

Grupa:
Numele:

1

Problema 1

3

a. Desenați rețeaua RETE pentru colecția de reguli următoare:

```
(defrule rule1
  (alpha ?a a ?a)
  (alpha a ?a ?a)
  (beta ?a ?x)
  =>
  (assert (gamma ?x ?a no)))

(defrule rule2
  (alpha a ?a ?a)
  (beta b ?a)
  =>
  (assert (gamma ?a ?a yes)))
```

2

b. desemnați instanțele de reguli care se acumulează în agendă la darea comenzii (reset) în rularea programului de la pct. a asupra faptelor:

```
(def facts done
  (alpha John a Mary)
  (alpha John a John)
  (alpha a Mary Mary)
  (alpha a John John)
  (beta John Mary)
  (beta Mary John)
  (beta b John)
)
```

1

c. Arătați care sunt faptele adăugate în baza după primul și cel de al doilea pas în rularea programului de la punctul anterior.

3

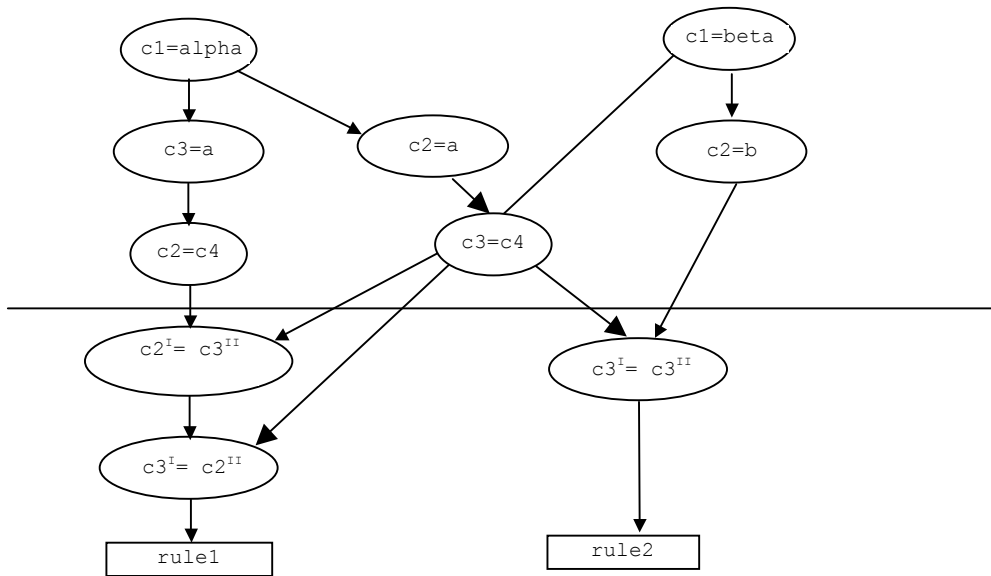
Problema 2

Aveți trei cutii în care se găsesc câte 100 de șnururi, în cutia A – numai roșii, în cutia B – numai galbene, în cutia C – numai albastre. În fiecare cutie șnururile sunt de diverse lungimi. Luând câte un șnur din fiecare cutie și legându-le între ele, doriți să creați 100 de șnururi tricolore, aproximativ de aceeași lungime. Cum procedați?

1
3

Soluții
Problema 1

a.



2

b. Agenda contine:
(rule1 f2, f4, f5; ?a←John; ?x←Mary)
(rule2 f4, f7; ?a←John)

1

c. Dupa primul pas:
(gamma John John yes)

Dupa al doilea pas:
(gamma Mary John no)

3

Problema 2

Sugestie de soluție. Se calculează lungimea media a șnururilor tricolore din fiecare culoare, fie ele MA, MB, MC. La fiecare alegere a 3 șnururi se selectează acele șnururi unicolore a căror lungime însumată este cea mai apropiată de MA+MB+MC.

```
; un șnur se identifică prin culoare (A, B, C), index unic și lungime
(deffacts snururi
  (snur A A1 10)
  (snur A A2 9)
  (snur A A3 8)
  (snur A A4 1)
  (snur A A5 4)
  (snur A A6 5)
  (snur A A7 11)
  (snur A A8 12)
  (snur A A9 7)
  (snur A A10 6)

  (snur B B1 10)
  (snur B B2 8)
  (snur B B3 8)
  (snur B B4 4)
  (snur B B5 4)
  (snur B B6 8)
  (snur B B7 7)
  (snur B B8 12)
```

```

(snur B B9 5)
(snur B B10 6)

(snur C C1 11)
(snur C C2 4)
(snur C C3 3)
(snur C C4 9)
(snur C C5 12)
(snur C C6 5)
(snur C C7 6)
(snur C C8 10)
(snur C C9 8)
(snur C C10 7)
(suma A 0 0)
(suma B 0 0)
(suma C 0 0)
)

; însumează lungimile șnururilor de aceeași culoare
(defrule media-add
  ?m <- (suma ?x ?sx ?Nx $?selx)
  (snur ?x ?idx ?lx)
  (test (not (member$ ?idx $?selx)))
  =>
  (retract ?m)
  (assert (suma ?x (+ ?sx ?lx) (+ ?Nx 1) $?selx ?idx))
)

; calculează mediile lungimilor șnururilor pe fiecare culoare
(defrule media-stop
  (declare (salience -10))
  ?s <- (suma ?x ?sx ?Nx $?)
  =>
  (retract ?s)
  (assert (media ?x (/ ?sx ?Nx)))
)

; însumează aceste medii
(defrule general-media
  ?ia <- (media A ?ma)
  ?ib <- (media B ?mb)
  ?ic <- (media C ?mc)
  =>
  (retract ?ia ?ib ?ic)
  (assert (lungMediuTric (+ ?ma ?mb ?mc)))
)

(defrule tricolor
  (lungMediuTric ?N)
  ?sa <- (snur A ?idxa ?la)
  ?sb <- (snur B ?idxb ?lb)
  ?sc <- (snur C ?idxc ?lc)
  (not (and (snur A ?idxaa&:(neq ?idxaa ?idxa) ?laa) (snur B ?idxbb&:(neq
?idxbb ?idxb) ?lbb) (snur C ?idxcc&:(neq ?idxcc ?idxc) ?lcc)
  (test (< (abs (- (+ ?laa ?lbb ?lcc) ?N)) (abs (- (+ ?la ?lb ?lc)
?N))))))
  =>
  (retract ?sa ?sb ?sc)
  (assert (tricolor (gensym) ?la ?lb ?lc))
)

```