



**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 и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Дисциплина: «Функциональное и логическое
программирование»

Лабораторная работа №16

Студент: Левушкин И. К.

Группа: ИУ7-62Б

Преподаватели: Толпинская Н. Б.,

Строганов Ю. В.

Москва, 2020 г.

Цель работы

Изучить использование правил в программе: структуру, особенности оформления, а также, способ и принципы выполнения таких программ на Prolog.

Задачи работы

Приобрести навыки эффективного декларативного описания предметной области с использованием фактов и правил. Изучить порядок использования фактов и правил в программе на Prolog, принципы и особенности сопоставления и отождествления термов, на основе механизма унификации. Способ формирования и изменения резольвенты. Порядок формирования ответа.

Задание

Ответить на вопросы:

- В каком случае система запускает алгоритм унификации? (Как эту необходимость на формальном уровне распознает система?)
- Каковы назначение и результат использования алгоритма унификации?
- Какое первое состояние резольвенты?
- Как меняется резольвента?
- В каких пределах программы уникальны переменные?
- Как применяется подстановка, полученная с помощью алгоритма унификации?
- В каких случаях запускается механизм отката?

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, 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. Т.к. резольвента хранится в виде стека, то состояние резольвенты требуется отображать в столбик: вершина – сверху! Новый шаг надо начинать с нового состояния резольвенты!

№ шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: $T1=T2$ и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (почему и к чему приводит?)
1...	Комментарий, вывод...
2

Реализация программы

predicates

```
first_ancestors(symbol Child, symbol Father, symbol Mother)
```

```
second_ancestors(symbol Child, symbol Grandfather_f,  
symbol Grandmother_f, symbol Grandfather_m, symbol Grandmother_m)
```

clauses

```
first_ancestors(ilya, kirill, tatiana).  
first_ancestors(kirill, sergei, nadya).  
first_ancestors(tatiana, tolya, luba).
```

```
first_ancestors(vasilisa, yura, sveta).  
first_ancestors(yura, andrei, zina).  
first_ancestors(sveta, kolya, gerda).
```

```
second_ancestors(Child, GF_f, GM_f, GF_m, GM_m) :-  
first_ancestors(Child, Father, Mother),  
first_ancestors(Father, GF_f, GM_f),  
first_ancestors(Mother, GF_m, GM_m).
```

Тесты

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

goal

```
second_ancestors(ilya, _, Grandmother_f, _, Grandmother_m).
```

%Вывод:

```
Grandmother_f=nadya, Grandmother_m=luba
```

```
1 Solution
```

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

goal

```
second_ancestors(ilya, Grandfather_f, _, Grandfather_m, _).
```

%Вывод:

```
Grandfather_f=sergei, Grandfather_m=tolya
```

```
1 Solution
```

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

```
goal
    second_ancestors(ilya, Grandfather_f, Grandmother_f,
        Grandfather_m, Grandmother_m).
%Вывод:
Grandfather_f=sergei, Grandmother_f=nadya,
Grandfather_m=tolya, Grandmother_m=luba
1 Solution
```

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

```
goal
    second_ancestors(ilya, _, _, _, Grandmother_m).
%Вывод:
Grandmother_m=luba
1 Solution
```

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

```
goal
    second_ancestors(ilya, _, _, Grandfather_m, Grandmother_m).
%Вывод:
Grandfather_m=tolya, Grandmother_m=luba
1 Solution
```

Порядок работы системы:

№ ша-га	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: $T1=T2$ и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (почему и к чему приводит?)
---------	---	--	---

1	second_ancestors (ilya, _, _, Grandfather_m, Grandmother_m)	T1 = second_ancestors (ilya, _, _, Grandfather_m, Grandmother_m); T2 = first_ancestors (ilya, kirill, tatiana). Неудача (функто- ры second_ancestors и first_ancestors не равны)	Прямой ход к следую- щему предложению. Аналогичная ситуация в следующих 5 пред- ложениях. Переход к следующему предложе- нию.
7	second_ancestors (ilya, _, _, Grandfather_m, Grandmother_m)	T1 = second_ancestors (ilya, _, _, Grandfather_m, Grandmother_m); T2 = second_ancestors (Child, GF_f, GM_f, GF_m, GM_m). Успех. Подстановка ilya = Child, Grandfather_m = GF_m, Grandmother_m = GM_m.	Прямой ход: first_ancestors(ilya, Father, Mother). Поиск с начала предложений.
8	first_ancestors (ilya, Father, Mother), first_ancestors (Father, _, _), first_ancestors (Mother, GF_m, GM_m).	T1 = first_ancestors (ilya, Father, Mother), T2 = first_ancestors (ilya, kirill, tatiana). Успех. Подстановка ilya=ilya, Father=kirill, Mother=tatiana.	Прямой ход к first_ancestors (kirill, _, _). Прямой ход к first_ancestors (tatiana, GF_m, GM_m). Поиск с начала предложений.
9	first_ancestors (tatiana, GF_m, GM_m).	T1 = first_ancestors (tatiana, GF_m, GM_m), T2 = first_ancestors (ilya, kirill, tatiana). Неудача, tatiana != ilya.	Прямой ход к следую- щему предложению.

10	first_ancestors (tatiana, GF_m, GM_m).	T1 = first_ancestors (tatiana, GF_m, GM_m), T2 = first_ancestors (kirill, sergei, nadya). Неудача, tatiana != kirill.	Прямой ход к следующему предложению.
11	first_ancestors (tatiana, GF_m, GM_m).	T1 = first_ancestors (tatiana, GF_m, GM_m), T2 = first_ancestors (tatiana, tolya, luba). Успех. Подстановка tatiana = tatiana, GF_m = tolya, GM_m = luba.	Вывод: Grandfather_m = tolya, Grandmother_m = luba. Прямой ход к следующему предложению, реконкретизация GF_m, GM_m.
12	first_ancestors (tatiana, GF_m, GM_m).	T1 = first_ancestors (tatiana, GF_m, GM_m), T2 = first_ancestors (vasilisa, yura, sveta). Неудача, tatiana != vasilisa.	Прямой ход к следующему предложению.
13	first_ancestors (tatiana, GF_m, GM_m).	T1 = first_ancestors (tatiana, GF_m, GM_m), T2 = first_ancestors (yura, andrei, zina). Неудача, tatiana != yura.	Прямой ход к следующему предложению.
14	first_ancestors (tatiana, GF_m, GM_m).	T1 = first_ancestors (tatiana, GF_m, GM_m), T2 = first_ancestors (sveta, kolya, gerda). Неудача, tatiana != sveta.	Прямой ход к следующему предложению.

15	first_ancestors (tatiana, GF_m, GM_m).	T1 = first_ancestors (tatiana, GF_m, GM_m), T2 = second_ancestors (Child, GF_f, GM_f, GF_m, GM_m). Неудача (функции first_ancestors и second_ancestors не равны).	Откат, переход к предыдущему состоя- нию резольвенты (шаг 8), реконкретизация Father, Mother.
16	first_ancestors (ilya, Father, Mother), first_ancestors (Father, _, _), first_ancestors (Mother, GF_m, GM_m).	T1 = first_ancestors (ilya, Father, Mother), T2 = first_ancestors (kirill, sergei, nadya). Неудача, ilya != kirill.	Прямой ход к следую- щему предложению.
17	first_ancestors (ilya, Father, Mother), first_ancestors (Father, _, _), first_ancestors (Mother, GF_m, GM_m).	T1 = first_ancestors (ilya, Father, Mother), T2 = first_ancestors (tatiana, tolya, luba). Неудача, ilya != tatiana.	Прямой ход к следую- щему предложению.
18	first_ancestors (ilya, Father, Mother), first_ancestors (Father, _, _), first_ancestors (Mother, GF_m, GM_m).	T1 = first_ancestors (ilya, Father, Mother), T2 = first_ancestors (vasilisa, yura, sveta). Неудача, ilya != vasilisa.	Прямой ход к следую- щему предложению.

19	first_ancestors (ilya, Father, Mother), first_ancestors (Father, _, _), first_ancestors (Mother, GF_m, GM_m).	T1 = first_ancestors (ilya, Father, Mother), T2 = first_ancestors (yura, andrei, zina). Неудача, ilya != yura.	Прямой ход к следующему предложению.
20	first_ancestors (ilya, Father, Mother), first_ancestors (Father, _, _), first_ancestors (Mother, GF_m, GM_m).	T1 = first_ancestors (ilya, Father, Mother), T2 = first_ancestors (sveta, kolya, gerda). Неудача, ilya != sveta.	Прямой ход к следующему предложению.
21	first_ancestors (ilya, Father, Mother), first_ancestors (Father, _, _), first_ancestors (Mother, GF_m, GM_m).	T1 = first_ancestors (ilya, Father, Mother), T2 = second_ancestors (Child, GF_f, GM_f, GF_m, GM_m). Неудача (функции first_ancestors и second_ancestors не равны).	Откат, переход к предыдущему состоянию резольвенты (шаг 7). Реконкретизация Grandfather_m, Grandmother_m.
22	second_ancestors (ilya, _, _, Grandfather_m, Grandmother_m); конец clauses; опустошение резольвенты; завершение работы.		

Ответы на вопросы

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

Процесс унификации запускается, если есть цель, которую необходимо доказать (формально: если резольвента не пуста).

Каковы назначение и результат использования алгоритма унификации

Назначение алгоритма унификации заключается в попарном сопоставлении термов и попытке построить для них общий пример.

Результатом использования алгоритма унификации может быть успех или тупиковая ситуация (неудача).

Какое первое состояние резольвенты?

Если задан простой вопрос, то сначала он попадает в резольвенту.

Как меняется резольвента?

Изменение резольвенты происходит в 2 этапа:

1. из стека выбирается подцель и для нее выполняется редукция, т.е. замена подцели на тело найденного правила;
2. к полученной конъюнкции целей применяется подстановка (наибольший общий унификатор выбранной цели и заголовка сопоставленного с этой целью правила).

В каких пределах программы уникальны переменные?

Переменные уникальны в пределах предложения.

Исключение – анонимные переменные – каждая такая переменная является отдельной сущностью и применяется, когда ее значение неважно для данного предложения.

Как применяется подстановка, полученная с помощью алгоритма унификации?

Применение подстановки $x_1 = t_1, \dots, x_n = t_n$ заключается в замене каждого вхождения переменной x_i на соответствующий терм t_i .

В каких случаях запускается механизм отката?

Механизм отката запускается, если возникла тупиковая ситуация (достигнут конец БЗ) либо резольвента пуста. В таких случаях происходит откат к предыдущему состоянию резольвенты.

Исправление ошибок 14-ой лабораторной работы.

Порядок работы алгоритма унификации 2-термов??? А это что???

Имелось в виду: *Порядок работы алгоритма унификации вопроса и подходящего заголовка правила при изменении порядка следования в Базе Знаний процедур и знаний в них.*

Исправить следующую запись в таблице порядка работы алгоритма унификации вопроса и подходящего заголовка правила:

```
get_info_by_model_color(bugatti, gold, Surname, Phone, Bank)=  
get_info_by_mode_color(Model, Color, Surname, Phone, Bank):- auto(Surname,  
Model, Color, _, _),  
bank_list(Surname,Bank, _, _),phone_list(Surname,Phone, _).
```

Ниже приведена исправленная таблица:

№ ша- га	Результирующая ячейка	Рабочее поле	П. алг.	Стек
0			1	get_info_by_model_color(bugatti, gold, Surname, Phone, Bank)=get_info_by_mode_color(Model, Color, Surname, Phone, Bank).

1		get_info_by_ model_color(bugatti, gold, Surname, Phone, Bank)=get_info_by_ mode_color(Model, Color, Surname, Phone, Bank).	e	auto(Surname, bugatti, gold, __, __) = auto(samkov, bugatti, gold, __, __), auto(Surname, bugatti, gold, __, __) = auto(ryazanova, bugatti, gold, __, __), bank_list(Surname, Bank, __, __), phone_list(Surname, Phone, __, __).
2	Surname = samkov, = Surname = ryazanova	auto(Surname, bugatti, gold, __, __) = auto(samkov, bugatti, gold, __, __), auto(Surname, bugatti, gold, __, __) = auto(ryazanova, bugatti, gold, __, __).	e	bank_list(samkov, Bank, __, __) = bank_list(samkov, sberbank, __, __), bank_list(ryazanova, Bank, __, __) = bank_list(ryazanova, tinkoff, __, __), bank_list(ryazanova, Bank, __, __) = bank_list(ryazanova, raiffeisen, __, __), phone_list(samkov, Phone, __, __), phone_list(ryazanova, Phone, __, __).
3	Surname = samkov, = Bank = sberbank, = Surname = ryazanova, = Bank = tinkoff, Bank = raiffeisen.	bank_list(samkov, Bank, __, __) = bank_list(samkov, sberbank, __, __), bank_list(ryazanova, Bank, __, __) = bank_list(ryazanova, tinkoff, __, __), bank_list(ryazanova, Bank, __, __) = bank_list(ryazanova, raiffeisen, __, __).	e	phone_list(samkov, Phone, __, __) = phone_list(samkov, «89899999», __, __), phone_list(ryazanova, Phone, __, __) = phone_list(ryazanova, «8911911911», __, __).

4	Surname = samkov, Bank = sberbank, Surname = ryazanova, Bank = tinkoff, Bank = raiffeisen.	phone_list(samkov, Phone, _, _) = phone_list(samkov, «89899999», _, _), phone_list(ryazanova, Phone, _, _) = phone_list(ryazanova, «8911911911», _, _).	e	
Вы- вод:	подстановка	Т. к. стек пуст - успех и в рез. ячейнке подста- новка		

Работа алгоритма унификации заключается в попарном сопоставлении термов и попытке построить для них общий пример ЗАЧЕМ???

Унификация двух термов является основным шагом доказательства, назначение которого – подобрать нужное в данный момент правило.