

Inteligența artificială

Laborator 1

Jess – mod de instalare

- Se descarcă eclipse juno sau o versiune mai veche
- Se descarcă jess de pe jessrules.com
- Se deschide eclipse juno, apoi se închide
- Se dezarchivează fișierele ce sunt în folderul jess->eclipse și se copiază în folderul eclipse juno
- Se deschide eclipse-ul și se verifică dacă jess este instalat: Help-> About Eclipse->



- Se creează un fișier gol cu extensia .clp într-un proiect java

Jess. Noțiuni introductive

1. Templates

(deftemplate <deftemplate-name> [<comment>] <slot-definition>*)

<slot-definition>::=<single-slot-definition> | <multislot-definition>

<single-slot-definition>::=(slot <slot-name> <template-attribute>*)

<multislot-definition>::=(multislot <slot-name> <template-attribute>*)

<template-attribute>::=<default-attribute> | <constraint-attribute>

<default-attribute>::=(default ?DERIVE | ?NONE | <expression>*) |

(default-dynamic <expression>*)

ex. (deftemplate object (slot name (type SYMBOL) (default ?DERIVE))
(slot location (type SYMBOL) (default ?DERIVE))
(slot on-top-of (type SYMBOL) (default floor))
(slot weight (allowed-values light heavy) (default light))
(multislot contents (type SYMBOL) (default ?DERIVE)))

```
deftemplate object1 (slot name(type SYMBOL)(default ?DERIVE))
                    (slot location(type SYMBOL)(default ?DERIVE))
                    (slot on-top-of(type SYMBOL)(default floor))
                    (slot weight(allowed-values light heavy)(default light))
                    (multislot contents(type SYMBOL)(default ?DERIVE))
```

2. Deffacts

```
(deffacts <deffacts-name> [<comment>] <RHS-pattern>*)

(deffacts startup "Refrigerator Status" (refrigerator light on)

                                     (refrigerator door open)

                                     (refrigerator temp (get-temp)))
```

3. Defrule

```
(defrule <rule-name> [<comment>]

    [<declaration>]                ; Rule Properties

    <conditional-element>*          ; Left-Hand Side (LHS)
    =>

    <action>*)                     ; Right-Hand Side (RHS)

<conditional-element> ::= <pattern-conditional-elements> |

                        <assigned-pattern-conditional-elements > |

                        <not-conditional-elements > |

                        <and-conditional-elements > |

                        <or-conditional-elements > |

                        <logical-conditional-elements > |

                        <test-conditional-elements > |

                        <exists-conditional-elements > |

                        <forall-conditional-elements >
```

ex.

```
(defrule rule-1 (factoid a)

    =>

    (assert (factoid b)))

(defrule rule-2

    ?f <- (factoid a) (factoid d)

    =>

    (retract ?f)

    (assert (factoid c)))
```

Variabile single sau multicamp

`<constraint> ::= <constant> | ? | $? | <single-field-variable> | <multifield-variable>`

`<single-field-variable> ::= ?<variable-symbol>`

`<multifield-variable> ::= $?<variable-symbol>`

Conectori pentru constrangeri

`<constraint> ::= ? | $? | <connected-constraint>`

`<connected-constraint> ::= <single-constraint> |`

`<single-constraint> & <connected-constraint> |`

`<single-constraint> | <connected-constraint>`

`<single-constraint> ::= <term> | ~<term>`

`<term> ::= <constant> | <single-field-variable> | <multifield-variable>`

ex.

`(defacts AB (data-A green)`

`(data-A blue)`

`(data-B (value red))`

`(data-B (value blue)))`

`(defrule example1-1 (data-A ~blue) =>)`

`(defrule example1-2 (data-B (value ~red&~green)) =>)`

`(defrule example1-3 (data-B (value green|red)) =>)`

Cei trei conectori sunt & (si), | (sau), si ~ (not). Constrangerea & este satisfacuta daca ambele constrangeri ce o formeaza sunt satisfacute. Constrangerea | este satisfacuta una din constrangerile ce o formeaza este satisfacuta. Constrangerea ~ este satisfacuta daca urmatoarea constrangere nu este satisfacuta.

1. Deffunction

`(deffunction <name> [<comment>] (<regular-parameter>* [<wildcard-parameter>]))
 <action>*)`

`<regular-parameter> ::= <single-field-variable>`

`<wildcard-parameter> ::= <multifield-variable>`

Un deffunction este format din cinci elemente: 1) un nume, 2) un comentariu opțional, 3) o listă cu zero sau mai multi parametri, 4) un parametru wildcard opțional care se ocupe de un număr variabil de argumente și 5) o secvență de acțiuni sau expresii, care vor fi executate în ordine, când deffunction este apelat.

```
(deffunction factorial (?a)
```

```
  (if (or (not (integerp ?a)) (< ?a 0)) then
```

```
    (printout t "Factorial Error!" crlf)
```

```
    else (if (= ?a 0) then 1
```

```
            else (* ?a (factorial (- ?a 1))))))
```

Bibliografie:

1. <http://www.jessrules.com/jess/docs/71/>

Probleme:

1. Convertiti urmatoarele pragrafe in fapte cu ajutorul lui assert sau deffacts. Pentru fiecare grup de fapte, definiti un template cu ajutorul lui deftemplate. Pentru fiecare din paragrafele de mai jos construiti cate o aplicatie JESS.
 - a. Ion Popescu este bunicul lui Ionel Andreescu. Soția sa se numeste Maria Popescu. Ion Popescu si Maria Popescu au trei fii: Cristian, Adrian si Ciprian, respectiv o fiica - Maria Andreescu. Ionel Avram este fiul lui Maria Andreescu. Sotul Mariei Andreescu este Petre Andreescu. Ionel Avram are mai multe matusi: Elena Popescu (soția lui Cristian Popescu), Cristina Popescu si Adriana Popescu. Ionel Avram are un verisor, Tudor Popescu care este fiul lui Adrian si Cristina Popescu.
 - b. In arca lui Noe au fost asezate mai multe animale in functie de inaltimea lor. Se stie ca au fost aduse in arca cate doua animale din fiecare specie, ordinea de asezare fiind mascul apoi femela. Asadar se considera pentru simplificare urmatoarea lista de animale ordonate dupa inaltime : soarece, veverita, pisica, caine, vaca, elefant.Pentru acest subiect se cere construirea de fapte care sa indice pozitionarea animalelor in arca.
2. Definiti unul sau mai multe template-uri utile pentru a descrie multimii algebrice. Template- ul va trebui sa contina numele multimii, elementele multimii sau o descriere a elementelor multimii (multimea in acest caz va fi descrisa analitic, de exemplu $A =$

$\{x|x \geq 2, x \in \mathbb{R}\}$). De asemenea template-ul va trebui sa specifice daca o multime este sau nu o submultime a altei multimi. Reprezentati urmatoarele multimi:

$A = \{1,2,3,4,5\}, A \subset C$

$B = \{\text{albastru, verde, galben, rosu, negru}\}$

$C = \{x|x \geq 0, x \in \mathbb{R}\}$

3. Calculati suma urmatoarelor elemente $A=2, B=3, C=7, D=1001$.
4. Gradinile Zoologice adapostesc cel mai adesea o multitudine și o diversitate extraordinara de specii de animale. Fiecare animal mananca anumite nutrimente, are un anumit mod de viata (diurn/nocturn), mediu de viata (terestru, aerian, suprateran, subteran, acvatic, semiacvatic, etc), o anumita modalitate de reproducere (prin oua, prin pui vii) etc. Pornind de la informatiile de mai sus, adaugati și alte informatii specifice animalelor pentru a defini un template care sa le descrie din perspectiva unei gradini zoologice.

Adaugati in baza de cunostinte mai multe fapte. Tineti cont de cerintele de mai jos la adaugarea acestora:

- a. Scrieti o regula JESS care sa permita afisarea tuturor animalelor care au mediul de viata semiacvatic (de exemplu: o broasca);
- b. Scrieti o regula JESS care sa permita afisarea tuturor animalelor care au regim alimentar vegetarian și traiesc in zona mediteraneana.
- c. Scrieti o regula care sa afiseze perechi de animale de tipul (animalConsumator, animalConsumat), de exemplu (lup, oaie). Regula va permite aceasta afisare, iar daca animalul consumat este furnica, atunci pradatorul este furnicar.
- d. In aceasta gradina zoologica exista din zona polara doar un urs si o foca. Scrieti o regula care sa permita afisarea lor utilizand antipatternuri.