



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 № 14

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). В разных городах есть однофамильцы, в одном городе – фамилия уникальна.

Обеспечить возможность поиска: По Марке и Цвету автомобиля найти Фамилию, Город, Телефон и Банки, в которых владелец автомобиля имеет вклады.

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  nameCityPhoneBankByAuto(model , color , lastname , city ,
11                             telephone , bank).
12 clauses
13   abonent(ellen , 111111, adress(moscow , tverskaya , 1, 1)).
14   abonent(john , 222222, adress(moscow , arbat , 11, 112)).
15   abonent(tom , 333333, adress(moscow , presnya , 10, 11)).
16   abonent(eric , 444444, adress(moscow , pokrovka , 21, 55)).
17   abonent(mark , 555555, adress(moscow , solyanka , 13, 13)).
18   abonent(mark , 888888, adress(kazan , pushkin , 22, 130)).
19   abonent(bill , 666666, adress(ekb , lenin , 12, 88)).
20   abonent(bill , 777777, adress(spbl , sadovaya , 1, 12)).
21   auto(ellen , moscow , bmw , red , 10000).
22   auto(tom , moscow , mercedes , white , 15000).
23   auto(eric , moscow , mercedes , white , 20000).
24   auto(eric , moscow , bmw , black , 15000).
25   auto(eric , moscow , porsche , red , 30000).
26   auto(mark , moscow , hyundai , silver , 7000).
```

```

27 auto(bill , ekb , hyundai , black , 10000).
28 auto(bill , spb , volvo , black , 13000).
29 depositor(ellen , moscow , sberbank , 10000 , 2000).
30 depositor(john , moscow , sberbank , 10000 , 3000).
31 depositor(john , moscow , vtb , 20000 , 5000).
32 depositor(tom , moscow , gazprom , 5000 , 3000).
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
38 nameCityPhoneBankByAuto(Model , Color , Name , City ,
39                               Telephone , Bank) :-
40     auto(Name , City , Model , Color , _),
41     abonent(Name , Telephone , adress(City , _ , _ , _)),
42     depositor(Name , City , Bank , _ , _).


```

Листинг 1: Пример 1.

```

1 goal
2   Model = mercedes ,
3   Color = white ,
4   nameCityPhoneBankByAuto(Model , Color , Name , City ,
5                               Telephone , Bank).

```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab4\Obj\goal\$000.exe]

Model=mercedes, Color=white, Name=tom, City=moscow, Telephone=333333, Bank=gazprom
Model=mercedes, Color=white, Name=eric, City=moscow, Telephone=444444, Bank=sberbank
2 Solutions

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
1	По nameCityPhoneBankByAuto(Model, Color, Name, City, Telephone, Bank) = nameCityPhoneBankByAuto(mercedes, white, Name, City, Telephone, Bank) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено, заносится в стек nameCityPhoneBankByAuto(mercedes, white, Name, City, Telephone, Bank), прямой ход

2	Начинает «раскрываться» правило, т.е. доказывается каждое целевое утверждение в теле правила последовательно слева направо <code>auto(Name, City, Model, Color, _)</code> , <code>abonent(Name, Telephone, adress(City, _, _, _))</code> , <code>depositor(Name, City, Bank, _, _)</code>	Заносится в стек <code>auto(Name, City, mercedes, white, _)</code>
3	По <code>auto(Name, City, mercedes, white, _)</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
4	Унификация <code>auto(ellen, moscow, bmw, red, 10000)</code> с <code>auto(Name, City, mercedes, white, _)</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
5	Унификация <code>auto(tom, moscow, mercedes, white, 15000)</code> с <code>auto(Name, City, mercedes, white, _)</code>	Результат сравнения термов: <code>true</code> , <code>Name</code> примет значение <code>tom</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)
6	Следующее целевое утверждение <code>abonent(Name, Telephone, adress(City, _, _, _))</code>	Заносится в стек <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code>
7	По <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
8	Унификация <code>abonent(ellen, 111111, adress(moscow, tverskaya, 1, 1))</code> с <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
9	Унификация <code>abonent(john, 222222, adress(moscow, arbat, 11, 112))</code> с <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)

10	Унификация abonent(tom, 333333, adress(moscow, presnya, 10, 11)) с abonent(tom, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: true, Telephone примет значение 333333, Street примет значение presnya. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
11	По depositor(tom, moscow, Bank, _, _) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
12	Унификация depositor(ellen, moscow, sberbank, 10000, 2000) с depositor(tom, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
13	Унификация depositor(john, moscow, sberbank, 10000, 3000) с depositor(tom, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
14	Унификация depositor(john, moscow, vtb, 20000, 5000) с depositor(tom, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
15	Унификация depositor(tom, moscow, gazprom, 5000, 3000) с depositor(tom, moscow, Bank, _, _)	Результат сравнения термов: true, вывод результата, переход к следующей строке (прямой ход)
16	Унификация depositor(eric, moscow, sberbank, 100000, 30000) с depositor(tom, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
17	Унификация depositor(mark, moscow, sberbank, 10000, 2000) с depositor(tom, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
18	Унификация depositor(mark, kazan, vtb, 10000, 2000) с depositor(tom, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
19	Унификация depositor(mark, moscow, gazprom, 10000, 2000) с depositor(tom, moscow, Bank, _, _)	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека depositor(tom, moscow, Bank, _, _)

20	Унификация <code>abonent(eric, 444444, adress(moscow, pokrovka, 21, 55))</code> с <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
21	Унификация <code>abonent(mark, 555555, adress(moscow, solyanka, 13, 13))</code> с <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
22	Унификация <code>abonent(mark, 888888, adress(kazan, pushkin, 22, 130))</code> с <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
23	Унификация <code>abonent(bill, 666666, adress(ekb, lenin, 12, 88))</code> с <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)
24	Унификация <code>abonent(bill, 777777, adress(sp, sadovaya, 1, 12))</code> с <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code>	Результат сравнения термов: <code>false</code> , в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека <code>abonent(tom, Telephone, adress(moscow, _, _, _))</code>
25	Унификация <code>auto(eric, moscow, mercedes, white, 20000)</code> с <code>auto(Name, City, mercedes, white, _)</code>	Результат сравнения термов: <code>true</code> , <code>Name</code> примет значение <code>eric</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)
26	Следующее целевое утверждение <code>abonent(Name, Telephone, adress(City, _, _, _))</code>	Заносится в стек <code>abonent(eric, Telephone, adress(moscow, _, _, _))</code>
27	По <code>abonent(eric, Telephone, adress(moscow, _, _, _))</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено

28	Унификация abonent(ellen, 111111, adress (moscow, tverskaya, 1, 1)) с abonent(eric, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
29	Унификация abonent(john, 222222, adress(moscow, arbat, 11, 112)) с abonent(eric, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
30	Унификация abonent(tom, 333333, adress(moscow, presnya, 10, 11)) с abonent(eric, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
31	Унификация abonent(eric, 444444, adress(moscow, pokrovka, 21, 55)) с abonent(tom, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: true, Telephone примет значение 444444, Street примет значение pokrovka. Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
32	По depositor(eric, moscow, Bank, _, _) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
33	Унификация depositor(ellen, moscow, sberbank, 10000, 2000) с depositor(eric, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
34	Унификация depositor(john, moscow, sberbank, 10000, 3000) с depositor(eric, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
35	Унификация depositor(john, moscow, vtb, 20000, 5000) с depositor(eric, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
36	Унификация depositor(tom, moscow, gazprom, 5000, 3000) с depositor(eric, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
37	Унификация depositor(eric, moscow, sberbank, 100000, 30000) с depositor(eric, moscow, Bank, _, _)	Результат сравнения термов: true, вывод результата, переход к следующей строке (прямой ход)

38	Унификация depositor(mark, moscow, sberbank, 10000, 2000) с depositor(eric, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
39	Унификация depositor(mark, kazan, vtb, 10000, 2000) с depositor(eric, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
40	Унификация depositor(mark, moscow, gazprom, 10000, 2000) с depositor(eric, moscow, Bank, _, _)	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаём из стека depositor(eric, moscow, Bank, _, _)
41	Унификация abonent(mark, 555555, adress(moscow, solyanka, 13, 13)) с abonent(eric, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
42	Унификация abonent(mark, 888888, adress(kazan, pushkin, 22, 130)) с abonent(eric, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
43	Унификация abonent(bill, 666666, adress(ekb, lenin, 12, 88)) с abonent(eric, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
44	Унификация abonent(bill, 777777, adress(sp, sadovaya, 1, 12)) с abonent(eric, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаём из стека abonent(eric, Telephone, adress(moscow, _, _, _)))
45	Унификация auto(eric, moscow, bmw, black, 15000) с auto(Name, City, mercedes, white, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
46	Унификация auto(eric, moscow, porsche, red, 30000) с auto(Name, City, mercedes, white, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
47	Унификация auto(mark, moscow, hyundai, silver, 7000) с auto(Name, City, mercedes, white, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
48	Унификация auto(bill, ekb, hyundai, black, 10000) с auto(Name, City, mercedes, white, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)


49	Унификация auto(bill, spb, volvo, black, 13000) с auto(Name, City, mersedes, white, _)	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека auto(Name, City, mersedes, white, _)
50	Достаем из стека nameCityPhoneBankByAuto(mersedes, white, Name, City, Telephone, Bank)	Стек пуст, завершение программы

Листинг 2: Пример 2.

```

1 goal
2   Model = bmw,
3   Color = red ,
4   nameCityPhoneBankByAuto (Model , Color , Name, City ,
5                               Telephone , Bank) .

```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab4\Obj\goal\$000.exe]

Model=bmw, Color=red, Name=ellen, City=moscow, Telephone=111111, Bank=sberbank
1 Solution|

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
1	По nameCityPhoneBankByAuto(Model, Color, Name, City, Telephone, Bank) = nameCityPhoneBankByAuto(mersedes, white, Name, City, Telephone, Bank) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено, заносится в стек nameCityPhoneBankByAuto(bmw, red, Name, City, Telephone, Bank), прямой ход
2	Начинает «раскрываться» правило, т.е. доказывается каждое целевое утверждение в теле правила последовательно слева направо auto(Name, City, Model, Color, _), abonent(Name, Telephone, adress(City, _, _)), depositor(Name, City, Bank, _, _)	Заносится в стек auto(Name, City, bmw, red, _)

3	По <code>auto(Name, City, bmw, red, _)</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
4	Унификация <code>auto(ellen, moscow, bmw, red, 10000)</code> с <code>auto(Name, City, bmw, red, _)</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)
5	Следующее целевое утверждение <code>abonent(Name, Telephone, adress(City, _, _, _))</code>	Заносится в стек <code>abonent(ellen, Telephone, adress(moscow, _, _, _))</code>
6	По <code>abonent(ellen, Telephone, adress(moscow, _, _, _))</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
7	Унификация <code>abonent(ellen, 111111, adress(moscow, tverskaya, 1, 1))</code> с <code>abonent(ellen, Telephone, adress(moscow, _, _, _))</code>	Результат сравнения термов: <code>true</code> , <code>Telephone</code> примет значение <code>111111</code> , <code>Street</code> примет значение <code>tverskaya</code> . Установим маркер. Анонимные переменные не связываются со значением. Переход к следующему целевому утверждению в теле правила (прямой ход)
8	По <code>depositor(ellen, moscow, Bank, _, _)</code> ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
9	Унификация <code>depositor(ellen, moscow, sberbank, 10000, 2000)</code> с <code>depositor(ellen, moscow, Bank, _, _)</code>	Результат сравнения термов: <code>true</code> , вывод результата, переход к следующей строке (прямой ход)
10	Унификация <code>depositor(john, moscow, sberbank, 10000, 3000)</code> с <code>depositor(ellen, moscow, Bank, _, _)</code>	Результат сравнения термов: <code>false</code> , переход к следующей строке (прямой ход)

11	Унификация depositor(john, moscow, vtb, 20000, 5000) с depositor(ellen, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
12	Унификация depositor(tom, moscow, gazprom, 5000, 3000) с depositor(ellen, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
13	Унификация depositor(eric, moscow, sberbank, 100000, 30000) с depositor(ellen, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
14	Унификация depositor(mark, moscow, sberbank, 10000, 2000) с depositor(ellen, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
15	Унификация depositor(mark, kazan, vtb, 10000, 2000) с depositor(ellen, moscow, Bank, _, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
16	Унификация depositor(mark, moscow, gazprom, 10000, 2000) с depositor(ellen, moscow, Bank, _, _)	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека depositor(ellen, moscow, Bank, _, _)
17	Унификация abonent(john, 222222, adress(moscow, arbat, 11, 112)) с abonent(ellen, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
18	Унификация abonent(tom, 333333, adress(moscow, presnya, 10, 11)) с abonent(ellen, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
19	Унификация abonent(eric, 444444, adress(moscow, pokrovka, 21, 55)) с abonent(ellen, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
20	Унификация abonent(mark, 555555, adress(moscow, solyanka, 13, 13)) с abonent(ellen, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
21	Унификация abonent(mark, 888888, adress(kazan, pushkin, 22, 130)) с abonent(ellen, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)

22	Унификация abonent(bill, 666666, adress(ekb, lenin, 12, 88)) с abonent(ellen, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, переход к следующей строке (прямой ход)
23	Унификация abonent(bill, 777777, adress(spб, sadowaya, 1, 12)) с abonent(ellen, Telephone, adress(moscow, _, _, _))	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека abonent(ellen, Telephone, adress(moscow, _, _, _))
24	Унификация auto(tom, moscow, mercedes, white, 15000) с auto(Name, City, bmw, red, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
25	Унификация auto(eric, moscow, mercedes, white, 20000) с auto(Name, City, bmw, red, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход))
26	Унификация auto(eric, moscow, bmw, black, 15000) с auto(Name, City, bmw, red, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
27	Унификация auto(eric, moscow, porsche, red, 30000) с auto(Name, City, bmw, red, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
28	Унификация auto(mark, moscow, hyundai, silver, 7000) с auto(Name, City, bmw, red, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
29	Унификация auto(bill, ekb, hyundai, black, 10000) с auto(Name, City, bmw, red, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
30	Унификация auto(bill, spб, volvo, black, 13000) с auto(Name, City, bmw, red, _)	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека auto(Name, City, bmw, red, _)
31	Достаем из стека nameCityPhoneBankByAuto(bmw, red, Name, City, Telephone, Bank)	Стек пуст, завершение программы

Листинг 3: Пример 3.

```


1 goal
2 Model = bmw,

```

```

3   Color = silver ,
4   nameCityPhoneBankByAuto(Model , Color , Name, City ,
5                                     Telephone , Bank ).

```

 [Inactive Y:\Desktop\university\3_course\sem6\Logical_programming\lab4\Obj\goal\$000.exe]

No Solution

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
1	По nameCityPhoneBankByAuto(Model, Color, Name, City, Telephone, Bank) = nameCityPhoneBankByAuto(bmw, silver, Name, City, Telephone, Bank) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено, заносится в стек nameCityPhoneBankByAuto(bmw, silver, Name, City, Telephone, Bank), прямой ход
2	Начинает «раскрываться» правило, т.е. доказывается каждое целевое утверждение в теле правила последовательно слева направо auto(Name, City, Model, Color, _), abonent(Name, Telephone, adress(City, _, _, _)), depositor(Name, City, Bank, _, _)	Заносится в стек auto(Name, City, bmw, silver, _)
3	По auto(Name, City, bmw, silver, _) ищется системой определение отношения (по имени предиката и списку (числу) аргументов)	Определение отношения найдено
4	Унификация auto(ellen, moscow, bmw, red, 10000) с auto(Name, City, bmw, silver, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
5	Унификация auto(tom, moscow, mercedes, white, 15000) с auto(Name, City, bmw, silver, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
6	Унификация auto(eric, moscow, mercedes, white, 20000) с auto(Name, City, bmw, silver, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
7	Унификация auto(eric, moscow, bmw, black, 15000) с auto(Name, City, bmw, silver, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)

8	Унификация auto(eric, moscow, porsche, red, 30000) с auto(Name, City, bmw, silver, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
9	Унификация auto(mark, moscow, hyundai, silver, 7000) с auto(Name, City, bmw, silver, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
10	Унификация auto(bill, ekb, hyundai, black, 10000) с auto(Name, City, bmw, silver, _)	Результат сравнения термов: false, переход к следующей строке (прямой ход)
11	Унификация auto(bill, spb, volvo, black, 13000) с auto(Name, City, bmw, silver, _)	Результат сравнения термов: false, в базе знаний больше ни одного утверждения с заданным именем, возврат, достаем из стека auto(Name, City, bmw, silver, _)
12	Достаем из стека nameCityPhoneBankByAuto(bmw, silver, Name, City, Telephone, Bank)	Стек пуст, завершение программы

Сравнение объема работ при разных порядках следования в БЗ процедур, и знаний в них: («Телефонный справочник», «Автомобили», «Вкладчики банков», или: «Автомобили», «Вкладчики банков», «Телефонный справочник»)

Заметим, что в пролог-программах предложения, определяющие один и тот же предикат, принято располагать вместе. Эти предложения иногда называют пролог-процедурой. В общем случае пролог-программа состоит из фактов и набора процедур.

Prolog обрабатывает правило в порядке следования предикатов в его теле, а значит порядок в Базе Знаний предикатов, никак не повлияет на получение ответа.

Порядок найденных решений соответствует порядку расположения в пролог-программе подтверждающих фактов, поскольку при доказательстве целей-вопросов интерпретатор последовательно просматривает предложения программы, начиная с первого.

Но объем работ остается тем же, таким образом таблицы остаются теми же.

Порядок работы алгоритма унификации вопроса и подходящего заголовка правила.

nameCityPhoneBankByAuto(bmw, red, Name, City, Telephone, Bank) = nameCityPhoneBankByAuto(bmw, red, Name, City, Telephone, Bank) :- auto(Name, City, Model, Color, _), abonent(Name, Telephone, adress(City, _, _)), depositor(Name, City, Bank, _, _)

Шаг унификации	Результующая ячейка	Рабочее поле	Пункт алгоритма	Стек
0			1	nameCityPhoneBankByAuto(bmw, red, Name, City, Telephone, Bank) = nameCityPhoneBankByAuto(Model, Color, Name, City, Telephone, Bank) :- auto(Name, City, Model, Color, _), abonent(Name, Telephone, adress(City, _, _)), depositor(Name, City, Bank, _, _)
1		nameCityPhoneBankByAuto(bmw, red, Name, City, Telephone, Bank) = nameCityPhoneBankByAuto(Model, Color, Name, City, Telephone, Bank) :- auto(Name, City, Model, Color, _), abonent(Name, Telephone, adress(City, _, _)), depositor(Name, City, Bank, _, _) →	e	auto(Name, City, bmw, red, _) = auto(ellen, moscow, bmw, red, 10000), abonent(Name, Telephone, adress(City, _, _)), depositor(Name, City, Bank, _, _)
2	Name = ellen, City = moscow	← auto(Name, City, bmw, red, _) = auto(ellen, moscow, bmw, red, 10000) →	e	abonent(Name, Telephone, adress(City, _, _)) = abonent(ellen, 111111, adress(moscow, tverskaya, 1, 1)), depositor(Name, City, Bank, _, _)

3	Name = ellen, City = moscow Telephone = 111111	← abonent (ellen, Telephone, adress(moscow, _, _)) = abonent(ellen, 111111, adress(moscow, tverskaya, 1, 1))	e	depositor (ellen, moscow, Bank, _, _)= depositor(ellen, moscow, sberbank, 10000, 2000)
4	Name = ellen, City = moscow Telephone = 111111 Bank = sberbank	← depositor (ellen, moscow, Bank, _, _) = depositor(ellen, moscow, sberbank, 10000, 2000) →	e	
Вывод	подстанов- ка	Успех		Стек пуст

Ответы на вопросы

1. В какой части правила сформулировано знание? Это знание о чем, с формальной точки зрения?

Знания о предметной области выражаются на языке Пролог в виде предложений, называемых утверждениями (clauses).

2. Что такое процедура?

В Visual Прологе предложения с одним и тем же предикатом в заголовке и одной и той же аргументностью должны идти одно за другим. Такая совокупность предложений называется **процедурой**.

3. Сколько в БЗ текущего задания процедур?

4 – abonent, auto, depositor, nameCityPhoneBankByAuto

4. Что такое пример терма, это частный случай терма, пример? Как строится пример?

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,

строая подстановки и примеры термов (вопроса и формулировки знания), используя базу знаний.

5. Что такое наиболее общий пример?

Терм C называется **общим примером термов** A и B , если существуют такие подстановки θ_1 и θ_2 , что $C = A\theta_1$ и $C = B\theta_2$

6. Назначение и результат работы алгоритма унификации. Что значит двунаправленная передача параметров при работе алгоритма унификации, поясните на примере одного из случаев пункта 3.

Унификация двух термов – это основной шаг доказательства. В процессе работы система выполняет большое число унификаций. **Унификация** – операция, которая позволяет формализовать процесс логического вывода.

С помощью алгоритма унификации происходит двунаправленная передача параметров процедурам. Например, из внешнего мира в программу для дальнейшего использования или из программы во внешний мир – значения интересующего нас параметра.

7. В каком случае запускается механизм отката?

Откат дает возможность получить много решений в одном вопросе к программе.

Во всех точках программы, где существуют альтернативы, в стек заносятся точки возврата.

Если впоследствии окажется, что выбранный вариант не приводит к успеху, то осуществляется откат к последней из имеющихся в стеке точек программы, где был выбран один из альтернативных вариантов.

Выбирается очередной вариант, программа продолжает свою работу. Если все варианты в точке уже были использованы, то регистрируется неудачное завершение и осуществляется переход на предыдущую точку возврата, если такая есть.

При откате все связанные переменные, которые были означены после этой точки, опять освобождаются.

8. Виды и назначение переменных в Prolog. Примеры из задания. Почему использованы те или другие переменные (примеры из задания)?

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. Используются, если нас не интересует значение данного параметра.

1

<code> depositor (ellen , moscow, Bank, _, _).</code>

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.