

Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет информационных технологий и управления

Кафедра интеллектуальных информационных технологий

Дисциплина: «Математические основы интеллектуальных систем»

Лабораторная работа №3 по теме:

«Конвертация результатов лабораторных работ 1-3 в SCs»

Студент гр. 121702

Заломов Р.А.

Проверил:

Коршунов Р.А.

Минск 2022

## **Тема**

Конвертация результатов лабораторных работ 1-3 в SCs

## **Цель**

Получить навыки формального представления в базе знаний высказываний на языках SCs и SCn.

## **Задание**

Индивидуальные задания из л.р. 1-2 представить при помощи языка SCs, для результатов л.р. 3 заменить идентификаторы на системные. Результаты л.р. 1-3 собрать в базу и продемонстрировать работу системы.

## **Выполнение задания**

## 1. Конвертация л.р. 1

```
1 geometrical_problem <- sc_node_class;;
2
3 geometrical_problem -> problem_1(* -> [*
4
5 triangle <- sc_node_class;;
6 isosceles_triangle <- sc_node_class;;
7 line_segment <- sc_node_class;;
8 square <- sc_node_class;;
9 circle <- sc_node_class;;
10 length <- sc_node_class;;
11 exact_value <- sc_node_class;;
12
13 ..node1 <- sc_node_class;;
14 ..node2 <- sc_node_class;;
15 ..node3 <- sc_node_class;;
16
17 ..tuple <- sc_node_tuple;;
18
19 nrel_base <- sc_node_norole_relation;;
20 nrel_geometrical_side <- sc_node_norole_relation;;
21 nrel_lateral_side <- sc_node_norole_relation;;
22 nrel_height <- sc_node_norole_relation;;
23 nrel_inscribed <- sc_node_norole_relation;;
24
25
26
27 triangle => nrel_inclusion: isosceles_triangle(* -> TriangleABC;;*);;
28 triangle -> TriangleBDC;;
29 TriangleBDC => nrel_geometrical_side:DC;BC;BD;;
30 line_segment -> BC;AC;BD;DC;AB;;
31 exact_value -> ..node1;;
32 length -> ..node1;;
33 ..node1 -> BC;AB;;
34 TriangleABC => nrel_lateral_side:AB;BC;;
35 ..node1 => nrel_measurment:6 (* <- number;; *);;
36 ..node1 <- exact_value;;
37 ..node1 <- length;;
38 TriangleBDC => nrel_geometrical_side:DC;BC;;
39 square -> ..node2;;
40 ..node2 <- exact_value;;
41 ..node2 => nrel_measurment: S;;
42 S <- number;;
43 number -> 1;;
44 ..node3 => nrel_measurment : 1;;
45 length -> ..node3;;
46 ..node3 <- exact_value;;
47 ..node3 -> ..node4;;
48 circle -> ..node5(* => nrel_radius: ..node4;;*);;
49 ..node5 => nrel_inscribed: TriangleBDC;;
50 ..tuple -> AC;BD;;
51 ..tuple => nrel_intersection : ..dot(* -> D;; *);;
52 TriangleABC => nrel_base: AC;;
53 ..node2 <- square;;
54 ..node2 <- exact_value;;
55 ..node2 -> TriangleABC;;
56 BD => nrel_height: TriangleABC;;
57 TriangleABC => nrel_base: AC;;
```

```

57 TriangleABC => nrel_base: AC;;
58
59 *];; *);;
60

```

## 2. Конвертация л.р. 2

```

1 //описываем класс предприятие
2 concept_facility <- sc_node_class;;
3
4 concept_facility => nrel_main_idtf: [facility]
5   (*
6     <-lang_en;;
7   *);;
8 concept_facility => nrel_main_idtf: [предприятие]
9   (*
10    <-lang_ru;;
11  *);;
12 concept_facility => nrel_system_idtf: [concept_facility];;
13
14 complex <- sc_node_class;;
15 activity <- sc_node_class;;
16
17 concept_facility
18 <- rrel_key_sc_element: ...
19   (*
20     <- sc_definition;;
21     => nrel_main_idtf: [Омп. Предприятие]
22     (* <- lang_ru;; *) ;;
23     <= nrel_sc_text_translation: ...
24       (*
25         -> rrel_example: [Предприятие - имущественный комплекс, используемый для осуществления предпринимательской деятельности]
26         (*<-lang_ru;;*);;
27       *);;
28     => nrel_using_constants: ...
29     (* <- complex;activity;; *);;
30   *);;
31
32 sc_statement <- sc_node_class;;
33 balance <- sc_node_class;;
34 bank <- sc_node_class;;
35 account <- sc_node_class;;
36 stamp <- sc_node_class;;
37 nomination <- sc_node_role_relation;;
38
39 concept_facility
40 <- rrel_key_sc_element: ...
41   (*
42     <- sc_statement;;
43     => nrel_main_idtf: [Учр.л Предприятие]
44     (* <- lang_ru;; *) ;;
45     <= nrel_sc_text_translation: ...
46       (*
47         -> rrel_example: [Предприятие имеет самостоятельный баланс, расчётный и иные счета в банках, печать со своим наименованием]
48         (*<-lang_ru;;*);;
49       *);;
50     => nrel_using_constants: ...
51     (* <- balance; bank; account; stamp; nomination;; *);;
52   *);;
53
54 property_complex <- sc_node_class;;
55
56 property_complex => nrel_include: concept_facility;;
57

```

```

59 concept_facility -> roga_i_kopyta(* => nrel_manufacturing_department: paintintg_department: assembly_department: probation_department;; *);;
60
61
62 //описываем неролевое отношение начальник*
63 nrel_chief <- sc_node_norole_relation;;
64
65 nrel_chief => nrel_main_idtf: [chief*]
66   (*
67     <-lang_en;;
68   *);;
69 nrel_chief => nrel_main_idtf: [начальник*]
70   (*
71     <-lang_ru;;
72   *);;
73 nrel_chief => nrel_idtf: [руководитель*]
74   (*
75     <-lang_ru;;
76   *);;
77 nrel_chief => nrel_main_idtf: [boss*]
78   (*
79     <-lang_en;;
80   *);;
81 nrel_chief => nrel_system_idtf: [nrel_chief];;
82
83
84 official <- sc_node_class;;
85 power <- sc_node_class;;
86 nrel_subordinate <- sc_node_norole_relation;;
87
88 nrel_chief
89 <- rrel_key_sc_element: ...
90   (*
91     <- sc_definition;;
92     => nrel_main_idtf: [Оп. Начальник]
93     (* <- lang_ru;; *) ;;
94     <= nrel_sc_text_translation: ...
95     (*
96       -> rrel_example: [Начальник - должностное лицо, имеющее подчинённых и располагающее властью]

```

```

97       (*<-lang_ru;;*);;
98     *);;
99     => nrel_using_constants: ...
100     (* <- official;power;nrel_subordinate;; *);;
101   *);;
102
103 nrel_employee <- sc_node_norole_relation;;
104 nrel_need <- sc_node_norole_relation;;
105
106 nrel_chief
107 <- rrel_key_sc_element: ...
108   (*
109     <- sc_statement;;
110     => nrel_main_idtf: [УТВ.1 Начальник]
111     (* <- lang_ru;; *) ;;
112     <= nrel_sc_text_translation: ...
113     (*
114       -> rrel_example: [Грамотный начальник понимает потребности своих сотрудников]
115       (*<-lang_ru;;*);;
116     *);;
117     => nrel_using_constants: ...
118     (* <- nrel_employee; nrel_need;; *);;
119   *);;
120
121 ..class_node <- sc_node_class;;
122 arity -> ..class_node
123   (*
124     => nrel_measuring: 2 (* <- number;; *);;
125     -> nrel_chief;;
126   *);;
127
128 symmetric_relation -|> nrel_chief;;
129 reflexive_relation -|> nrel_chief;;
130 transitive_relation -|> nrel_chief;;
131 oriented_relation -> nrel_chief;;
132
133
134 human <- sc_node_class;;
135
136 human -> Vasiliy_Ivanovich;;
137 concept_facility -> roga_i_kopyta(* => nrel_chief: Vasiliy_Ivanovich;; *);;
138
139 nrel_chief => nrel_second_domain: human;;
140 nrel_chief => nrel_first_domain: manufacturing;;
141
142 nrel_chief => nrel_definition_domain: ...
143   (*
144     <= nrel_union:
145       {
146         human;
147         manufacturing
148       };;
149   *);;
150

```

### 3. Конвертация л.р. 3

```
1 logical_statement <- sc_node_class;; // Для удобства создадим класс "логическое высказывание"
2 atomic_existence <- sc_node_class;;
3
4 //Первое высказывание
5
6 logical_statement -> logical_statement_1 (* -> [*
7
8 ..tuple1 <- sc_node_tuple;;
9 ..tuple2 <- sc_node_tuple;;
10 ..tuple3 <- sc_node_tuple;;
11 nrel_responsible_for <- sc_node_norole_relation;;
12
13 vars <- sc_node_struct;
14 -> _x;;
15 -> _y;;
16
17 nrel_generality -> ..tuple1
18   (*
19     -> rrel_bind_variables: vars;;
20     -> ..tuple2
21       (*
22         <- nrel_implication;;
23         -> rrel_if: [* human _-> _x;; concept_facility _-> _y;; *] (* <- atomic_existence;; *);;
24         -> rrel_then: ..tuple3
25           (*
26             <- nrel_equivalence;;
27             -> [* x _=> nrel_responsible_for: _y;;*] (* <- atomic_existence;; *);;
28             -> [* _y _=> nrel_chief: _x;;*] (* <- atomic_existence;; *);;
29             *);;
30           *);;
31         *);;
32       *];; *);;
33     *);;
34   *);;
35
36 //Второе высказывание
37 logical_statement -> logical_statement_2 (* -> [*
38
39 ..tuple4 <- sc_node_tuple;;
40 ..tuple5 <- sc_node_tuple;;
41 bad <- sc_node_class;;
42 vars2 <- sc_node_struct;
43 -> _z;;
44
45 nrel_liquidate <- sc_node_norole_relation;;
46
47 nrel_existence -> ..tuple4
48   (*
49     -> rrel_bind_variables: vars2 (* <- atomic_existence;; *);;
50     -> ..tuple5
51       (*
52         <- nrel_implication;;
53         -> rrel_if: [* concept_facility _-> _z;; bad _-> _z;; *] (* <- atomic_existence;; *);;
54         -> rrel_then: [* _z _=> nrel_liquidate: _z;;*] (* <- atomic_existence;; *);;
55         *);;
56       *);;
57     *];; *);;
58   *);;
```

## Вывод

В результате лабораторной работы были получены практические навыки формального представления в базе высказываний на языке SCs. Помимо этого, были получены навыки работы с языком SCn. Также были получены навыки, связанные с работой на платформе OSTIS.