



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»

ОТЧЕТ

по Лабораторной работе №9

по курсу «Функциональное и логическое программирование»

на тему: «Структура программы на Prolog»

Студент ИУ7-63Б
(Группа)

(Подпись, дата)

Миронов Г. А.
(И. О. Фамилия)

Преподаватель

(Подпись, дата)

Толшинская Н. Б.
(И. О. Фамилия)

2022 г.

1 Практическая часть

Задание 12: составить программу, то есть модель предметной области – базу знаний, объединив в ней информацию – знания:

- «Телефонный справочник»: Фамилия, №тел, Адрес – структура (Город, Улица, №дома, №кв)
- «Автомобили»: Фамилия_владельца, Марка, Цвет, Стоимость и др.
- «Вкладчики банков»: Фамилия, Банк, счет, сумма, др.

Владелец может иметь несколько телефонов, автомобилей, вкладов (Факты). Используя правила, обеспечить возможность поиска:

1. А. По № телефона найти: Фамилию, Марку автомобиля, Стоимость автомобиля (может быть несколько)
В. Используя сформированное в предыдущем пункте правило, по №телефона найти только Марку автомобиля (автомобилей может быть несколько)
2. Используя простой, не составной вопрос: по Фамилии (уникальна в городе, но в разных городах есть однофамильцы) и Городу проживания найти: Улицу, проживания, Банки, в которых есть вклады и №телефона.

Для задания 1 и 2 для одного из вариантов ответов, и для А. и для В., описать словесно порядок поиска ответа на вопрос, указав, как выбираются знания, и, при этом, для каждого этапа унификации, выписать подстановку – наибольший общий унификатор, и соответствующие примеры термов.

Листинг 1.1 – Реализация базы знаний, Часть 1

```
1 domains
2     surname = string
3     city, street = string
4     house, flat = integer
5     phone = string
6     address = addr(city, street, house, flat)
7     mark = string
8     color = string
9     price = integer
10    bank = string
11    id, amount = integer
12
13 predicates
14     person(surname, phone, address)
15     car(surname, mark, color, price)
16     bank_depositor(surname, bank, id, amount)
17     car_by_phone(phone, surname, mark, price)
18     only_mark_by_phone(phone, mark)
19     data_by_surname_and_city(surname, city, street, bank, phone)
20
21 clauses
22     person("Andreev", "+79999999999", addr("Moscow", "Lesnaya", 12,
23         2)).
24     person("Borisov", "+71111111111", addr("Moscow", "Lesnaya", 13,
25         87)).
26     person("Dmitriev", "+73333333333", addr("Ekaterinburg", "
27         Kamennaya", 13, 87)).
28     person("Fedorov", "+66666666666", addr("Moscow", "Hospital'naya
29         ", 123, 87)).
30
31     car("Andreev", "bmw", "green", 1000).
32     car("Andreev", "volkswagen", "red", 10000).
33     car("Dmitriev", "lada", "black", 20000).
34     car("Fedorov", "opel", "white", 10).
35
36     bank_depositor("Andreev", "Sber", 22, 1000).
37     bank_depositor("Borisov", "Sber", 33, 10000).
38     bank_depositor("Dmitriev", "Alfa", 44, 20000).
39     bank_depositor("Fedorov", "Sper", 238, 10).
40     bank_depositor("Borisob", "Maze", 1, 10000).
```

Листинг 1.2 – Реализация базы знаний, Часть 1

```
38 car_by_phone(Phone, Surname, Mark, Price) :-  
39     person(Surname, Phone, _),  
40     car(Surname, Mark, _, Price).  
41  
42 only_mark_by_phone(Phone, Mark) :- car_by_phone(Phone, _, Mark,  
43     _).  
44  
44 data_by_surname_and_city(Surname, City, Street, Bank, Phone) :-  
45     person(Surname, Phone, addr(City, Street, _, _)),  
46     bank_depositor(Surname, Bank, _, _).  
47  
48 goal  
49     %car_by_phone("+79999999999", Surname, Mark, Price).  
50     only_mark_by_phone("+79999999999", Mark).  
51     %data_by_surname_and_city("Borisov", "Moscow", Street, Bank,  
52         Phone).
```

В Таблицах 1.1-1.2 представлен порядок поиска ответа на вопрос для запросов 1 и 2.

Таблица 1.1 – Порядок формирования результата для 1-го вопроса

Шаг	Сравниваемые термы; результаты	Дальнейшие действия	Резольвента	Подстановка
1	car_by_phone("+79999999999", Surname, Mark, Price) и person("Andreev", "+79999999999", addr("Moscow", "Lesnaya", 12, 2)). Главные функторы не равны	Прямой ход Переход к след. предл.	car_by_phone("+79999999999", Surname, Mark, Price)	
...
5	car_by_phone("+79999999999", Surname, Mark, Price) и person("bmw", "green", 1000). Главные функторы не равны	Прямой ход Переход к след. предл.	car_by_phone("+79999999999", Surname, Mark, Price)	
...
9	car_by_phone("+79999999999", Surname, Mark, Price) и bank_depositor("Andreev", "Sber", 22, 1000). Главные функторы не равны	Прямой ход Переход к след. предл.	car_by_phone("+79999999999", Surname, Mark, Price)	
...
Продолжение на следующей странице				

Таблица 1.1 – продолжение

Шаг	Сравниваемые термины; результаты	Дальнейшие действия	Результата	Подстановка
14	car_by_phone("+79999999999", Surname, Mark, Price) и car_by_phone(Phone, Surname, Mark, Price).	Прямой ход унификация person(Surname, "+79999999999", _)	person(Surname, "+79999999999", _), car(Surname, Mark, _, Price)	Phone = "+79999999999"
15	person(Surname, "+79999999999", _) и person("Andreev", "+79999999999", addr("Moscow", "Lesnaya", 12, 2))	Прямой ход унификация car("Andreev", Mark, _, Price)	car(Surname, Mark, _, Price)	Phone = "+79999999999" Surname = "Andreev"
16	car("Andreev", Mark, _, Price) и person("Andreev", "+79999999999", addr("Moscow", "Lesnaya", 12, 2)) Главные функторы не равны	Прямой ход Переход к след. предл.	car("Andreev", Mark, _, Price)	Phone = "+79999999999" Surname = "Andreev"
...
20	car("Andreev", Mark, _, Price) и car("Andreev", "bmw", "green", 1000)	Найден ответ Откат		Phone = "+79999999999" Surname = "Andreev" Mark = "bmw" Price = 1000
Продолжение на следующей странице				

Таблица 1.1 – продолжение

Шаг	Сравниваемые термины; результаты	Дальнейшие действия	Результата	Подстановка
21	car("Andreev", Mark, _, Price) и car("Andreev", "volkswagen", "red", 10000)	Найден ответ Откат		Phone = "+79999999999" Surname = "Andreev" Mark = "volkswagen" Price = 10000
22	car("Andreev", Mark, _, Price) и car("Dmitriev", "lada", "black", 20000) Компоненты попарно не унифицируемы	Прямой ход Переход к след. предл.	car("Andreev", Mark, _, Price)	Phone = "+79999999999" Surname = "Andreev"
...
24	car("Andreev", Mark, _, Price) и bank_depositor("Andreev", "Sber", 22, 1000) Главные функторы не равны	Прямой ход Переход к след. предл.	car("Andreev", Mark, _, Price)	Phone = "+79999999999" Surname = "Andreev"
...
31	car("Andreev", Mark, _, Price) и data_by_surname_and_city(Surname, City, Street, Bank, Phone) Главные функторы не равны	Откат. Продолж. отн. 15	person(Surname, "+79999999999", _), car(Surname, Mark, _, Price)	Phone = "+79999999999"
Продолжение на следующей странице				

Таблица 1.1 – продолжение

Шаг	Сравниваемые термины; результаты	Дальнейшие действия	Резольвента	Подстановка
...
46	car("Andreev", Mark, _, Price) и data_by_surname_and_city(Surname, City, Street, Bank, Phone) Главные функторы не равны	Откат. Продолж. отн. 14	person(Surname, "+79999999999", _), car(Surname, Mark, _, Price)	
...
48	car_by_phone("+79999999999", Surname, Mark, Price) и data_by_surname_and_city(Surname, City, Street, Bank, Phone) Главные функторы не равны	Завершение работы 2 подст. в рез-те	car_by_phone("+79999999999", Surname, Mark, Price)	
Конец таблицы				

Таблица 1.2 – Порядок формирования результата для 2-го вопроса

Шаг	Сравниваемые термины; результаты	Дальнейшие действия	Результата	Подстановка
1	only_mark_by_phone("+79999999999", Mark) и person("Andreev", "+79999999999", addr("Moscow", "Lesnaya", 12, 2)) Главные факторы не равны	Прямой ход Переход к след. предл.	only_mark_by_phone("+79999999999", Mark)	
...
15	only_mark_by_phone("+79999999999", Mark) и only_mark_by_phone(Phone, Mark)	Прямой ход	car_by_phone("+79999999999", _, Mark, _)	Phone = "+79999999999"
...
29	car_by_phone("+79999999999", _, Mark, _) и car_by_phone(Phone, Surname, Mark, Price)	Прямой ход	person(Surname, "+79999999999", _), car(Surname, Mark, _, _)	Phone = "+79999999999"
30	person(_, "+79999999999", _) и person("Andreev", "+79999999999", addr("Moscow", "Lesnaya", 12, 2))	Прямой ход	car("Andreev", Mark, _, _)	Phone = "+79999999999" Surname = "Andreev"
Продолжение на следующей странице				

Таблица 1.2 – продолжение

Шаг	Сравниваемые термы; результаты	Дальнейшие действия	Резольвента	Подстановка
...
35	car("Andreev", Mark, _, _) и car("Andreev", "bmw", "green", 1000)	Найден ответ Откат		Phone = "+79999999999" Surname = "Andreev" Mark = "bmw"
36	car("Andreev", Mark, _, _) и car("Andreev", "volkswagen", "red", 10000)	Найден ответ Откат		Phone = "+79999999999" Surname = "Andreev" Mark = "volkswagen"
...
64	only_mark_by_phone("+79999999999", Mark) и data_by_surname_and_city(Surname, City, Street, Bank, Phone) Главные функторы не равны	Завершение работы 2 подст. в рез-те	only_mark_by_phone("+79999999999", Mark)	
Конец таблицы				

Задание 12 (2): Используя базу знаний, хранящую знания:

- «Телефонный справочник»: Фамилия, №тел, Адрес – структура (Город, Улица, №дома, №кв),
- «Автомобили»: Фамилия_владельца, Марка, Цвет, Стоимость, и др.,
- «Вкладчики банков»: Фамилия, Банк, счет, сумма, др.

Владелец может иметь несколько телефонов, автомобилей, вкладов (Факты). В разных городах есть однофамильцы, в одном городе – фамилия уникальна.

Используя конъюнктивное правило и простой вопрос, обеспечить возможность поиска:

По Марке и Цвету автомобиля найти Фамилию, Город, Телефон и Банки, в которых владелец автомобиля имеет вклады. Лишней информации не находить и не передавать!!!

Владельцев может быть несколько (не более 3-х), один и ни одного.

1. Для каждого из трех вариантов словесно подробно описать порядок формирования ответа (в виде таблицы). При этом, указать – отметить моменты очередного запуска алгоритма унификации и полный результат его работы. Обосновать следующий шаг работы системы. Выписать унификаторы – подстановки. Указать моменты, причины и результат отката, если он есть.
2. Для случая нескольких владельцев (2-х): приведите примеры (таблицы) работы системы при разных порядках следования в БЗ процедур, и знаний в них: («Телефонный справочник», «Автомобили», «Вкладчики банков», или: «Автомобили», «Вкладчики банков», «Телефонный справочник»). Сделайте вывод: Одинаковы ли: множество работ и объем работ в разных случаях?
3. Оформите 2 таблицы, демонстрирующие порядок работы алгоритма унификации вопроса и подходящего заголовка правила (для двух случаев из пункта 2) и укажите результаты его работы: ответ и побочный эффект.

Листинг 1.3 – Реализация базы знаний

```
1 domains
2   surname = string
3   city, street = string
4   house, flat = integer
5   phone = string
6   address = addr(city, street, house, flat)
7   mark = string
8   color = string
9   price = integer
10  bank = string
11  id, amount = integer
12
13 predicates
14   person(surname, phone, address)
15   car(surname, mark, color, price)
16   bank_depositor(surname, bank, id, amount)
17   man_by_car(mark, color, surname, city, phone, bank)
18
19 clauses
20   person("Andreev", "+79999999999", addr("Moscow", "Lesnaya", 12,
21     2)).
22   person("Borisov", "+71111111111", addr("Moscow", "Lesnaya", 13,
23     87)).
24   person("Dmitriev", "+73333333333", addr("Ekaterinburg", "
25     Kamennaya", 13, 87)).
26   person("Fedorov", "+66666666666", addr("Moscow", "Hospital'naya
27     ", 123, 87)).
28
29   car("Andreev", "bmw", "green", 1000).
30   car("Andreev", "volkswagen", "red", 10000).
31   car("Dmitriev", "lada", "black", 20000).
32   car("Fedorov", "lada", "black", 20000).
33   car("Fedorov", "opel", "white", 10).
34
35   bank_depositor("Andreev", "Sber", 22, 1000).
36   bank_depositor("Borisov", "Sber", 33, 10000).
37   bank_depositor("Dmitriev", "Alfa", 44, 20000).
38   bank_depositor("Fedorov", "Sber", 238, 10).
39   bank_depositor("Borisob", "Maze", 1, 10000).
40
41   man_by_car(Mark, Color, Surname, City, Phone, Bank) :-
42     car(Surname, Mark, Color, _),
43     person(Surname, Phone, addr(City, _, _, _)),
44     bank_depositor(Surname, Bank, _, _).
45
46 goal
47   man_by_car("bmw", "green", Surname, City, Phone, Bank).
48   %man_by_car("lada", "black", Surname, City, Phone, Bank).
49   %man_by_car("volkswagen", "blue", Surname, City, Phone, Bank)
50   .
```

Таблица 1.3 – Порядок формирования результата для 2-го вопроса

Шаг	Сравниваемые термы; результаты	Дальнейшие действия	Резольвента	Подстановка
1	man_by_car("lada", "black", Surname, City, Phone, Bank) и person("Andreev", "+79999999999", addr("Moscow", "Lesnaya", 12, 2)) Главные функторы не равны	Прямой ход Переход к след. предл.	man_by_car("lada", "black", Surname, City, Phone, Bank)	
...
15	man_by_car("lada", "black", Surname, City, Phone, Bank) и man_by_car(Mark, Color, Surname, City, Phone, Bank)	Прямой ход	car(Surname, "lada", "black", _), person(Surname, Phone, addr(City, _, _, _)), bank_depositor(Surname, Bank, _, _)	Mark = "lada" Color = "black"
...
23	car(Surname, "lada", "black", _) и car("Dmitriev", "lada", "black", 20000)	Прямой ход	person("Dmitriev", Phone, addr(City, _, _, _)), bank_depositor("Dmitriev", Bank, _, _)	Mark = "lada" Color = "black" Surname = "Dmitriev"
...
Продолжение на следующей странице				

Таблица 1.3 – продолжение

Шаг	Сравниваемые термы; результаты	Дальнейшие действия	Результата	Подстановка
34	person("Dmitriev", Phone, addr(City, _, _, _)) и person("Dmitriev", "+73333333333", addr("Ekaterinburg", "Kamennaya", 13, 87))	Прямой ход	bank_depositor("Dmitriev", Bank, _, _)	Mark = "lada" Color = "black" Surname = "Dmitriev" Phone = "+73333333333" City = "Ekaterinburg"
...
58	bank_depositor("Dmitriev", Bank, _, _) и и bank_depositor("Dmitriev", "Alfa", 44, 20000)	Найден ответ Откат		Mark = "lada" Color = "black" Surname = "Dmitriev" Phone = "+73333333333" City = "Ekaterinburg" Bank = "Alfa"
...
73	car(Surname, "lada", "black", _) и car("Fedorov", "lada", "black", 20000)	Прямой ход	person("Dmitriev", Phone, addr(City, _, _, _)), bank_depositor("Fedorov", Bank, _, _)	Mark = "lada" Color = "black" Surname = "Fedorov"
...
Продолжение на следующей странице				

Таблица 1.3 – продолжение

Шаг	Сравниваемые термы; результаты	Дальнейшие действия	Результата	Подстановка
77	person("Fedorov", Phone, addr(City, _, _, _)) и person("Fedorov", "+666666666666", addr("Moscow", "Hospital'naya", 123, 87))	Прямой ход	bank_depositor("Fedorov", Bank, _, _)	Mark = "lada" Color = "black" Surname = "Fedorov" Phone = "+666666666666" City = "Moscow"
...
90	bank_depositor("Fedorov", Bank, _, _) и bank_depositor("Fedorov", "Sber", 238, 10)	Найден ответ Откат		Mark = "lada" Color = "black" Surname = "Fedorov" Phone = "+666666666666" City = "Moscow" Bank = "Sber"
...
109	car(Surname, "lada", "black", _) и data_by_surname_and_city(man_by_car(Mark, Color, Surname, City, Phone, Bank) Главные функторы не равны	Т.к. откат-ся некуда, зав. работу, 2 подст. в рез-те	person(Surname, Phone, addr(City, _, _, _)), bank_depositor(Surname, Bank, _, _)	Mark = "lada" Color = "black"
Конец таблицы				

2 Контрольный вопросы

2.1 Что такое терм?

Терм - основной элемент языка `Prolog`. Терм – это:

1. Константа:

- Число (целое, вещественное),
- Символьный атом (комбинация символов латинского алфавита, цифр и символа подчеркивания, начинающаяся со строчной буквы),
- Строка: последовательность символов, заключенных в кавычки.

2. Переменная:

- Именованная – обозначается комбинацией символов латинского алфавита, цифр и символа подчеркивания, начинающейся с прописной буквы или символа подчеркивания,
- Анонимная - обозначается символом подчеркивания

3. Составной терм: Это средство организации группы отдельных элементов знаний в единый объект, синтаксически представляется: $f(t_1, t_2, \dots, t_m)$, где f - функтор (отношение между объектами), t_1, t_2, \dots, t_m – термы, в том числе и составные.

2.2 Что такое предикат в матлогике (математике)?

Предикат (n -местный, или n -арный) — это функция с множеством значений $\{0, 1\}$ (или $\{\text{ложь}, \text{истина}\}$), определённая на множестве $M^n = (M_1, M_2, \dots, M_n)$. Таким образом, каждый набор элементов множества характеризуется либо как «истинный», либо как «ложный».

2.3 Что описывает предикат в Prolog?

Предикат в `Prolog` описывает отношение между аргументами процедуры. Процедурой в `Prolog` является совокупность всех правил, описывающих определенное отношение.

2.4 Назовите виды предложений в программе и приведите примеры таких предложений из Вашей программы. Какие предложения являются основными, а какие – не основными? Каковы: синтаксис и семантика (формальный смысл) этих предложений (основных и неосновных)?

В Prolog есть два типа предложений: правила и факты. Правило имеет вид: $A :- B_1, \dots, B_n$. A называется заголовком правила, а B_1, \dots, B_n – телом правила. Заголовок содержит некоторое знание, а тело — условие истинности этого знания. Факт является частным случаем правила — в нем отсутствует тело.

Пример факта из программы:

Листинг 2.1 – Реализация базы знаний, Часть 1

```
42 goal
43 %car_by_phone("+79999999999", Surname, Mark, Price).
44 only_mark_by_phone("+79999999999", Mark).
45 %data_by_surname_and_city("Borisov", "Moscow", Street, Bank,
    Phone).
```

```
car("Andreev", "bmw", "green", 1000).
```

Пример правила из программы:

```
car_by_phone(Phone, Surname, Mark, Price) :- phone(Surname,
Phone, _), car(Surname, Mark, _, Price).
```

Основными называются предложения, не содержащие переменных. Предложения, содержащие переменные называются неосновными.

Синтаксис предложения: **заголовок (составной терм) :- тело (один или последовательность термов).**

Предложения используются для формирования базы знаний о некоторой предметной области. Основное предложение описывает отношение конкретных объектов предметной области. Неосновное предложение описывает множество отношений, потому что переменная, входящая в предложение базы знаний, рассматривается только с квантором всеобщности.

2.5 Каковы назначение, виды и особенности использования переменных в программе на Prolog? Какое предложение БЗ сформулировано в более общей – абстрактной форме: содержащее или не содержащее переменных?

Переменные предназначены для обозначения некоторого неизвестного объекта предметной области. Переменные бывают именованными и анонимными. Именованные переменные уникальны в рамках предложения, а анонимная переменная – любая уникальна. В разных предложениях может использоваться одно имя переменной для обозначения разных объектов.

В ходе выполнения программы выполняется связывание переменных с различными объектами, этот процесс называется конкретизацией. Это относится только к именованным переменным. Анонимные переменные не могут быть связаны со значением.

В более общей форме сформулировано предложение, содержащее переменные, так как заранее неизвестно, каким объектом будет конкретизирована переменная.

2.6 Что такое подстановка?

Пусть дан терм: (X_1, X_2, \dots, X_n) . Подстановка — множество пар, вида: $\{X_i = t_i\}$, где X_i — переменная, а t_i — терм.

2.7 Что такое пример терма? Как и когда строится? Как Вы думаете, система строит и хранит примеры?

Пусть $\Theta = \{X_1 = t_1, X_2 = t_2, \dots, X_n = t_n\}$ — подстановка, A — терм. Результат применения подстановки к терму обозначается $A\Theta$.

Примером терма A называется терм B , если существует подстановка Θ такая, что $B = A\Theta$.

Примеры термов строятся в ходе логического вывода. Для построения примера терма его переменные конкретизируются.