



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	По <code>autoByNumber(Number, Name, Model, Price) = autoByNumber(666666, Name, Model, Price)</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено, заносится в стек <code>autoByNumber(666666, Name, Model, Price)</code> , прямой ход

2	Начинает «раскрываться» правило, т.е. доказывается каждое целевое утверждение в теле правила последовательно слева направо abonent(Name, Number, adress (City, _, _, _)), auto(Name, City, Model, _, Price)	Заносится в стек abonent(Name, 666666, adress (City, _, _, _))
3	По abonent(Name, 666666, adress (City, _, _, _)) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
4	Унификация abonent(ellen, 111111, adress (moscow, tverskaya, 1, 1)) с abonent(Name, 666666, adress (City, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
5	Унификация abonent(john, 222222, adress(moscow, arbat, 11, 112)) с abonent(Name, 666666, adress (City, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
6	Унификация abonent(tom, 333333, adress(moscow, presnya, 10, 11)) с abonent(Name, 666666, adress (City, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
7	Унификация abonent(eric, 444444, adress(moscow, pokrovka, 21, 55)) с abonent(Name, 666666, adress (City, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
8	Унификация abonent(mark, 555555, adress(moscow, solyanka, 13, 13)) с abonent(Name, 666666, adress (City, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
9	Унификация abonent(mark, 888888, adress(kazan, pushkin, 22, 130)) с abonent(Name, 666666, adress (City, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)

10	Унификация abonent(bill, 666666, adress(ekb, lenin, 12, 88)) с abonent(Name, 666666, adress (City, _, _))	Результат сравнения термов: true, Name примет значение bill, City примет значение ekb. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
11	Следующее целевое утверждение auto(Name, City, Model, _, Price)	Заносится в стек auto(bill, ekb, Model, _, Price)
12	По auto(bill, ekb, Model, _, Price) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
13	Унификация auto(ellen, moscow, bmw, red, 10000) с auto(bill, ekb, Model, _, Price)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
14	Унификация auto(tom, moscow, mercedes, black, 15000) с auto(bill, ekb, Model, _, Price)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
15	Унификация auto(eric, moscow, mercedes, white, 20000) с auto(bill, ekb, Model, _, Price)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
16	Унификация auto(eric, moscow, bmw, black, 15000) с auto(bill, ekb, Model, _, Price)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
17	Унификация auto(eric, moscow, porsche, red, 30000) с auto(bill, ekb, Model, _, Price)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
18	Унификация auto(mark, moscow, hyundai, silver, 7000) с auto(bill, ekb, Model, _, Price)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
19	Унификация auto(bill, ekb, hyundai, black, 10000) с auto(bill, ekb, Model, _, Price)	Результат сравнения термов: true, вывод результата, переход к следующей строке (прямой ход)

20	Унификация auto(bill, spb, volvo, black, 13000) с auto(bill, ekb, Model, _, Price)	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаём из стека auto(bill, ekb, Model, _, Price)
21	Унификация abonent(bill, 777777, adress(spb, sadovaya, 1, 12)) с abonent(Name, 666666, adress (City, _, _))	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаём из стека abonent(Name, 666666, adress (City, _, _, _))
22	Достаём из стека autoByNumber(666666, Name, Model, Price)	Стек пуст, завершение программы


(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) .

```

**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	По <code>modelAutoByNumber(Number, Model) = modelAutoByNumber(111111, Model)</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено, заносится в стек <code>modelAutoByNumber(111111, Model)</code> , прямой ход
2	Начинает «раскрываться» правило, т.е. доказывается каждое целевое утверждение в теле правила последовательно слева направо <code>autoByNumber(Number, _, Model, _)</code>	Заносится в стек <code>autoByNumber(111111, _, Model, _)</code> , прямой ход
3	По <code>autoByNumber(111111, _, Model, _)</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
4	Начинает «раскрываться» правило, т.е. доказывается каждое целевое утверждение в теле правила последовательно слева направо <code>abonent(Name, Number, adress (City, _, _, _))</code> , <code>auto(Name, City, Model, _, _)</code>	Заносится в стек <code>abonent(Name, 111111, adress (City, _, _, _))</code> , прямой ход
5	По <code>abonent(Name, 111111, adress (City, _, _, _))</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
6	Унификация <code>abonent(ellen, 111111, adress (moscow, tverskaya, 1, 1))</code> с <code>abonent(Name, 111111, adress (City, _, _, _))</code>	Результат сравнения термов: <code>true</code> , <code>Name</code> примет значение <code>ellen</code> , <code>City</code> примет значение <code>moscow</code> . Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)



7	Следующее целевое утверждение auto(Name, City, Model, _, Price)	Заносится в стек auto(ellen, moscow, Model, _, _)
8	По auto(ellen, moscow, Model, _, _) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
9	Унификация auto(ellen, moscow, bmw, red, 10000) с auto(ellen, moscow, Model, _, _)	Результат сравнения термов: true, вывод результата, переход к следующей строке (прямой ход)
10	Унификация auto(tom, moscow, mersedes, black, 15000) с auto(ellen, moscow, Model, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
11	Унификация auto(eric, moscow, mersedes, white, 20000) с auto(ellen, moscow, Model, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
12	Унификация auto(eric, moscow, bmw, black, 15000) с auto(ellen, moscow, Model, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
13	Унификация auto(eric, moscow, porsche, red, 30000) с auto(ellen, moscow, Model, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
14	Унификация auto(mark, moscow, hyundai, silver, 7000) с auto(ellen, moscow, Model, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
15	Унификация auto(bill, ekb, hyundai, black, 10000) с auto(ellen, moscow, Model, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
16	Унификация auto(bill, spb, volvo, black, 13000) с auto(ellen, moscow, Model, _, _)	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с задан- ным именем, возврат, достаем из стека auto(bill, ekb, Model, _, _)
17	Унификация abonent(john, 222222, adress(moscow, arbat, 11, 112)) с abonent(Name, 111111, adress (City, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)

18	Унификация abonent(tom, 333333, adress(moscow, presnya, 10, 11)) с abonent(Name, 111111, adress (City, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
19	Унификация abonent(eric, 444444, adress(moscow, pokrovka, 21, 55)) с abonent(Name, 111111, adress (City, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
20	Унификация abonent(mark, 555555, adress(moscow, solyanka, 13, 13)) с abonent(Name, 111111, adress (City, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
21	Унификация abonent(mark, 888888, adress(kazan, pushkin, 22, 130)) с abonent(Name, 111111, adress (City, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
22	Унификация abonent(bill, 666666, adress(ekb, lenin, 12, 88)) с abonent(Name, 111111, adress (City, _, _))	Результат сравнения термов: true, Name примет значение bill, City примет значение ekb. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
23	Унификация abonent(bill, 777777, adress(sp, sadovaya, 1, 12)) с abonent(Name, 111111, adress (City, _, _))	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека abonent(Name, 111111, adress (City, _, _, _)), откат
24	Достаем из стека autoByNumber(111111, _, Model, _)	Откат, больше нет целевых утверждений
25	Достаем из стека modelAutoByNumber(111111, Model)	Откат, стек пуст, завершение программы

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


#### Листинг 7: Пример 1 задание 2

1	goal
2	<b>Name</b> = 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) .

```

 [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	По <code>adressBankByLastnameCity(Name, City, Street, Bank, Number) = adressBankByLastnameCity(mark, kazan, Street, Bank, Number)</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено, заносится в стек <code>adressBankByLastnameCity(mark, kazan, Street, Bank, Number)</code> , прямой ход

2	Начинает «раскрываться» правило, т.е. доказывается каждое целевое утверждение в теле правила последовательно слева направо <code>abonent(Name, Number, adress(City, Street, _, _)), depositor(Name, City, Bank, _, _)</code>	Заносится в стек <code>abonent(mark, Number, adress(kazan, Street, _, _))</code>
3	По <code>abonent(mark, Number, adress(kazan, Street, _, _))</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
4	Унификация <code>abonent(ellen, 111111, adress(moscow, tverskaya, 1, 1))</code> с <code>abonent(mark, Number, adress(kazan, Street, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
5	Унификация <code>abonent(john, 222222, adress(moscow, arbat, 11, 112))</code> с <code>abonent(mark, Number, adress(kazan, Street, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
6	Унификация <code>abonent(tom, 333333, adress(moscow, presnya, 10, 11))</code> с <code>abonent(mark, Number, adress(kazan, Street, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
7	Унификация <code>abonent(eric, 444444, adress(moscow, pokrovka, 21, 55))</code> с <code>abonent(mark, Number, adress(kazan, Street, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
8	Унификация <code>abonent(mark, 555555, adress(moscow, solyanka, 13, 13))</code> с <code>abonent(mark, Number, adress(kazan, Street, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
9	Унификация <code>abonent(mark, 888888, adress(kazan, pushkin, 22, 130))</code> с <code>abonent(mark, Number, adress(kazan, Street, _, _))</code>	Результат сравнения термов: <code>true</code> , Number примет значение 888888, Street примет значение pushkin. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)

11	Следующее целевое утверждение depositor(Name, City, Bank, _, _)	Заносится в стек depositor(mark, kazan, Bank, _, _)
12	По depositor(mark, kazan, Bank, _, _) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
13	Унификация depositor(ellen, moscow, sberbank, 10000, 2000) с depositor(mark, kazan, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
14	Унификация depositor(john, moscow, sberbank, 10000, 3000) с depositor(mark, kazan, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
15	Унификация depositor(john, moscow, vtb, 20000, 5000) с depositor(mark, kazan, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
16	Унификация depositor(eric, moscow, sberbank, 100000, 30000) с depositor(mark, kazan, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
17	Унификация depositor(mark, moscow, sberbank, 10000, 2000) с depositor(mark, kazan, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
18	Унификация depositor(mark, kazan, vtb, 10000, 2000) с depositor(mark, kazan, Bank, _, _)	Результат сравнения термов: true, вывод результата, переход к следующей строке (прямой ход)
19	Унификация depositor(mark, moscow, gazprom, 10000, 2000) с depositor(mark, kazan, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
20	Унификация depositor(bill, spb, vtb, 30000, 3000) с depositor(mark, kazan, Bank, _, _)	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека depositor(mark, kazan, Bank, _, _)
10	Унификация abonent(bill, 666666, adress(ekb, lenin, 12, 88)) с abonent(mark, Number, adress(kazan, Street, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)

21	Унификация <code>abonent(bill, 777777, adress(spb, sadovaya, 1, 12))</code> с <code>abonent(mark, Number, adress(kazan, Street, _, _))</code>	Результат сравнения термов: <code>false</code> , в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека <code>abonent(mark, Number, adress(kazan, Street, _, _))</code>
22	Достаем из стека <code>adressBankByLastnameCity(mark, kazan, Street, Bank, Number)</code>	Стек пуст, завершение программы

## Ответы на вопросы

### 1. Что такое терм?

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

- (а) 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;
- (с) 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  $\theta$ , 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.