

MEDII DE PROGRAMARE IN INTELIGENTA ARTIFICIALA

Jess – fapte si reguli

1. Definirea templaturilor pentru fapte neordonate.
 2. Definirea regulilor.
 3. <Gasiti fratii si surorile> via JESS.
 4. Sisteme expert in Jess.
 5. Exercitii
1. **Fapte ordonate:** Rulati urmatoarele instructiuni la prompterul Jess>

```
Jess> (watch facts)
```

```
Jess> (assert (culori rosu verde albastru))
```

```
Jess> (reset)
```

```
Jess> (bind ?f (fact-id 1))
```

```
Jess>(retract ?f)
```

```
Jess> (deffacts catalog "Product catalog"
```

```
(product 354 set-creioane "50RON")
```

```
(product 355 paper-clips "6RON")
```

```
(product 356 set-pixuri "70RON"))
```

Notati raspunsurile obtinute si explicati ce inseamna acestea.

Fapte neordonate: Definiti urmatoarele template-uri:

```
(deftemplate mar
```

```
(slot soi (type SYMBOL) )
```

```
(slot dimensiune (type SYMBOL))
```

```
(slot culoare (type SYMBOL) (default rosu))
```

```
)
```

Incarcati in memorie urmatoarele fapte:

```
(deffacts tipuri_mere
```

```

(mar (variety Cortland) (size large) (color rosu))

(mar (variety Golden) (size large) (color galben))

(mar (variety Dublu-Rosu) (size mediu) (color rosu))

(mar (variety Granny-Smith) (size mare) (color verde))

)

```

Testati continutul memoriei de lucru prin folosirea comenzilor (facts), (reset), (watch facts).

2. Definim o regula care in functie de caracteristicile indicate, incarca un memorie un fapt de tip mar cu attributele specificate:

```

(defrule ask-size

=>

(printout t "Introduceti caracteristicile dorite : " crlf)

(printout t "Denumirea:" crlf)

(bind ?ans1 (read))

(printout t "- culoare (rosu, galben, verde) : ")

(bind ?ans2 (read))

(printout t crlf "-dimensiune (mare sau mediu) : ")

(bind ?ans3 (read))

(assert (mar (soi ?ans1) (culoare ?ans2) (dimensiune ?ans3)))

)

```

3. <Gasiti fratii si surorile> via JESS.

```

(deftemplate copil

(slot nume (type STRING))

(slot mama (type STRING))

(slot tata (type STRING))

(slot gen))

(deffacts incarca-copii

(copil (nume ana) (mama maria) (tata ion) (gen fem))

(copil (nume george) (mama elena) (tata ion) (gen masc)))

```

(defrule gaseste-frati

(copil (nume ?n1) (mama ?m1) (tata ?t1) (gen ?g1))

(copil (nume ?n2) (mama ?m2) (tata ?t2) (gen ?g2))

(not (test ((new java.lang.String ?n1) equals ?n2)))

(or (test ((new java.lang.String ?m1) equals ?m2))

(test ((new java.lang.String ?t1) equals ?t2)))

=>

(printout t ?n1 " si " ?n2 " sunt frati." crlf))

(watch all)

(reset)

(run)

(facts)

Exercitiu: Modificati programul de mai sus astfel incat sa diferentieze fratii de surori.

4. Exemplu de sistem expert in Jess.

Consideram urmatoarele reguli de rationament:

- Daca led-ul calculatorului nu este pornit atunci e indicat sa se cerceteze daca computerul este alimentat.
- Daca computerul porneste si boot-eaza SO dar apoi se stinge atunci e indicat sa se reinstaleze SO.
- Daca computerul functioneaza bine dupa reinstalarea SO atunci problema poate fi considerata rezolvata.

Construirea templaturilor pentru faptele programului:

(deftemplate stage

(slot value))

(deftemplate mains_light_on

(slot value))

(deftemplate boots

(slot value))

(deftemplate crashes

```
(slot value))
```

Regula de start a pogramului (nu are nicio conditie):

```
(defrule start
```

```
=>
```

```
(assert (stage (value 1)))
```

```
(printout t "Este led-ul computerului pornit?")
```

```
(assert (mains_light_on (value (readline))))
```

```
(printout t "Booteaza SO?")
```

```
(assert (boots (value (readline))))
```

```
(printout t "Computerul se opreste dupa?")
```

```
(assert (crashes (value (readline))))
```

```
)
```

Regulile de rationament:

```
(defrule idiot_check
```

```
(stage (value 1))
```

```
(mains_light_on (value "nu"))
```

```
=>
```

```
(printout t "Ar trebui sa alimentezi cu energie computerul !" crlf)
```

```
)
```

```
(defrule try_reinstalling
```

```
(boots (value "da"))
```

```
?cr <- (crashes (value "da"))
```

```
?st <- (stage (value 1))
```

```
=>
```

```
(printout t "Incerca sa instalezi SO" crlf)
```

```
(printout t "Acum computerul se mai opreste?")
```

```
(retract ?st ?cr)
```

```

(assert (crashes (value (readline))))

(assert (stage (value 2)))

)

(defrule dunno

  (stage (value 2))

  (crashes (value "da"))

  =>

  (printout t "Ar trebui sa duci computerul la service" crlf)

)

```

Rularea aplicatiei:

```
(reset)
```

```
(run)
```

5. **Exercitiu 1** Modificati programul de la punctul (3) astfel incat, daca utilizatorul raspunde cu "nu" la intrebarea atasata faptului <crashes> atunci sa i re raspunda

"Ma bucur ca v-am fost de ajutor. Nota de plata o veti primi prin posta"

Exercitiu 2 Modificati programul de la punctul (2) astfel incat toate faptele de tip mar ce au culoarea rosie sa devina "albastre".