



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

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 16

Использование правил в программе на Prolog

Дисциплина	Функциональное и логическое программирование
Студент	Сиденко А.Г.
Группа	ИУ7-63Б
Преподаватель	Толпинская Н.Б., Строганов Ю.В.

Москва, 2020 г.

Задание

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

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 и конкретной БЗ составить таблицу, отражающую конкретный порядок работы системы

Программа

```
1 domains
2   name = symbol
3
4 predicates
5   mother(name, name).
6   father(name, name).
7   grandfatherF(name, name).
8   grandfatherM(name, name).
9   grandmotherF(name, name).
10  grandmotherM(name, name).
11  allGrandmothers(name, name).
12  allGrandfathers(name, name).
13  allGrandparents(name, name).
14  grandparentsM(name, name)
15
16 clauses
17   mother(maria, ann).
18   mother(maria, yulia).
19   mother(irina, dima).
20   mother(ann, petr).
```

```

21 father(ivan, ann).
22 father(ivan, yulia).
23 father(maksim, dima).
24 father(dima, petr).
25
26 % Grandfather on the mother line
27 grandfatherM(X, Y):- father(X, Z), mother(Z, Y).
28 % Grandfather on the father line
29 grandfatherF(X, Y):- father(X, Z), father(Z, Y).
30 % Grandmother on the mother line
31 grandmotherM(X, Y):- mother(X, Z), mother(Z, Y).
32 % Grandmother on the father line
33 grandmotherF(X, Y):- mother(X, Z), father(Z, Y).
34
35 % All grandmothers
36 allGrandmothers(X, Y):- grandmotherM(X, Y).
37 allGrandmothers(X, Y):- grandmotherF(X, Y).
38
39 % All grandfathers
40 allGrandfathers(X, Y):- grandfatherM(X, Y).
41 allGrandfathers(X, Y):- grandfatherF(X, Y).
42
43 % All grandparents
44 allGrandparents(X, Y):- allGrandfathers(X, Y).
45 allGrandparents(X, Y):- allGrandmothers(X, Y).
46
47 % Grandparents on the mother line
48 grandparentsM(X, Y):- grandfatherM(X, Y).
49 grandparentsM(X, Y):- grandmotherM(X, Y).

```

Примеры работы:


1. по имени субъекта определить всех его бабушек (предки 2-го колена),

```

1 goal
2   allGrandmothers(X, petr).

```


```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab6\Obj\goal$000.exe]
X=maria
X=irina
2 Solutions|

```

2. по имени субъекта определить всех его дедушек (предки 2-го колена),


```
1 goal
2 allGrandfathers(X, petr).
```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab6\Obj\goal\$000.exe]

```
X=ivan
X=maksim
2 Solutions|
```

3. по имени субъекта определить всех его бабушек и дедушек (предки 2-го колена),


```
1 goal
2 allGrandparents(X, petr).
```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab6\Obj\goal\$000.exe]

```
X=ivan
X=maksim
X=maria
X=irina
4 Solutions|
```

4. по имени субъекта определить его бабушку по материнской линии (предки 2-го колена),


```
1 goal
2 grandmotherM(X, petr).
```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab6\Obj\goal\$000.exe]

```
X=maria
1 Solution|
```

5. по имени субъекта определить его бабушку и дедушку по материнской линии (предки 2-го колена).

```
1 goal
2 grandparentsM(X, petr).
```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab6\Obj\goal\$000.exe]

```
X=ivan
X=maria
2 Solutions|
```

Приведем таблицу для задания 4.

№ шага	Состояние резольвенты	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
1	grandmotherM(X, petr)	По grandmotherM(X, petr) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено, заносится в стек grandmotherM(X, petr), прямой ход
2	mother(X, Z), mother(Z, petr)	Начинает «раскрываться» правило, т.е. доказывается каждое целевое утверждение в теле правила последовательно слева направо mother(X, Z), mother(Z, petr)	Заносится в стек mother(X, Z)
3	mother(X, Z), mother(Z, petr)	По mother(X, Z) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
4	mother(X, Z), mother(Z, petr)	Унификация mother(X, Z) с mother(maria, ann)	Результат сравнения термов: true, X примет значение maria, Z = ann. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
5	mother(ann, petr)	По mother(ann, petr) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
6	mother(ann, petr)	Унификация mother(ann, petr) с mother(maria, ann)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
7	mother(ann, petr)	Унификация mother(ann, petr) с mother(maria, yulia)	Результат сравнения термов: false, переход к следующей строке (прямой ход)

8	mother(ann, petr)	Унификация mother(ann, petr) с mother(irina, dima)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
9	mother(ann, petr)	Унификация mother(ann, petr) с mother(ann, petr)	Результат сравнения термов: true. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
10	Резольвента пустая, успех		В базе знаний больше ни одного утверждения с заданным именем, возврат, достаём из стека mother(ann, petr)
11	mother(X, Z), mother(Z, petr)	Унификация mother(X, Z) с mother(maria, yulia)	Результат сравнения термов: true, X примет значение maria, Z = yulia. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
12	mother(yulia, petr)	По mother(yulia, petr) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
13	mother(yulia, petr)	Унификация mother(yulia, petr) с mother(maria, ann)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
14	mother(yulia, petr)	Унификация mother(yulia, petr) с mother(maria, yulia)	Результат сравнения термов: false, переход к следующей строке (прямой ход)

15	mother(yulia, petr)	Унификация mother(yulia, petr) с mother(irina, dima)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
16	mother(yulia, petr)	Унификация mother(yulia, petr) с mother(ann, petr)	Результат сравнения термов: false, В базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека mother(yulia, petr)
17	mother(X, Z), mother(Z, petr)	Унификация mother(X, Z) с mother(irina, dima)	Результат сравнения термов: true, X примет значение irina, Z = dima. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
18	mother(dima, petr)	По mother(dima, petr) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
19	mother(dima, petr)	Унификация mother(dima, petr) с mother(maria, ann)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
20	mother(dima, petr)	Унификация mother(dima, petr) с mother(maria, yulia)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
21	mother(dima, petr)	Унификация mother(dima, petr) с mother(irina, dima)	Результат сравнения термов: false, переход к следующей строке (прямой ход)

22	mother(dima, petr)	Унификация mother(dima, petr) с mother(ann, petr)	Результат сравнения термов: false, В базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека mother(dima, petr)
23	mother(X, Z), mother(Z, petr)	Унификация mother(X, Z) с mother(ann, petr)	Результат сравнения термов: true, X примет значение ann, Z = petr. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
24	mother(petr, petr)	По mother(petr, petr) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
25	mother(petr, petr)	Унификация mother(petr, petr) с mother(maria, ann)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
26	mother(petr, petr)	Унификация mother(petr, petr) с mother(maria, yulia)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
27	mother(petr, petr)	Унификация mother(petr, petr) с mother(irina, dima)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
28	mother(petr, petr)	Унификация mother(petr, petr) с mother(ann, petr)	Результат сравнения термов: false, В базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека mother(petr, petr)

29		Больше нет правил с заданным именем, достаем из стека grandmotherM(X, petr)	Стек пуст, завершение программы
----	--	---	---------------------------------

Ответы на вопросы

1. В каком случае система запускает алгоритм унификации? (Как эту необходимость на формальном уровне распознает система?)

Пролог выполняет унификацию в двух случаях: когда цель сопоставляется с заголовком предложения или когда используется знак равенства, который является инфиксным предикатом (предикатом, который расположен между своими аргументами, а не перед ними).

2. Каковы назначение и результат использования алгоритма унификации?

Унификация двух термов – это основной шаг доказательства. В процессе работы система выполняет большое число унификаций. **Унификация** – операция, которая позволяет формализовать процесс логического вывода.

Унификация представляет собой процесс сопоставления цели с фактами и правилами базы знаний. Цель может быть согласована, если она может быть сопоставлена с заголовком какого-либо предложения базы.

3. Какое первое состояние резольвенты?

Вопрос.

4. Как меняется резольвента?

Резольвента - текущая цель, существующая на любой стадии вычислений. Резольвенты порождаются целью и каким-либо правилом или фактом, которые просматриваются последовательно сверху вниз. Если резольвента существует при наиболее общей унификации, она вычисляется. Если пустая резольвента с помощью такой стратегии не найдена, то ответ на вопрос отрицателен.

5. В каких пределах программы уникальны переменные?

Областью действия переменной в Прологе является одно предложение. В разных предложениях может использоваться одно имя переменной для обозначения разных объектов. Исключением является анонимная переменная. Каждая анонимная переменная – это отдельный объект.

6. Как применяется подстановка, полученная с помощью алгоритма унификации?

При согласовании переменные получают значения, указанные с другой стороны от знака «=», если переменные еще не были связаны. Переменные становятся связанными и после успешного согласования всех целевых утверждений, будет напечатано значение связанных переменных.

7. В каких случаях запускается механизм отката?

Откат дает возможность получить много решений в одном вопросе к программе.

Во всех точках программы, где существуют альтернативы, в стек заносятся точки возврата.

Если впоследствии окажется, что выбранный вариант не приводит к успеху, то осуществляется откат к последней из имеющихся в стеке точек программы, где был выбран один из альтернативных вариантов.

Выбирается очередной вариант, программа продолжает свою работу. Если все варианты в точке уже были использованы, то регистрируется неудачное завершение и осуществляется переход на предыдущую точку возврата, если такая есть.

При откате все связанные переменные, которые были означены после этой точки, опять освобождаются.