



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»

Лабораторная работа №16

*По предмету: «Функциональное и логическое
программирование»*

Преподаватель: Строганов Ю.В.
Студент: Гасанзаде М.А.,
Группа: ИУ7-66Б

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, 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...
...

Теоретические вопросы.

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

Если резольвента не пуста – запускается алгоритм унификации.

2. Каковы назначение и результат использования алгоритма унификации?

Алгоритм унификации попарно сопоставляет термы в попытке построить для них общий пример. Унификация может завершаться успехом или тупиковой ситуацией (неудачей).

3. Какое первое состояние резольвенты?

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

4. Как меняется резольвента?

Изменение резольвенты происходит в 2 этапа:

- 1) Из стека выбирается верхняя подцель и для нее выполняется редукция (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).

5. В каких пределах программы уникальны переменные?

Именованные переменные уникальны в пределах предложения. В случае с анонимными переменными каждая такая переменная является отдельной сущностью и применяется, когда ее значение неважно для данного предложения.

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

Заменяется каждое вхождение переменной x_i на соответствующий терм t_i . $\{x_1=t_1, \dots, x_n=t_n\}$

7. В каких случаях запускается механизм отката?

Если достигнут конец БЗ (тупиковая ситуация) или резольвента пуста.

После этого происходит откат к предыдущему состоянию резольвенты.

Текст программы

```
1.predicates
2.    person(symbol Name, symbol Mother, symbol Father).
3.    grandparent(symbol Name, symbol Father_f, symbol Mother_f,symbol
    Father_m, symbol Mother_m).
4.clauses
5.    person(igor, ira, vova).
6.
7.    person(sasha, maria, nikita).
8.    person(katya, lena, roma).
9.    person(andrey, katya, sasha).
10.
11.        person(stas, elena, kostya).
12.        person(inna, kseniya, egor).
13.        person(kostya, inna, stas).
14.
15.
16.        grandparent(Name, Father_f, Mother_f, Father_m, Mother_m) :-
17.            person(Name, Mother, Father),
18.            person(Mother, Mother_m, Father_m ),
19.            person(Father, Mother_f, Father_f).
20.    goal
21.        %1. All grandmothers
22.        %grandparent(andrey, _, Mother_f, _, Mother_m).
23.        %2. All grandfathers
24.        %grandparent(andrey, Father_f, _, Father_m, _).
25.        %3. All grandparent
26.        %grandparent(andrey, Father_f, Mother_f, Father_m, Mother_m).
27.        %4. Mother's mother
28.        %grandparent(andrey, _, _, _, Mother_m).
29.        %5. Mother's parents
30.        %grandparent(andrey, _, _, Father_m, Mother_m).
```

Пример работы программы:

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

`grandparent (andrey, _, Mother_f, _, Mother_m) .`

`Mother_f=maria, Mother_m=lena`

`1 Solution`

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

```
grandparent(andrey, Father_f, _, Father_m, _).
```

```
Father_f=nikita, Father_m=roma  
1 Solution
```

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

```
grandparent(andrey, Father_f, Mother_f, Father_m, Mother_m).
```

```
Father_f=nikita, Mother_f=maria, Father_m=roma, Mother_m=lena  
1 Solution
```

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

```
grandparent(andrey, _, _, Mother_m).
```

```
Mother_m=lena  
1 Solution
```

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

```
grandparent(andrey, _, _, Father_m, Mother_m).
```

```
Father_m=roma, Mother_m=lena  
1 Solution
```

Описание порядка работы системы

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

№ шага	Состояния 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	<pre>grandparent(andrey, Father_f, _, Father_m, _)</pre>	Попытка унификации: <pre>grandparent(andrey, Father_f, _, Father_m, _) = person(alexei, elena, vladimir)</pre> Результат: Неудача (разные функторы)	Прямой ход, переход к следующему предложению.
2	<pre>grandparent(andrey, Father_f, _, Father_m, _)</pre>	Попытка унификации: <pre>grandparent(andrey, Father_f, _, Father_m, _) = person(inna, kseniya, egor)</pre> Результат: Неудача (разные функторы)	Прямой ход, переход к следующему предложению.
3	<pre>grandparent(andrey, Father_f, _, Father_m, _)</pre>	Попытка унификации: <pre>grandparent(andrey, Father_f, _, Father_m, _) =</pre>	Прямой ход, переход к следующему предложению

		<code>person(vladimir, inna, alexey)</code> Результат: Неудача (разные функторы)	
4	<code>grandparent(andrey, Father_f, _, Father_m, _)</code>	Попытка унификации: <code>grandparent(andrey, Father_f, _, Father_m, _)</code> <code>=</code> <code>person(sasha, maria, nikita)</code> Результат: Неудача (разные функторы)	Прямой ход, переход к следующему предложению
5	<code>grandparent(andrey, Father_f, _, Father_m, _)</code>	Попытка унификации: <code>grandparent(andrey, Father_f, _, Father_m, _)</code> <code>=</code> <code>person(katya, lena, roma)</code> Результат: Неудача (разные функторы)	Прямой ход, переход к следующему предложению
6	<code>grandparent(andrey, Father_f, _, Father_m, _)</code>	Попытка унификации: <code>grandparent(andrey, Father_f, _, Father_m, _)</code> <code>=</code> <code>person(andrey, katya, sasha)</code> Результат: Неудача (разные функторы)	Прямой ход, переход к следующему предложению
7	<code>grandparent(andrey, Father_f, _, Father_m, _)</code>	Попытка унификации: <code>grandparent(andrey, Father_f, _, Father_m, _)</code> <code>=</code> <code>grandparent(Child, Father_f, Mother_f, Father_m, Mother_m)</code> Результат: Успех (подстановка) <code>{Child = andrey, Father_f = Father_f, Father_m = Father_m}</code>	Редукция цели, прямой ход к сопоставлению <code>person(Child, Father, Mother)</code> , поиск с начала предложений.
8	<code>person(andrey, Father, Mother),</code> <code>person(Father, Father_f, _),</code> <code>person(Mother, Father_m, _)</code>	Попытка унификации: <code>person(andrey, Father, Mother)</code> <code>=</code> <code>person(alexey, elena, vladimir)</code> Результат: Неудача (<code>andrey <> alexey</code>)	Прямой ход, переход к следующему предложению
9	<code>person(andrey, Father, Mother),</code> <code>person(Father, Father_f, _),</code> <code>person(Mother, Father_m, _)</code>	Попытка унификации: <code>person(andrey, Father, Mother)</code> <code>=</code> <code>person(inna, egor, kseniya)</code> Результат: Неудача (<code>andrey <> inna</code>)	Прямой ход, переход к следующему предложению

10	<pre> person(andy, Father, Mother), person(Father, Father_f, _), person(Mother, Father_m, _) </pre>	<p>Попытка унификации:</p> <pre> person(andy, Father, Mother) = person(vladimir, inna, alexey) </pre> <p>Результат: Неудача (andy <> vladimir)</p>	Прямой ход, переход к следующему предложению
11	<pre> person(andy, Father, Mother), person(Father, Father_f, _), person(Mother, Father_m, _) </pre>	<p>Попытка унификации:</p> <pre> person(andy, Father, Mother) = person(sasha, maria, nikita) </pre> <p>Результат: Неудача (andy <> sasha)</p>	Прямой ход, переход к следующему предложению
12	<pre> person(andy, Father, Mother), person(Father, Father_f, _), person(Mother, Father_m, _) </pre>	<p>Попытка унификации:</p> <pre> person(andy, Father, Mother) = person(katya, lena, roma) </pre> <p>Результат: Неудача (andy <> katya)</p>	Прямой ход, переход к следующему предложению
13	<pre> person(andy, Father, Mother), person(Father, Father_f, _), person(Mother, Father_m, _) </pre>	<p>Попытка унификации:</p> <pre> person(andy, Father, Mother) = person(andy, katya, sasha) </pre> <p>Результат: Успех (подстановка) {Father = sasha, Mother = katya}</p>	Прямой ход к сопоставлению person(sasha, Father_f, _), поиск с начала предложений.
14	<pre> person(Father, Father_f, _), person(Mother, Father_m, _) </pre>	<p>Попытка унификации:</p> <pre> person(sasha, Father_f, _) = person(alexey, elena, vladimir) </pre> <p>Результат: Неудача (sasha <> alexey)</p>	Прямой ход, переход к следующему предложению
15	<pre> person(Father, Father_f, _), person(Mother, Father_m, _) </pre>	<p>Попытка унификации:</p> <pre> person(sasha, Father_f, _) = person(inna, kseniya, egor) </pre> <p>Результат: Неудача (sasha <> inna)</p>	Прямой ход, переход к следующему предложению
16	<pre> person(Father, Father_f, _), person(Mother, Father_m, _) </pre>	<p>Попытка унификации:</p> <pre> person(sasha, Father_f, _) = person(vladimir, inna, alexey) </pre> <p>Результат: Неудача (sasha <> vladimir)</p>	Прямой ход, переход к следующему предложению

17	<pre> person(Father, Father_f, _), person(Mother, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre> person(sasha, Father_f, _) = person(sasha, maria, nikita)</pre> <p>Результат: Успех (подстановка) {Father_f = kirill}</p>	<p>Прямой ход к сопоставлению person(katya, Father_m, _), поиск с начала предложений.</p>
18	<pre> person(katya, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre> person(katya, Father_m, _) = person(alexeу, elena, vladimir)</pre> <p>Результат: Неудача (katya <> alexey)</p>	<p>Прямой ход, переход к следующему предложению</p>
19	<pre> person(katya, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre> person(katya, Father_m, _) = person(inna, kseniya, egor)</pre> <p>Результат: Неудача (katya <> inna)</p>	<p>Прямой ход, переход к следующему предложению</p>
20	<pre> person(katya, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre> person(katya, Father_m, _) = person(vladimir, inna, alexey)</pre> <p>Результат: Неудача (katya <> vladimir)</p>	<p>Прямой ход, переход к следующему предложению</p>
21	<pre> person(katya, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre> person(katya, Father_m, _) = person(sasha, maria, nikita)</pre> <p>Результат: Неудача (katya <> sasha)</p>	<p>Прямой ход, переход к следующему предложению</p>
22	<pre> person(katya, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre> person(katya, Father_m, _) = person(katya, lena, roma)</pre> <p>Результат: Успех (подстановка) {Father_m = andrey}</p>	<p>Вывод: Father_f = kirill, Father_m = andrey.</p> <p>Опустошение резольвенты вследствие успешной унификации.</p> <p>Откат, переход к предыдущему состоянию резольвенты. Реконкретизация Father_m.</p>
23	<pre> person(katya, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre> person(katya, Father_m, _) = person(andrey, katya, sasha)</pre> <p>Результат: Неудача (katya <> andrey)</p>	<p>Прямой ход к следующему предложению.</p>

24	<pre>person(katya, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre>person(katya, Father_m, _) = grandparent(Child, Father_f, Mother_f, Father_m, Mother_m)</pre> <p>Результат: Неудача (разные функторы)</p>	Конец БЗ, откат, переход к предыдущему состоянию резольвенты (шаг 17), реконкретизация Father_f.
25	<pre>person(Father, Father_f, _), person(Mother, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre>person(sasha, Father_m, _) = person(katya, lena, roma)</pre> <p>Результат: Неудача (sasha <> katya)</p>	Прямой ход к следующему предложению.
26	<pre>person(Father, Father_f, _), person(Mother, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre>person(sasha, Father_m, _) = person(andrey, katya, sasha)</pre> <p>Результат: Неудача (sasha <> andrey)</p>	Прямой ход к следующему предложению.
27	<pre>person(Father, Father_f, _), person(Mother, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre>person(sasha, Father_m, _) = grandparent(Child, Father_f, Mother_f, Father_m, Mother_m)</pre> <p>Результат: Неудача (разные функторы)</p>	Конец БЗ, откат, переход к предыдущему состоянию резольвенты (шаг 13), реконкретизация Father и Mother.
28	<pre>person(andrey, Father, Mother), person(Father, Father_f, _), person(Mother, Father_m, _)</pre>	<p>Попытка унификации:</p> <pre>person(andrey, Father, Mother) = grandparent(Child, Father_f, Mother_f, Father_m, Mother_m)</pre> <p>Результат: Неудача (разные функторы)</p>	Откат, переход к предыдущему состоянию резольвенты (шаг 7).
29	<pre>grandparent(andrey, Father_f, _, Father_m, _)</pre> <p>конец clauses; опустошение резольвенты; завершение работы.</p>		