



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Московский государственный технический университет имени
Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика, искусственный интеллект и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчёт по лабораторной работе № 9 по курсу «Функциональное и логическое программирование»

Тема Использование правил в программе на Prolog

Студент Калашков П. А.

Группа ИУ7-66Б

Оценка (баллы) _____

Преподаватели Толпинская Н. Б., Строганов Ю. В.

Задание

Часть 1

Создать базу знаний «Предки», позволяющую наиболее эффективным способом (за меньшее количество шагов, что обеспечивается меньшим количеством предложений БЗ - правил), и используя разные варианты (примеры) простого вопроса, (указать: какой вопрос для какого варианта) определить:

1. По имени субъекта определить всех его бабушек (предки 2-го колена).
2. По имени субъекта определить всех его дедушек (предки 2-го колена).
3. По имени субъекта определить всех его бабушек и дедушек (предки 2-го колена).
4. По имени субъекта определить его бабушку по материнской линии (предки 2-го колена).
5. По имени субъекта определить его бабушку и дедушку по материнской линии (предки 2-го колена).

Минимизировать количество правил и количество вариантов вопросов. Использовать конъюнктивные правила и простой вопрос. Для одного из вариантов ВОПРОСА задания 1 составить таблицу, отражающую конкретный порядок работы системы.

Часть 2

Дополнить базу знаний правилами, позволяющими найти:

1. Максимум из двух чисел без использования отсечения.
2. Максимум из двух чисел с использованием отсечения.
3. Максимум из трёх чисел без использования отсечения.

4. Максимум из трёх чисел с использованием отсечения.

Убедиться в правильности результатов. Для каждого случая пункта 2 обосновать необходимость всех условий тела. Для одного из вариантов ВОПРОСА и каждого варианта задания 2 составить таблицу, отражающую конкретный порядок работы системы.

Для одного из вариантов ВОПРОСА и конкретной БЗ составить таблицу, отражающую конкретный порядок работы системы, с объяснениями: очередная проблема на каждом шаге и метод ее решения; каково новое текущее состояние резольвенты, как получено; какие дальнейшие действия? (Запускается ли алгоритм унификации? Каких термов? Почему этих?) ; вывод по результатам очередного шага и дальнейшие действия.

Решение

Листинг 1 – Листинг программы

```
1 domains
2   name = string.
3   gender = string.
4
5   number = integer.
6
7 predicates
8   parent(name, name, gender).
9   grandParent(name, name, gender, gender).
10
11   two_max(number, number, number).
12   two_max_cut(number, number, number).
13   three_max(number, number, number, number).
14   three_max_cut(number, number, number, number).
15
16 clauses
17   parent("B", "A", "Male").
18   parent("C", "A", "Female").
19   parent("D", "B", "Male").
20   parent("E", "B", "Female").
21   parent("F", "C", "Male").
```

```

22 parent("G", "C", "Female").
23
24 grandParent(GrandParentName, ChildName, GrandParentGender,
    ParentGender):-
25     parent(GrandParentName, ParentName, GrandParentGender),
26     parent(ParentName, ChildName, ParentGender).
27
28 two_max(First, Second, First):- First >= Second.
29 two_max(First, Second, Second):- First < Second.
30
31 two_max_cut(First, Second, First):- First >= Second, !.
32 two_max_cut(_, Second, Second).
33
34 three_max(First, Second, Third, First):- First >= Second, First
    >= Third.
35 three_max(First, Second, Third, Second):- Second >= First, Second
    >= Third.
36 three_max(First, Second, Third, Third) :- Third >= First, Third
    >= Second.
37
38 three_max_cut(First, Second, Third, First):- First >= Second,
    First >= Third, !.
39 three_max_cut(_, Second, Third, Second):- Second >= Third, !.
40 three_max_cut(_, _, Third, Third).
41
42
43
44 goal
45 %grandParent(GrandParentName, "A", "Female", _).
46 %grandParent(GrandParentName, "A", "Male", _).
47 %grandParent(GrandParentName, "A", _, _).
48 %grandParent(GrandParentName, "A", "Female", "Female").
49 %grandParent(GrandParentName, "A", _, "Female").
50 %two_max(1, 2, MAX).
51 %two_max(2, 1, MAX).
52 %two_max_cut(1, 2, MAX).
53 %two_max_cut(2, 1, MAX).
54 %three_max(1, 2, 3, MAX).
55 %three_max(3, 2, 1, MAX).
56 %three_max(1, 3, 2, MAX).
57 %three_max_cut(1, 2, 3, MAX).

```

```
58  %three_max_cut(3, 2, 1, MAX).  
59  %three_max_cut(1, 3, 2, MAX).
```