Testované známe fakty sa nachádzajú v súbore **vstupy_faktov.txt**.

Meno: Jakub Hlavačka
AIS ID: 96876

V rámci toho aby som zistil, či systém funguje nad rôznymi rodinnými pravidlami som vytvoril nové pravidlo sestra, ktoré sa nachádza v **pravidla.txt** a systém s ním automaticky pracuje.

Počas riešenia príkladu je možné využiť krokovania, pričom sa v každej iterácii vypíše názov pravidla a vykonané akcie.

Do jednoduchého produkčného systému je možné pridávať rôzne pravidlá aj známe fakty, len je potrebné dodržiavať štruktúry súborov, ktoré sú opísane v Špecifických vlastnostiach.