Katedra Wytrzymałości Materiałów
i Metod Komputerowych Mechaniki
www.kwmimkm.polsl.pl

Wydział Mechaniczny Technologiczny
Politechnika Śląska

Inżynieria wiedzy

Instrukcja do zajęć laboratoryjnych

3. Reguły złożone w języku CLIPS

Opracował: mgr inż. Jacek Ptaszny
jacek.ptaszny@polsl.pl

Gliwice 2008

1 Cel ćwiczenia

Wykonując ćwiczenie udoskonalisz umiejętność formułowania reguł złożonych za pomocą języka CLIPS.

2 Zanim przejdziemy dalej

Zapoznaj się z informacjami podanymi poniżej.

*Wyróżniamy dwa rodzaje reguł: **proste** oraz **złożone**.*

***Reguły proste** mają postać wniosków pośrednich. Ich zaletą jest łatwość weryfikacji zbioru reguł i ograniczenie ich redundancji. Wadą jest konieczność realizacji złożonych zadań przez procedurę wnioskującą, ponieważ konieczne jest uaktywnienie wielu reguł. Kolejne wnioski pośrednie tworzą łańcuch wnioskowania [5].*

***Reguły złożone** umożliwiają bezpośrednie formułowanie wniosków przez system ekspertowy. Reguły te nie wymagają skomplikowanej procedury wnioskującej. Do osiągnięcia wyniku wystarcza uaktywnienie jednej reguły. Wadą reguł złożonych jest trudność formułowania zbioru reguł oraz złożony sposób jego weryfikacji i uzupełniania [5].*

Podczas tworzenia reguł złożonych przydatne będą operatory i polecenia języka CLIPS zestawione w poniższych tablicach.

Elementarne operatory arytmetyczne

Operator	Opis działania	Przykład użycia	Wynik działania
+	Dodawanie	+ 1 2 3	1+2+3
-	Odejmowanie	- 2 ?a	2-?a
*	Mnożenie	* ?x 10 2	?x*10*2
/	Dzielenie	/ 1 2	1/2
**	Potęgowanie	** 10 2	10^2
=	Utworzenie atomu z wyniku działania	(assert (wynik =(** 10 2)))	(wynik 100)

Operatory i funkcje w języku CLIPS mają postać prefiksową - najpierw występuje operator lub funkcja, a później jego (jej) argumenty.

Funkcje logiczne, funkcje porównania oraz ich użycie

Funkcja	Znaczenie
!	Negacja (NOT)
&&	Koniunkcja (AND)
||	Suma logiczna (OR)
=	Równy (tylko dla liczb)
eq	Równy (dla liczb i dla łańcuchów znaków)
!=	Nie równy
>=	Większy lub równy
>	Większy
<=	Mniejszy lub równy
<	Mniejszy
test	Funkcja testująca, np. (test (|| (< ?a 10) (> ?b 20)))

Operatory logiczne dla atomów

Operator	Znaczenie	Przykład użycia
∼	Negacja	(liczba ∼10)
&	Koniunkcja (AND)	(liczba ?b&∼10)
|	Suma logiczna (OR)	(liczba 10|20)

3 Zaczynamy!

Twoim zadaniem jest napisanie programu, który określi wartości zmiennych spełniających następujące działanie[1]:

$$\begin{array}{cccccc} & A & B & C & D & E \\ + & E & D & C & B & A \\ \hline = & F & F & F & F & F \end{array}$$

Zmienne A÷F są liczbami całkowitymi z przedziału od 0 do 9. Dodatkowo zmienne A÷E są różnymi liczbami, a ich suma wynosi 10. Liczba utworzona z cyfr ABCDE zawiera się w przedziale od 20 000 do 99 999.

Jak się pewnie domyślasz, aby rozwiązać zadanie należy najpierw wygenerować pewien zbiór kombinacji przyporządkowujących literom liczby (te kombinacje to fakty). Następnie spośród tych kombinacji należy wybrać te, które spełnią wymienione powyżej warunki. Posłuży do tego odpowiednia reguła złożona.

Wykonaj poniższe polecenia.

3.1 Wprowadź kod programu

```
 (defrule start
  =>
  (printout t t "Definicja zadania" t t)
  (printout t "   A B C D E" t)
  (printout t " + E D C B A" t)
  (printout t "   ---------" t)
  (printout t " = F F F F F" t t)
  (assert (cyfra 0)
          (cyfra 1)
          (cyfra 2)
          (cyfra 3)
          (cyfra 4)
          (cyfra 5)
          (cyfra 6)
          (cyfra 7)
          (cyfra 8)
          (cyfra 9)
          (litera A)
          (litera B)
          (litera C)
          (litera D)
          (litera E)
          (litera F)))

(defrule generuj-kombinacje
  (cyfra ?x)
  (litera ?a)
  =>
  (assert (kombinacja ?a ?x)))
```

[1] Przykład zaczerpnięto z książki [3]

```
(defrule znajdz-rozwiazanie
  (kombinacja A ?a)
  (kombinacja B ?b&~?a)
  (kombinacja C ?c&~?b&~?a)
  (kombinacja D ?d&~?c&~?b&~?a)
  (kombinacja E ?e&~?d&~?c&~?b&~?a)
  (kombinacja F ?f)
  (test (= (+ ?e ?a) ?f ) )
  (test (= (+ ?d ?b) ?f ) )
  (test (= (+ ?c ?c) ?f ) )
  (test (= (+ ?a ?b ?c ?d ?e) 10))
  (test (>= ?a 2) )
  =>
  (printout t "Rozwiazanie:" t)
  (printout t ?a ?b ?c ?d ?e t))
```

3.2 Uruchom program i sprawdź jego działanie

Program powinien znaleźć cztery rozwiązania: ABCDE = 34201, 30241, 43210, 41230.

3.3 Napisz program rozwiązujący zadanie przedstawione przez prowadzącego zajęcia

4 Podsumowanie

Wykonując ćwiczenie zdobyłeś(aś) umiejętność tworzenia reguł złożonych w jezyku CLIPS, z wykorzystaniem:

- operatorów arytmetycznych,
- funkcji logicznych,
- funkcji porównania,
- operatorów logicznych dla atomów.

Literatura

[1] Cholewa W., Pedrycz W., *Systemy doradcze.* Wydawnictwo Politechniki Śląskiej, Gliwice 1987.

[2] Giarratano J. C., *CLIPS User's Guide*, http://clipsrules.sourceforge.net/.

[3] Jeleński S., Rozrywki matematyczne. I Lilavati, WSiP, Warszawa 1992.

[4] Kendal S., Creen M., *An Introduction to Knowledge Engineering.* Springer-Verlag, London 2007.

[5] Mulawka J. J., *Systemy ekspertowe.* WNT, Warszawa 1996.

[6] Rutkowski L., *Metody i techniki sztucznej inteligencji.* WNT, Warszawa 2005.

[7] Russel S., Norvig P., *Artificial intelligence: A Modern Approach.* Prentice Hall, 2002.