



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 №13

Дисциплина

Функциональное и

логическое программирование

Тема

Работа программы на Prolog

Студент

Набиев Ф.М.

Группа

ИУ7-63Б

Преподаватель

Толпинская Н.Б., Строганов Ю.В.

Москва, 2020 г.

ВВЕДЕНИЕ

Цель работы — получить навыки построения модели предметной области, разработки и оформления программы на Prolog, изучить принципы, логику формирования программы и отдельные шаги выполнения программы на Prolog.

Задачи работы: приобрести навыки декларативного описания предметной области с использованием фактов и правил. Изучить способы использования термов, переменных, фактов и правил в программе на Prolog, принципы и правила сопоставления и отождествления, порядок унификации.

1 Теоретическая часть

В этом разделе приведены ответы на контрольные вопросы.

1.1 Что такое терм?

Терм — основной элемент языка. Терм — это константа (число, символьный атом, строка), переменная (именованная, анонимная), составной терм.

1.2 Что такое предикат в матлогике?

Предикат — это логическая функция от одного или нескольких аргументов. Другими словами, предикат — это функция, отображающая множество произвольной природы в множество {ложь, истина}.

1.3 Что описывает предикат в Prolog?

Название отношений, существующих между объектами.

1.4 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)?

Предложения бывают двух видов — факты и правила.

1. Правило состоит из тела и головы. Голову так же называют заголовком. Синтаксически правило оформляется следующим образом: **голова :- тело.**

Причем, заголовок и тело — это термы, а символ «:-» это специальный символ-разделитель.

2. Факт — частный случай правила, в котором отсутствует тело символ-разделитель.

Пример факта из программы приведён в листинге 1.1.

Листинг 1.1 – Пример факта

```
1 phonebook(petrov , 74400297, address(moscow, lenina , 4, 2)).
```

Пример правила из программы приведён в листинге 1.2.

Листинг 1.2 – Пример правила

```
1 search(Phonenum, Carbrand) :- search(Phonenum, _, Carbrand, _).
```

Если составные термы, факты, правила и вопросы не содержат переменных, то они называются основными. Составные термы, факты, правила и вопросы в момент фиксации в программе могут содержать переменные, тогда они называются неосновными

1.5 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 являются частью процесса сопоставления и предназначены для передачи значений, но не для хранения их. Виды переменных:

- именованная — обозначается комбинацией символов латинского алфавита, цифр и символа подчеркивания, начинающейся с прописной буквы или символа подчеркивания («X», «A21», «_X»);
- анонимная — обозначается символом подчеркивания «_». Любая анонимная переменная уникальна.

Во время вычисления, именованные переменные могут конкретизироваться. Кроме того, они могут быть конкретизированы повторно

путем «отката» вычислительного процесса и отмены ранее проведенной конкретизации для нахождения новых решений.

Анонимные переменные не могут быть связаны со значениями.

Предложения базы знаний, содержащие переменные, сформулировано в более общей — абстрактной форме.

1.6 Что такое подстановка?

Пусть $A(X_1, X_2, \dots, X_n)$ — терм. Тогда подстановкой называют множество пар вида $\{X_i = t_i\}$, $i = \overline{1, n}$, где X_i — переменная, а t_i — терм.

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

Пусть A, B — термы. Терм B называют примером терма A , если для A существует такая подстановка α , что $A\alpha = B$, где $A\alpha$ — это результат применения подстановки α к терму A .

На мой взгляд, примеры термов строятся при поиске решения заданной пользователем цели или внутренних целей, а хранятся они до получения решения.

2 Практическая часть

В этом разделе приведено описание задания и его решения.

2.1 Условие

Составить программу, т.е. 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);
б) Используя сформированное в пункте **(а)** 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 одного из вариантов ответов, и для (а) и для (б), 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.

2.2 База знаний

В листинге 2.1 приведён текст базы знаний, удовлетворяющей условию задачи.

Листинг 2.1 – База знаний

```
1 domains
2   lastname , city , street , carbrand , color , bankname = symbol .
3   phonenum , housenum , apartnum , price , account , deposit = integer .
4   address = address(city , street , housenum , apartnum) .
5 predicates
6   phonebook(lastname , phonenum , address) .
7   car(lastname , carbrand , color , price) .
8   depositor(lastname , bankname , account , deposit) .
9
10  search(phonenum , lastname , carbrand , price) .
11  search(phonenum , carbrand) .
12  search(lastname , city , street , bankname , phonenum) .
13 clauses
14  phonebook(petrov , 74400297 ,
15            address(moscow , lenina , 4 , 2)) .
16  phonebook(igorev , 77270935 ,
17            address(moscow , marksa , 3 , 5)) .
18  phonebook(ivanov , 79345669 ,
19            address(moscow , pushkinskaya , 11 , 1)) .
20  phonebook(ivanov , 70244559 ,
21            address(spb , lenina , 3 , 5)) .
22  phonebook(stasov , 74024456 ,
23            address(spb , marksa , 4 , 4)) .
24  phonebook(petrov , 78771841 ,
25            address(ekb , lenina , 7 , 9)) .
26  phonebook(igorev , 73148253 ,
27            address(ekb , marksa , 6 , 8)) .
28  phonebook(igorev , 73243243 ,
29            address(volgograd , lenina , 9 , 9)) .
30
31  car(petrov , bugatti , red , 2400000) .
32  car(petrov , ferrari , red , 55000) .
33  car(ivanov , ford , pink , 25000) .
34  car(igorev , tesla , purple , 44000) .
35  car(igorev , bmw , green , 3700) .
36  car(igorev , lexus , yellow , 14000) .
37
38  depositor(petrov , agricole , 5 , 52150322) .
39  depositor(igorev , paribas , 4 , 32242424) .
40  depositor(ivanov , sberbank , 2 , 242342142) .
41  depositor(ivanov , sberbank , 6 , 442423123) .
```

```

42 depositor(stasov , sberbank , 1 , 423424233).
43 depositor(petrov , agricole , 6 , 42421234).
44 depositor(igorev , paribas , 3 , 41424214).
45 depositor(igorev , sberbank , 8 , 421342352).
46
47 search(Phonenum , Lastname , Carbrand , Price) :-
48     phonebook(Lastname , Phonenum , _ ) ,
49     car(Lastname , Carbrand , _ , Price) .
50
51 search(Phonenum , Carbrand) :-
52     search(Phonenum , _ , Carbrand , _ ) .
53
54 search(Lastname , City , Street , Bankname , Phonenum) :-
55     phonebook(Lastname , Phonenum , address(City , Street , _ , _ ) ) ,
56     depositor(Lastname , Bankname , _ , _ ) .

```

2.3 Цели

Рассмотрим примеры целей для каждого задания.

2.3.1 Задание 1.а

В листингах 2.2, 2.3, 2.4 приведены цели, использующие предикат `search(phonenum, lastname, carbrand, price)`, а так же их решение.

Листинг 2.2 – Пример №1

```

1 goal
2     search(73243243 , Lastname , Carbrand , Price) .
3
4 % Lastname=igorev , Carbrand=tesla , Price=44000
5 % Lastname=igorev , Carbrand=bmw , Price=3700
6 % Lastname=igorev , Carbrand=lexus , Price=14000
7 % 3 Solutions

```

Листинг 2.3 – Пример №2

```

1 goal
2     search(79345669 , Lastname , Carbrand , Price) .
3
4 % Lastname=ivanov , Carbrand=ford , Price=25000

```


Листинг 2.4 – Пример №3

```

1 goal
2   search(09345669, Lastname, Carbrand, Price).
3
4 % No Solution

```

Рассмотрим подробнее поиск ответа на вопрос из листинга 2.3. Описание этого процесса приведено в таблице 2.1.

Таблица 2.1 – Поиск решения в примере №2

№ шага	Сравниваемые термы; результат; подстановка (если есть)	Дальнейшие действия: прямой ход или откат
1	<p>search(79345669, Lastname, Carbrand, Price) и search(Phonenum, Lastname, Carbrand, Price) совпадают по имени и списку аргументов ⇒ происходит унификация:</p> <ol style="list-style-type: none"> 1) переменная Phonenum унифицируется с константой 79345669, т.е. конкретизируются её значением; 2) переменная Lastname унифицируется с переменной Lastname, т.е. остаётся неконкретизированной; 3) Carbrand унифицируется с Carbrand; 4) Price унифицируется с Price. <p>Результат — search(79345669, Lastname, Carbrand, Price).</p>	<p>В стек откладывается терм search(79345669, Lastname, Carbrand, Price).</p> <p>Прямой ход: на резольвенту попадают термы, из которых состоит тело правила search(Phonenum, Lastname, Carbrand, Price).</p> <p>На вершине стека вопросов находится подцель phonebook(Lastname, Phonenum, _).</p>
2	<p>phonebook(Lastname, Phonenum, _) и phonebook(petrov, 74400297, address(moscow, lenina, 4, 2)) унифицируются:</p> <ol style="list-style-type: none"> 1) Lastname конкретизируется значением petrov; 2) конкретизированная значением 79345669 переменная Phonenum не унифицируются с константой 74400297, так как их значения не совпадают. <p>Унификация не выполняется.</p>	Прямой ход.

3	<p>phonebook(Lastname, Phonenum, _) и phonebook(igorev, 77270935, address(moscow, marka, 3, 5)) не унифицируются, потому что значение конкретизированной переменной Phonenum и константы 77270935 не совпадают.</p>	Прямой ход.
4	<p>phonebook(Lastname, Phonenum, _) и phonebook(ivanov, 79345669, address(moscow, pushkinskaya, 11, 1)) унифицируются:</p> <ol style="list-style-type: none"> 1) Lastname конкретизируется значением константы ivanov; 2) конкретизированная переменная Phonenum унифицируется с константой 79345669, так как их значения совпадают; 3) анонимная переменная унифицируется с константой address(...). <p>Результат — phonebook(igorev, 79345669, _, 3, 5)).</p>	<p>В стек откладывается терм phonebook(Lastname, Phonenum, _).</p> <p>Прямой ход: на вершине стека вопросов находится вторая подцель — car(Lastname, Carbrand, _, Price).</p>
5	<p>car(Lastname, Carbrand, _, Price) и car(petrov, bugatti, red, 2400000) унифицируются:</p> <ol style="list-style-type: none"> 1) конкретизированная значением ivanov переменная Lastname не унифицируется с константой petrov. <p>Унификация не выполняется.</p>	Прямой ход.
6	<p>car(Lastname, Carbrand, _, Price) и car(petrov, ferrari, red, 55000) унифицируются:</p> <ol style="list-style-type: none"> 1) конкретизированная значением ivanov переменная Lastname не унифицируется с константой petrov. <p>Унификация не выполняется.</p>	Прямой ход.

7	<p>car(Lastname, Carbrand, _, Price) и car(ivanov, ford, pink, 25000) унифицируются:</p> <ol style="list-style-type: none"> 1) конкретизированная значением ivanov переменная Lastname унифицируется с константой ivanov. 2) Carbrand конкретизируется значением константы ford; 3) анонимная переменная унифицируется с константой pink; 4) Price конкретизируется значением константы 25000. <p>Результат — car(ivanov, ford, _, 25000).</p>	<p>Найдено первое решение: Lastname = ivanov; Carbrand = ford; Price = 25000.</p> <p>В стек откладывается терм car(Lastname, Carbrand, _, Price). Резольвента опустела.</p> <p>Откат: отложенный терм восстанавливается.</p>
8	car(Lastname, Carbrand, _, Price) не унифицируется с car(igorev, tesla, purple, 44000)	Прямой ход.
9	car(Lastname, Carbrand, _, Price) не унифицируется с car(igorev, bmw, green, 3700)	Прямой ход.
10	car(Lastname, Carbrand, _, Price) не унифицируется с car(igorev, lexus, yellow, 14000)	<p>Обход базы знаний завершен.</p> <p>Откат: восстановление из стека термина phonebook(Lastname, Phonenum, _).</p>
11–15	phonebook(Lastname, Phonenum, _) больше не унифицируется ни с одним из оставшихся в базе знаний фактом или правилом.	<p>Обход базы знаний завершен.</p> <p>Откат: восстановление из стека цели search(79345669, Lastname, Carbrand, Price) и удаление из резольвенты её подцелей.</p>
16	search(79345669, Lastname, Carbrand, Price) больше не унифицируется ни с одним из оставшихся в базе знаний фактом или правилом.	<p>Обход базы знаний завершен.</p> <p>Цель search(79345669, Lastname, Carbrand, Price) покидает резольвенту.</p> <p>Завершение поиска решений. Найдено 1 решение.</p>

2.3.2 Задание 1.6

В листингах 2.5, 2.6, 2.7 приведены цели, использующие предикат search(phonenum, carbrand), а так же их решение.

Листинг 2.5 – Пример №1

```
1 goal
2   search(73243243, Carbrand) .
3
4 % Carbrand=tesla
5 % Carbrand=bmw
6 % Carbrand=lexus
7 % 3 Solutions
```

Листинг 2.6 – Пример №2

```
1 goal
2   search(79345669, Carbrand) .
3
4 % Carbrand=ford
5 % 1 Solution
```

Листинг 2.7 – Пример №3

```
1 goal
2   search(09345669, Carbrand) .
3
4 % No Solution
```

Рассмотрим подробнее поиск ответа на вопрос из листинга 2.6. Описание этого процесса приведено в таблице 2.2.

Таблица 2.2 – Поиск решения в примере №

№ шага	Сравниваемые термы; результат; подстановка (если есть)	Дальнейшие действия: прямой ход или откат
1	1)	

2.3.3 Задание 2

В листингах 2.8, 2.9, 2.10 приведены цели, использующие предикат `search(lastname, city, street, bankname, phonenum)`, а так же их

решение.

Листинг 2.8 – Пример №1

```
1 goal
2   search(petrov , moscow, Street , Bankname, Phonenum) .
3
4 % Street=lenina , Bankname=agricole , Phonenum=74400297
5 % Street=lenina , Bankname=agricole , Phonenum=74400297
6 % 2 Solutions
```

Листинг 2.9 – Пример №2

```
1 goal
2   search(stasov , spb , Street , Bankname, Phonenum) .
3
4 % Street=marksa , Bankname=sberbank , Phonenum=74024456
5 % 1 Solution
```

Листинг 2.10 – Пример №3

```
1 goal
2   search(stasov , moscow, Street , Bankname, Phonenum) .
3
4 % No Solution
```

Рассмотрим подробнее поиск ответа на вопрос из листинга 2.9. Описание этого процесса приведено в таблице 2.3.

Таблица 2.3 – Поиск решения в примере №

№ шага	Сравниваемые термы; результат; подстановка (если есть)	Дальнейшие действия: прямой ход или откат
1	1)	