



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 № 6

По дисциплине «Функциональное и логическое программирование»

Студент: Тимонин А. С.

Группа ИУ7-626

Преподаватель Толпинская Н. Б.

Москва.
2020 г.

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. колена),
5. по имени субъекта определить его бабушку по материнской линии (предки 2-го колена),
6. 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!

Практическая часть

Листинг 1. Реализация программы

```
domains
    firstname, lastname = symbol.
    children, parent, predok = person(firstname, lastname).

predicates
    nondeterm father(children, parent).
    nondeterm father(parent, predok).
    nondeterm mother(children, parent).
    nondeterm mother(parent, predok).

    nondeterm allGrandMa(children, predok).
    nondeterm allGrandDa(children, predok).
    nondeterm allGrand(children, predok).
    nondeterm mothersMama(children, predok).
    nondeterm mothersPapa(children, predok).
    nondeterm mothersMaPa(children, predok).

clauses
    father(person("Anton", "Lomin"), person("Sergio", "Lomin")).
    father(person("Sergio", "Lomin"), person("Gena", "Lomin")).
    father(person("Egor", "Lomin"), person("Sergio", "Lomin")).
    father(person("Evgeny", "Lomin"), person("Sergio", "Lomin")).
    father(person("Mir", "Lomina"), person("Evgeny", "Lomin")).
    father(person("Olga", "Lomina"), person("Sasha", "Spyi")).

    mother(person("Mira", "Lomina"), person("Valerya", "Lobor")).
    mother(person("Anton", "Lomin"), person("Olga", "Lomina")).
    mother(person("Evgeny", "Lomin"), person("Olga", "Lomina")).
    mother(person("Egor", "Lomin"), person("Olga", "Lomina")).
    mother(person("Sergio", "Lomin"), person("Valya", "Lomina")).
    mother(person("Olga", "Lomina"), person("Tamara", "Spyi")).

    allGrandMa(Children, Babushka) :-
        father(Children, Father),
        mother(Father, Babushka).

    allGrandMa(Children, Babushka) :-
        mother(Children, Mother),
        mother(Mother, Babushka).

    allGrandDa(Children, Dedushka) :-
        father(Children, Papa),
        father(Papa, Dedushka).

    allGrandDa(Children, Dedushka) :-
        mother(Children, Mama),
        father(Mama, Dedushka).

    allGrand(Children, Babushka) :-
        allGrandMa(Children, Babushka).

    allGrand(Children, Dedushka) :-
        allGrandDa(Children, Dedushka).
```

```

mothersMama(Children, BabushkaForMa) :-
    mother(Children, Mama),
    mother(Mama, BabushkaForMa).

mothersPapa(Children, DedushkaForMa) :-
    mother(Children, Mama),
    father(Mama, DedushkaForMa).

mothersMaPa(Children, Babushka) :-
    mothersMama(Children, Babushka).

mothersMaPa(Children, Dedushka) :-
    mothersPapa(Children, Dedushka).


```

goal

```

%allGrandMa(person("Anton", "Lomin"), Grandma).
%allGrandDa(person("Anton", "Lomin"), Dedushka).
%allGrand(person("Anton", "Lomin"), GrandMaPa).
%mothersMama(person("Anton", "Lomin"), BabushkaForMa).
mothersMaPa(person("Anton", "Lomin"), GrandMaPaForMothersLine).

```

 [Inactive C:\Users\ANTONT~1\AppData\Local\Temp\goal\$000.exe]

```

Grandma=person("Valya","Lomina")
Grandma=person("Tamara","Spyi")
2 Solutions|

```

Рисунок 1. allGrandMa(person("Anton", "Lomin"), Grandma).

```
[Inactive C:\Users\ANTONT~1\AppData\Local\Temp\goal$000.exe]
Dedushka=person("Gena","Lomin")
Dedushka=person("Sasha","Spyi")
2 Solutions|
```

Рисунок 2. allGrandDa(person("Anton", "Lomin"), Dedushka).

```
[Inactive C:\Users\ANTONT~1\AppData\Local\Te
GrandMaPa=person("Valya","Lomina")
GrandMaPa=person("Tamara","Spyi")
GrandMaPa=person("Gena","Lomin")
GrandMaPa=person("Sasha","Spyi")
4 Solutions|
```

Рисунок 3. allGrand(person("Anton", "Lomin"), GrandMaPa).

```
[Inactive C:\Users\ANTONT~1\AppData\Local\Temp\goal$000.exe]
BabushkaForMa=person("Tamara","Spyi")
1 Solution|
```

Рисунок 4. mothersMama(person("Anton", "Lomin"), BabushkaForMa).

```

[Inactive C:\Users\ANTONT~1\AppData\Local\Temp\g
Kek=person("Tamara","Spyi")
Kek=person("Sasha","Spyi")
2 Solutions|

```

Рисунок 5. mothersMaPa(person("Anton", "Lomin"), GrandMaPaForMothersLine).

Формирование ответа

Таблица 1. mothersMama(person("Anton", "Lomin"), BabushkaForMa).

№ 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	mother(Children, Mama), mother(Mama, BabushkaForMa)	Подстановка Children = person(«Anton», «Lomin») BabushkaForMa = BabushkaForMa	Прямой ход
2	mother(Mama, BabushkaForMa)	Сравнение person(«Anton», «Lomin») и person(«Mira», «Lomina») mother(person("Mira", "Lomina"), person("Valerya", «Lobor»)) mother(Children, Mama) Нет	Прямой ход
3	mother(Mama, BabushkaForMa)	Сравнение person(«Anton», «Lomin») и person(«Anton», «Lomin») mother(person("Anton", "Lomin"), person("Olga", "Lomina")) mother(Children, Mama) Успех	Прямой ход
4	mother(Mama, BabushkaForMa)	Подстановка Mama = person("Olga", "Lomina")	Прямой ход

5	Пусто	Сравнение person("Olga", "Lomina") и person("Mira", «Lomina») mother(person("Mira", "Lomina"), person("Valerya", «Lobor»)) mother(Mama, BabushkaForMa) Нет	Прямой ход
6	Пусто	Сравнение person("Olga", "Lomina") и person("Anton", «Lomin») mother(person("Anton", "Lomin"), person("Olga", «Lomina»)) mother(Mama, BabushkaForMa) Нет	Прямой ход
7	Пусто	Сравнение person("Olga", "Lomina") и person("Evgeny", «Lomin») mother(person("Evgeny", "Lomin"), person("Olga", «Lomina»)) mother(Mama, BabushkaForMa) Нет	Прямой ход
8	Пусто	Сравнение person("Olga", "Lomina") и person("Egor", «Lomin») mother(person("Egor", "Lomin"), person("Olga", "Lomina")) mother(Mama, BabushkaForMa) Нет	Прямой ход
9	Пусто	Сравнение person("Olga", "Lomina") и person("Sergio", «Lomin») mother(person("Sergio", "Lomin"), person("Valya", "Lomina")) mother(Mama, BabushkaForMa) Нет	Прямой ход
10	Пусто	Сравнение person("Olga", "Lomina") и person("Olga", «Lomina») mother(person("Olga", "Lomina"), person("Tamara", "Spyi")) mother(Mama, BabushkaForMa) Успех	Прямой ход
11	Пусто	Подстановка BabushkaForMa = person("Tamara", "Spyi")	Прямой ход
12	Пусто	Результат BabushkaForMa = person(«Tamara», «Spyi»)	Обратный ход

13	mother(Mama, BabushkaForMa)	<p>Сравнение person(«Anton», «Lomin») и person(«Evgeny», «Evgeny»)</p> <p>mother(person("Evgeny", "Lomin"), person("Olga", «Lomina")) mother(Children, Mama)</p> <p>Нет</p>	Прямой ход
14	mother(Mama, BabushkaForMa)	<p>Сравнение person(«Anton», «Lomin») и person(«Egor», «Lomin»)</p> <p>mother(person("Egor", "Lomin"), person("Olga", «Lomina")) mother(Children, Mama)</p> <p>Нет</p>	Прямой ход
15	mother(Mama, BabushkaForMa)	<p>Сравнение person(«Anton», «Lomin») и person(« Sergio», «Lomin»)</p> <p>mother(person("Sergio", "Lomin"), person("Valya", «Lomina")) mother(Children, Mama)</p> <p>Нет</p>	Прямой ход
16	mother(Mama, BabushkaForMa)	<p>Сравнение person(«Anton», «Lomin») и person(«Olga», «Lomina»)</p> <p>mother(person("Olga", "Lomina"), person("Tamara", «Spyi")) mother(Children, Mama)</p> <p>Нет</p>	Обратный ход

Теоретическая часть

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

Алгоритм унификации запускается в двух случаях:

1. когда цель сопоставима с заголовком предложения,
2. когда используется знак равенства, являющимся предикатом, расположенным между переменными.

Каковы назначение и результат использования алгоритма унификации?

Унификация – операция, которая позволяет формализовать процесс логического вывода. Алгоритм унификации сопоставляет подцель с заданной переменной.

```
1 goal
2     a = man("Anton", Surname).
3     a = man(Name, "Timonin")
```

Например, если в строке 2, в переменной a Name сопоставится с «Anton», а в строке 3, переменная Surname сопоставится с «Timonin». Если бы в строке 2 мы поставили бы фамилию, например, «Konin», тогда строка 3 выдала бы ошибку, так как «Konin» не равно «Timonin». Это связано с тем, что когда параметр в какой-либо переменной занят, он перестает сопоставлять переменные, а начинает их сравнивать.

Какое первое состояние резольвенты?

Вопрос

Как меняется резольвента?

Состояние резольвенты меняется в процессе доказательства (Для хранения резольвенты система использует стек). Успешное завершение работы программы достигается тогда, когда резольвента пуста. Преобразования резольвенты выполняются с помощью редукции.

Редукцией цели G с помощью программы P называется замена цели G телом того правила из P , заголовок которого унифицируется с целью. Новая резольвента образуется в два этапа

1. в текущей резольвенте выбирается одна из подцелей (по стековому принципу - верхняя) и для неё выполняется редукция - замена подцели на тело найденного (подобранного, если удалось) правила (а как подбирается правило?),
2. затем, к полученной конъюнкции целей применяется подстановка, полученная как наибольший общий унификатор цели (выбранной) и заголовка сопоставленного с ней правила.

В каких пределах программы уникальны переменные?

Именованные переменные уникальны в рамках предложения, а анонимная переменная – любая уникальна. В разных предложениях может использоваться одно имя переменной для обозначения разных объектов.

Как применяется подстановка, полученная с помощью алгоритма унификации?

Применение подстановки заключается в замене каждого вхождения переменной x_i на соответствующий терм. Если в результате алгоритма унификации случился успех, тогда переменная связывается со значением, которое успешно унифицировано.

В каких случаях запускается механизм отката?

Когда в программе возможен выбор нескольких вариантов, Пролог заносит в стек точку возврата, для последующего отката по этой точке возврата.

Пролог унифицирует выбранный вариант, если унификация прошла успешно, тогда пролог подготавливает ответ, и далее по точке возврата происходит унификация с другими вариантами. Если пролог не видит дальнейшие

варианты, которые он мог бы проунифицировать, тогда по точке возврата программа возвращается на еще более раннюю стадию.