



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический
университет имени Н.Э. Баумана»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 13

Работа программы на Prolog

Дисциплина	Функциональное и логическое программирование
Студент	Сиденко А.Г.
Группа	ИУ7-63Б
Преподаватель	Толпинская Н.Б., Строганов Ю.В.

Москва, 2020 г.

Задание

Составить программу, т.е. модель предметной области – базу знаний, объединив в ней информацию – знания:

1. **«Телефонный справочник»:** Фамилия, №тел, Адрес – структура (Город, Улица, №дома, №кв),
2. **«Автомобили»:** Фамилия_владельца, Марка, Цвет, Стоимость, и др.,
3. **«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 задания1 и задания2:

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.

Программа

```
1 domains
2  lastname , city , street , model , color , bank = symbol.
3  telephone , house , flat , price , account , deposit = integer.
4  adress = adress(city , street , house , flat).
5 predicates
6  abonent(lastname , telephone , adress).
7  auto(lastname , city , model , color , price).
8  depositor(lastname , city , bank , account , deposit).
9
10 autoByNumber(telephone , lastname , model , price).
11 modelAutoByNumber(telephone , model).
12 adressBankByLastnameCity(lastname , city , street , bank , telephone).
13 clauses
```

```

14 abonent(ellen , 111111, adress(moscow, tverskaya , 1, 1)).
15 abonent(john , 222222, adress(moscow, arbat , 11, 112)).
16 abonent(tom , 333333, adress(moscow, presnya , 10, 11)).
17 abonent(eric , 444444, adress(moscow, pokrovka , 21, 55)).
18 abonent(mark , 555555, adress(moscow, solyanka , 13, 13)).
19 abonent(mark , 888888, adress(kazan , pushkin , 22, 130)).
20 abonent(bill , 666666, adress(ekb , lenin , 12, 88)).
21 abonent(bill , 777777, adress(sp , sadovaya , 1, 12)).
22 auto(ellen , moscow, bmw, red , 10000).
23 auto(tom , moscow, mercedes , black , 15000).
24 auto(eric , moscow, mercedes , white , 20000).
25 auto(eric , moscow, bmw, black , 15000).
26 auto(eric , moscow, porsche , red , 30000).
27 auto(mark , moscow, hyundai , silver , 7000).
28 auto(bill , ekb , hyundai , black , 10000).
29 auto(bill , sp , volvo , black , 13000).
30 depositor(ellen , moscow, sberbank , 10000, 2000).
31 depositor(john , moscow, sberbank , 10000, 3000).
32 depositor(john , moscow, vtb , 20000, 5000).
33 depositor(eric , moscow, sberbank , 100000, 30000).
34 depositor(mark , moscow, sberbank , 10000, 2000).
35 depositor(mark , kazan , vtb , 10000, 2000).
36 depositor(mark , moscow, gazprom , 10000, 2000).
37 depositor(bill , sp , vtb , 30000, 3000).
38
39 autoByNumber(Number, Name, Model, Price) :-
40     abonent(Name, Number, adress(City , _ , _ , _)),
41     auto(Name, City , Model, _ , Price).
42
43 modelAutoByNumber(Number, Model) :-
44     autoByNumber(Number, _ , Model, _).
45
46 adressBankByLastnameCity(Name, City , Street , Bank , Number):-
47     abonent(Name, Number, adress(City , Street , _ , _)),
48     depositor(Name, City , Bank, _ , _).

```

1. Задание 1

- (а) По № телефона найти: Фамилию, Марку автомобиля, Стоимость автомобиля


Листинг 1: Пример 1 задание 1а

```
1 | goal
```

```

2   Number = 444444,
3   autoByNumber(Number, Name, Model, Price).

```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab3\Obj\goal\$000.exe]

```

Number=444444, Name=eric, Model=mersedes, Price=20000
Number=444444, Name=eric, Model=bmw, Price=15000
Number=444444, Name=eric, Model=porsche, Price=30000
3 Solutions


```

Листинг 2: Пример 2 задание 1a

```

1 goal
2   Number = 666666,
3   autoByNumber(Number, Name, Model, Price).

```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab3\Obj\goal\$000.exe]

```

Number=666666, Name=bill, Model=hyundai, Price=10000
1 Solution


```

Листинг 3: Пример 3 задание 1a

```

1 goal
2   Number = 444444,
3   autoByNumber(Number, Name, Model, Price).

```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab3\Obj\goal\$000.exe]

No Solution

Поиск ответа на вопрос из примера 2.

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
1	(подстановка) autoByNumber(Number, Name, Model, Price) = autoByNumber(666666, Name, Model, Price)	Прямой ход, Number=Number, Name=Name, Model=Model, Price=Price
2	abonent(Name, Number, adress (City, _, _, _)), auto(Name, City, Model, _, Price) ↓ (подстановка) abonent(Name, Number, adress (City, _, _, _)) = abonent(ellen, 666666, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)

3	<p>abonent(Name, Number, adress (City, _, _, _)), auto(Name, City, Model, _, Price)</p> <p>↓ (подстановка)</p> <p>abonent(Name, Number, adress (City, _, _, _)) = abonent(john, 666666, adress(moscow, _, _, _)))</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
4	<p>abonent(Name, Number, adress (City, _, _, _)), auto(Name, City, Model, _, Price)</p> <p>↓ (подстановка)</p> <p>abonent(Name, Number, adress (City, _, _, _)) = abonent(tom, 666666, adress(moscow, _, _, _)))</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
5	<p>abonent(Name, Number, adress (City, _, _, _)), auto(Name, City, Model, _, Price)</p> <p>↓ (подстановка)</p> <p>abonent(Name, Number, adress (City, _, _, _)) = abonent(eric, 666666, adress(moscow, _, _, _)))</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
6	<p>abonent(Name, Number, adress (City, _, _, _)), auto(Name, City, Model, _, Price)</p> <p>↓ (подстановка)</p> <p>abonent(Name, Number, adress (City, _, _, _)) = abonent(mark, 666666, adress(moscow, _, _, _)))</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
7	<p>abonent(Name, Number, adress (City, _, _, _)), auto(Name, City, Model, _, Price)</p> <p>↓ (подстановка)</p> <p>abonent(Name, Number, adress (City, _, _, _)) = abonent(mark, 666666, adress(kazan, _, _, _)))</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
8	<p>abonent(Name, Number, adress (City, _, _, _)), auto(Name, City, Model, _, Price)</p> <p>↓ (подстановка)</p> <p>abonent(Name, Number, adress (City, _, _, _)) = abonent(bill, 666666, adress(ekb, _, _, _)))</p>	Результат сравнения термов: true, удаляем значение из стека (откат)

9	(подстановка) auto(Name, City, Model, _, Price) = auto(bill, ekb, Model, _, Price)	Прямой ход, Name = bill, City = ekb
10	auto(Name, City, Model, _, Price) ↓ (подстановка) auto(Name, City, Model, _, Price) = auto(bill, ekb, bmw, _, 10000)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
11	auto(Name, City, Model, _, Price) ↓ (подстановка) auto(Name, City, Model, _, Price) = auto(bill, ekb, mercedes, _, 15000)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
12	auto(Name, City, Model, _, Price) ↓ (подстановка) auto(Name, City, Model, _, Price) = auto(bill, ekb, mercedes, _, 20000)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
13	auto(Name, City, Model, _, Price) ↓ (подстановка) auto(Name, City, Model, _, Price) = auto(bill, ekb, bmw, _, 15000)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
14	auto(Name, City, Model, _, Price) ↓ (подстановка) auto(Name, City, Model, _, Price) = auto(bill, ekb, porsche, _, 30000)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
15	auto(Name, City, Model, _, Price) ↓ (подстановка) auto(Name, City, Model, _, Price) = auto(bill, ekb, hyundai, _, 7000)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
16	auto(Name, City, Model, _, Price) ↓ (подстановка) auto(Name, City, Model, _, Price) = auto(bill, ekb, hyundai, _, 10000)	Результат сравнения термов: true, печать результата, удаляем значение из стека (откат)


Далее поиск следующих результатов по аналогии.

(b) По № телефона найти: Марку автомобиля

Листинг 4: Пример 1 задание 1b

```


1 goal
2   Number = 666666,
3   modelAutoByNumber(Number, Model).
```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab3\Obj\goal\$000.exe]

Number=666666, Model=hyundai
1 Solution

Листинг 5: Пример 2 задание 1b


```
1 goal
2   Number = 222222,
3   modelAutoByNumber (Number, Model).
```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab3\Obj\goal\$000.exe]

No Solution

Листинг 6: Пример 3 задание 1b

```
1 goal
2   Number = 111111,
3   modelAutoByNumber (Number, Model).
```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab3\Obj\goal\$000.exe]

Number=111111, Model=bmw
1 Solution

Поиск ответа на вопрос из примера 3.

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
1	modelAutoByNumber(Number, Model) = autoByNumber(111111, Model)	Прямой ход, Number=Number, Model=Model
2	autoByNumber(Number, _, Model, _)	Прямой ход
3	autoByNumber(Number, Name, Model, Price) = autoByNumber(111111, _, Model, _)	Прямой ход, Number=Number, Model=Model
4	abonent(Name, Number, adress (City, _, _, _)), auto(Name, City, Model, _, Price) ↓ (подстановка) abonent(Name, Number, adress (City, _, _, _)) = abonent(ellen, 111111, adress(moscow, _, _, _))	Результат сравнения термов: true, удаляем значение из стека (откат)

5	(подстановка) auto(Name, City, Model, _, Price) = auto(ellen, moscow, Model, _, Price)	Прямой ход, Name = ellen, City = moscow
6	auto(Name, City, Model, _, Price) ↓ (подстановка) auto(Name, City, Model, _, Price) = auto(ellen, moscow, bmw, _, 10000)	Результат сравнения термов: true, печать результата, удаляем значение из стека (откат)

Далее поиск следующих результатов по аналогии.


- По Фамилии и Городу проживания найти: Улицу проживания, Банки, в которых есть вклады и №телефона.

Листинг 7: Пример 1 задание 2

```

1 goal
2   Name = mark ,
3   City = moscow ,
4   adressBankByLastnameCity (Name, City , Street , Bank , Number) .

```


 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab3\Obj\goal\$000.exe]
Name=mark, City=moscow, Street=solyanka, Bank=sberbank, Number=555555
Name=mark, City=moscow, Street=solyanka, Bank=gazprom, Number=555555
2 Solutions

Листинг 8: Пример 2 задание 2

```

1 goal
2   Name = mark ,
3   City = kazan ,
4   adressBankByLastnameCity (Name, City , Street , Bank , Number) .

```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab3\Obj\goal\$000.exe]
Name=mark, City=kazan, Street=pushkin, Bank=vtb, Number=888888
1 Solution

Листинг 9: Пример 3 задание 2

```

1 goal
2   Name = mark ,
3   City = spb ,
4   adressBankByLastnameCity (Name, City , Street , Bank , Number) .

```


No Solution

Поиск ответа на вопрос из примера 2.

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
1	(подстановка) adresaBankByLastnameCity(Name, City, Street, Bank, Number) = adresaBankByLastnameCity(mark, kazan, Street, Bank, Number)	Прямой ход, Name = Name, City = City, Street = Street, Bank = Bank, Number = Number
2	abonent(Name, Number, adresa(City, Street, _, _)), depositor(Name, City, Bank, _, _) ↓ (подстановка) abonent(Name, Number, adresa(City, Street, _, _)) = abonent(mark, 111111, adresa(kazan, tverskaya, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
3	abonent(Name, Number, adresa(City, Street, _, _)), depositor(Name, City, Bank, _, _) ↓ (подстановка) abonent(Name, Number, adresa(City, Street, _, _)) = abonent(mark, 222222, adresa(kazan, arbat, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
4	abonent(Name, Number, adresa(City, Street, _, _)), depositor(Name, City, Bank, _, _) ↓ (подстановка) abonent(Name, Number, adresa(City, Street, _, _)) = abonent(mark, 333333, adresa(kazan, presnya, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
5	abonent(Name, Number, adresa(City, Street, _, _)), depositor(Name, City, Bank, _, _) ↓ (подстановка) abonent(Name, Number, adresa(City, Street, _, _)) = abonent(mark, 444444, adresa(kazan, pokrovka, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)

6	<p>abonent(Name, Number, adress(City, Street, _, _)), depositor(Name, City, Bank, _, _)</p> <p>↓ (подстановка)</p> <p>abonent(Name, Number, adress(City, Street, _, _)) = abonent(mark, 555555, adress(kazan, pushkin, _, _))</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
7	<p>abonent(Name, Number, adress(City, Street, _, _)), depositor(Name, City, Bank, _, _)</p> <p>↓ (подстановка)</p> <p>abonent(Name, Number, adress(City, Street, _, _)) = abonent(mark, 888888, adress(kazan, pushkin, _, _))</p>	Результат сравнения термов: true, удаляем значение из стека (откат)
8	<p>(подстановка)</p> <p>depositor(Name, City, Bank, _, _) = depositor(mark, kazan, Bank, _, _)</p>	Прямой ход, Name = mark, City = kazan
9	<p>depositor(Name, City, Bank, _, _)</p> <p>↓ (подстановка)</p> <p>depositor(Name, City, Bank, _, _) = depositor(mark, kazan, sberbank, _, _)</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
10	<p>depositor(Name, City, Bank, _, _)</p> <p>↓ (подстановка)</p> <p>depositor(Name, City, Bank, _, _) = depositor(mark, kazan, sberbank, _, _)</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
11	<p>depositor(Name, City, Bank, _, _)</p> <p>↓ (подстановка)</p> <p>depositor(Name, City, Bank, _, _) = depositor(mark, kazan, vtb, _, _)</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
12	<p>depositor(Name, City, Bank, _, _)</p> <p>↓ (подстановка)</p> <p>depositor(Name, City, Bank, _, _) = depositor(mark, kazan, sberbank, _, _)</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
13	<p>depositor(Name, City, Bank, _, _)</p> <p>↓ (подстановка)</p> <p>depositor(Name, City, Bank, _, _) = depositor(mark, kazan, sberbank, _, _)</p>	Результат сравнения термов: false, переход к следующей строке (прямой ход)
14	<p>depositor(Name, City, Bank, _, _)</p> <p>↓ (подстановка)</p> <p>depositor(Name, City, Bank, _, _) = depositor(mark, kazan, vtb, _, _)</p>	Результат сравнения термов: true, печать результата, удаляем значение из стека (откат)

Далее поиск следующих результатов по аналогии.

Ответы на вопросы

1. Что такое терм?

Термом будем называть выражение, образованное из переменных и констант, возможно, с применением функций, а точнее:

- (a) всякая переменная или константа есть терм;
- (b) если t_1, \dots, t_n — термы, а f — n -местный функциональный символ, то $f(t_1, \dots, t_n)$ — терм;
- (c) других термов нет.

2. Что такое предикат в матлогике (математике)?

Предикатом называется функция нескольких переменных, которая в области задания этих переменных, может принимать лишь два значения 1 или 0 (которые мы как всегда можем рассматривать как истину или ложь). Если предикат зависит от n переменных, то он называется n -местным.

3. Что описывает предикат в Prolog?

Утверждения программы — это предикаты.

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.

Предложение более общего вида — **правило** имеет вид: $A :- B_1, \dots, B_n$.

A называется заголовком правила, а B_1, \dots, B_n — телом правила.

Факт — это частный случай правила. Факт — это предложение, в котором отсутствует тело (т.е. тело пустое).

Пример предложения:

```
1 autoByNumber(Number, Name, Model, Price) :-  
2   abonent(Name, Number, adress(City, _, _, _)),  
3   auto(Name, City, Model, _, Price).
```

Пример факта:

```
1 abonent(ellen, 111111, adress(moscow, tverskaya, 1, 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?

При поступлении вопроса с переменной в Пролог-систему. Например:

1 `universe("Mark", X).`

X – переменная, входящая в вопрос, изначально является **неконкретизированной**. Пролог просматривает базу данных в поисках факта, сопоставимого с вопросом. Если неконкретизированная переменная появляется в качестве одного из аргументов, то Пролог считает, что такой аргумент сопоставим с любым другим аргументом, находящимся в том же факте. При обнаружении такого факта переменная X становится **конкретизированной**, обозначая объект, являющийся вторым аргументом найденного факта.

Это относится только к именованным переменным. **Анонимные переменные** не могут быть связаны со значением.

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.

6. Что такое подстановка?

Пусть дан терм: $A(X_1, X_2, \dots, X_n)$

Подстановкой называется множество пар, вида: $\{X_i = t_i\}$, где X_i – переменная, а t_i – терм.

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, тогда результат применения подстановки к терму обозначается: $A\theta$. Применение подстановки заключается в замене каждого вхождения переменной X_i на соответствующий терм. Терм B называется **примером терма** A , если существует такая подстановка θ , что $B = A\theta$.

В процессе выполнения программы – система, используя встроенный алгоритм унификации, пытается обосновать возможность истинности вопроса, строя подстановки и примеры термов (вопроса и формулировки знания), используя базу знаний.