



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*

*По предмету: «Функциональное и логическое  
программирование»*

**Тема:**

Преподаватель: Строганов Ю.В.

Студент: Гасанзаде М.А.,

Группа: ИУ7-66Б

Москва, 2020 г.

## 1. Текст задания

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),  
в) 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 задания 1 и задания 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.

## 2. Ответы на вопросы

1. Что такое терм?

Терм – основной элемент языка Prolog. Может быть константой (число, символьный атом, строка), преременной (именованной или анонимной) или составным термом (функтуатор(терм1, терм2, ...)).

2. Что такое предикат в матлогике (математике)?

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

### 3. Что описывает предикат в Prolog?

В Prolog предикат описывает общий вид фактов и правил.

### 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)?

Предложение может быть фактом или правилом. Предложения бывают основные (не содержащие переменные) и неосновные (содержащие).

Пример:

*person("Ivanov", "000-000", address("Example", "street", 0, 0)).* - **основное**  
*all\_by\_phone(Number, Surname, Brand, Price) :- person(Surname, Number, \_), car(Surname, Brand, \_, Price).* – **неосновное** (правило)  
*all\_by\_phone("000-000", Surname, Brand, Price).* – **неосновное** (запрос)

синтаксис основных: имя\_предиката(значение1, значение2, ...).

синтаксис неосновных: имя\_предиката(значение/имя\_переменной, ...).

### 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?

Переменная обозначает данные, одинаковые внутри предложения. Переменные должны начинаться с большой буквы или символа подчеркивания и состоять только из символов латинского алфавита и символов подчеркивания. Переменные бывают именованные и анонимные (их имя – один символ «\_»). Предложение, сформулированное с помощью переменных, является более абстрактным, так как не требует заранее зафиксированного значения, а лишь описывает, какие значения должны совпадать. Соответственно, по запросам с разными значениями одни и те же предложения могут давать разные ответы, если были описаны с переменными.

### 6. Что такое подстановка?

Подстановка – набор термов  $t_1, t_2, \dots, t_N$ , подставляемых в терм  $A(X_1, X_2, \dots, X_N)$ . Применение подстановки заключается в замене каждого вхождения переменной  $x_i$  на соответствующий терм. Пусть  $\Theta = \{ x_1 = t_1, x_2 = t_2, \dots, x_n = t_n \}$  – подстановка. Обозначение ее результата:  $A\Theta$ .

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

Терм  $B$  называется примером терма  $A$ , если существует такая подстановка  $\Theta$ , что  $B=A\Theta$ .

Вероятно, система строит примеры при вычислении ответа на вопрос, и может их хранить, если в данном правиле часто идет обращение к данному примеру.

### 3. Листинг кода

```
domains
    surname, number, city, street, brand, model, color, bank, account =
symbol.
    price, money = integer. % thousand
    address_t = address(city, street, integer, integer).

predicates
    person(surname, number, address_t).
    car(surname, brand, model, color, price).
    deposit(surname, address_t, bank, account, money).
    all_by_phone(number, surname, brand, price).
    brand_by_phone(number, brand).
    by_surname_city(surname, city, street, bank, number).

clauses
    person("Ivanov", "000-000", address("Example", "street", 0, 0)).
    person("Ivanov", "111-111", address("City-17", "Gordon street", 0, 0)).
    person("Petrov", "001-917", address("St. Petersburg", "Lenina", 24,
42)).
    person("Sidorov", "555-555", address("Los Angeles", "Apple street", 0,
1)).
    person("A", "123-456", address("B", "C avenue", 13, 14)).
    person("Another", "123-321", address("One", "Good street", 3, 12)).
    person("One", "999-666", address("More", "Pioneer street", 3, 4)).
    person("Not", "987-654", address("Enough", "Bad Fantasy avenue", 9,
9)).

    car("Ivanov", "Bugatti", "La Voiture Noire", "Black", 1178000).
    car("Ivanov", "Aston Martin", "Valkyrie", "Grey", 230000).
    car("Petrov", "Lada", "Kalina", "White", 200).
    car("Sidorov", "Ford", "Focus", "Red", 400).

    deposit("Ivanov", address("Example", "street", 0, 0), "Sberbank", "0-0-
0-0", 999999999).
    deposit("Ivanov", address("Example", "street", 0, 0), "VTB", "0-0-0-1",
1).
    deposit("Ivanov", address("City-17", "Gordon street", 0, 0), "Tinkoff",
"0-1-0-1", 987654).
    deposit("Petrov", address("St. Petersburg", "Lenina", 24, 42), "Alfa",
"1-2-3-4", 999999999).
    deposit("Sidorov", address("Los Angeles", "Apple street", 0, 1),
"Mavrodi", "6-9-6-9", 1).
```

```

person("Another", "123-321", address("One", "Good street", 3, 12)).
person("One", "999-666", address("More", "Pioneer street", 3, 4)).
person("Not", "987-654", address("Enough", "Bad Fantasy avenue", 9,
9)).

car("Ivanov", "Bugatti", "La Voiture Noire", "Black", 1178000).
car("Ivanov", "Aston Martin", "Valkyrie", "Grey", 230000).
car("Petrov", "Lada", "Kalina", "White", 200).
car("Sidorov", "Ford", "Focus", "Red", 400).

deposit("Ivanov", address("Example", "street", 0, 0), "Sberbank", "0-
0-0-0", 999999999).
deposit("Ivanov", address("Example", "street", 0, 0), "VTB", "0-0-0-
1", 1).
deposit("Ivanov", address("City-17", "Gordon street", 0, 0),
"Tinkoff", "0-1-0-1", 987654).
deposit("Petrov", address("St. Petersburg", "Lenina", 24, 42),
"Alfa", "1-2-3-4", 999999999).
deposit("Sidorov", address("Los Angeles", "Apple street", 0, 1),
"Mavrodi", "6-9-6-9", 1).
deposit("Another", address("One", "Good street", 3, 12), "Bankname",
"10-20-30-40", 40302010).

all_by_phone(Number, Surname, Brand, Price) :- person(Surname,
Number, _), car(Surname, Brand, _, _, Price).
brand_by_phone(Number, Brand) :- all_by_phone(Number, _, Brand, _).
by_surname_city(Surname, City, Street, Bank, Number) :-
person(Surname, Number, address(City, Street, _, _)), deposit(Surname,
address(City, Street, _, _), Bank, _, _).

goal
%all_by_phone("000-000", Surname, Brand, Price).
%brand_by_phone("000-000", Brand).
%by_surname_city("Ivanov", "Example", Street, Bank, Number).

```

## 4. Примеры работы

```

Surname=Ivanov, Brand=Bugatti, Price=11780000
Surname=Ivanov, Brand=Aston Martin, Price=2300000
2 Solutions

```

*Ответ на вопрос all\_by\_phone("000-000", Surname, Brand, Price).*

```

Brand=Bugatti
Brand=Aston Martin
2 Solutions

```

*Ответ на вопрос brand\_by\_phone("000-000", Brand).*

```

Street=street, Bank=Sberbank, Number=000-000
Street=street, Bank=VTB, Number=000-000
2 Solutions

```

*Ответ на вопрос by\_surname\_city("Ivanov", "Example", Street, Bank, Number).*

## 5. Описание порядка поиска ответа

*all\_by\_phone("000-000", Surname, Brand, Price).*

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	<i>all_by_phone("000-000", Surname, Brand, Price).</i> и <i>all_by_phone(Number, Surname, Brand, Price) :- person(Surname, Number, _), car(Surname, Brand, _, _, Price).</i>  Number = "000-000"	прямой ход. Нужно вычислить person и car
2	<i>person(Surname,"000-000",_)</i> и <i>person("Ivanov", "000-000", address("Example", "street", 0, 0)).</i>  Surname="Ivanov"	person вычислен, вычисляем car
3	<i>car("Ivanov", Brand, _, _, Price)</i> и <i>car("Ivanov", "Bugatti", "La Voiture Noire", "Black", 1178000)</i>  Brand="Bugati", Price=1178000	car вычислен, получен результат 1. Ищем остальные результаты
4	<i>car("Ivanov", Brand, _, _, Price)</i> и <i>car("Ivanov", "Aston Martin", "Valkyrie", "Grey", 230000).</i>  Brand="Aston Martin", Price=230000	car вычислен, получен результат 2. Ищем остальные результаты

5	<p>car("Ivanov", Brand, _, _, Price) и car("Petrov", "Lada", "Kalina", "White", 200)</p> <p>Примеры не совпадают</p>	<p>car вычислен, новых результатов нет, ищем остальные</p>
6	<p>car("Ivanov", Brand, _, _, Price) и car("Sidorov", "Ford", "Focus", "Red", 400)</p> <p>Примеры не совпадают</p>	<p>car вычислен, новых результатов нет, ищем остальные</p> <p>все примеры с car для Surname="Ivanov" разобраны, ищем другие Surname</p>
7	<p>person(Surname,"000-000", _) и person("Ivanov", "111-111", address("City-17", "Gordon street", 0, 0)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
8	<p>person(Surname,"000-000", _) и person("Petrov", "001-917", address("St. Petersburg", "Lenina", 24, 42)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
9	<p>person(Surname,"000-000", _) и person("Sidorov", "555-555", address("Los Angeles", "Apple street", 0, 1)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
10	<p>person(Surname,"000-000", _) и person("A", "123-456", address("B", "C avnenue", 13, 14)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>

11	<p>person(Surname,"000-000", _) и person("Another", "123-321", address("One", "Good street", 3, 12)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
12	<p>person(Surname,"000-000", _) и person("One", "999-666", address("More", "Pioneer street", 3, 4)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
13	<p>person(Surname,"000-000", _) и person("Not", "987-654", address("Enough", "Bad Fantasy avenue", 9, 9)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет. Все примеры с person разобраны. all_by_phone вычислен, получено 2 результата</p>

*brand\_by\_phone("000-000", Brand).*

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	<p>brand_by_phone("000-000", Brand) и brand_by_phone(Number, Brand) :- all_by_phone(Number, _, Brand, _).</p> <p>Number = "000-000"</p>	<p>прямой ход. Нужно вычислить all_by_phone</p>
2	<p>all_by_phone("000-000", _, Brand, _). и all_by_phone(Number, Surname, Brand, Price) :- person(Surname, Number, _),</p>	<p>прямой ход. Нужно вычислить person и car</p>



	car(Surname, Brand, _, _, Price).  Number = "000-000"	
3	person(Surname,"000-000", _) и person("Ivanov", "000-000", address("Example", "street", 0, 0)).  Surname="Ivanov"	person вычислен, вычисляем car
4	car("Ivanov", Brand, _, _, Price) и car("Ivanov", "Bugatti", "La Voiture Noire", "Black", 1178000)  Brand="Bugati", Price=1178000	car вычислен, получен результат 1. Ищем остальные результаты
5	car("Ivanov", Brand, _, _, Price) и car("Ivanov", "Aston Martin", "Valkyrie", "Grey", 230000).  Brand="Aston Martin", Price=230000	car вычислен, получен результат 2. Ищем остальные результаты
6	car("Ivanov", Brand, _, _, Price) и car("Petrov", "Lada", "Kalina", "White", 200)  Примеры не совпадают	car вычислен, новых результатов нет, ищем остальные
7	car("Ivanov", Brand, _, _, Price) и car("Sidorov", "Ford", "Focus", "Red", 400)  Примеры не совпадают	car вычислен, новых результатов нет, ищем остальные  все примеры с car для Surname="Ivanov" разобраны, ищем другие Surname

8	<p>person(Surname,"000-000", _) и person("Ivanov", "111-111", address("City-17", "Gordon street", 0, 0)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
9	<p>person(Surname,"000-000", _) и person("Petrov", "001-917", address("St. Petersburg", "Lenina", 24, 42)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
10	<p>person(Surname,"000-000", _) и person("Sidorov", "555-555", address("Los Angeles", "Apple street", 0, 1)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
11	<p>person(Surname,"000-000", _) и person("A", "123-456", address("B", "C avnenue", 13, 14)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
12	<p>person(Surname,"000-000", _) и person("Another", "123-321", address("One", "Good street", 3, 12)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
12	<p>person(Surname,"000-000", _) и person("One", "999-666", address("More", "Pioneer street", 3, 4)).</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
13	<p>person(Surname,"000-000", _) и person("Not", "987-654",</p>	<p>person вычислен, новых результатов нет. Все примеры</p>

	address("Enough", "Bad Fantasy avenue", 9, 9)).  Примеры не совпадают	с person разобраны. all_by_phone вычислен, brand_by_phone вычислен получено 2 результата
--	---	---

*by\_surname\_city("Ivanov", "Example", Street, Bank, Number).*

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	by_surname_city("Ivanov", "Example", Street, Bank, Number) и by_surname_city(Surname, City, Street, Bank, Number) :- person(Surname, Number, address(City, Street, _, _)), deposit(Surname, address(City, Street, _, _), Bank, _, _)  Surname="Ivanov" City="Example"	Прямой ход. Необходимо вычислить person и deposit
2	person("Ivanov", Number, address("Example", Street, _, _)) и person("Ivanov", "000-000", address("Example", "street", 0, 0))  Number="000-000" Street="street"	Person вычислен, вычисляем deposit
3	deposit("Ivanov", address("Example", "street", _, _), Bank, _, _) и deposit("Ivanov", address("Example", "street", 0,	deposit вычислен, получен результат 1, ищем остальные результаты

	0), "Sberbank", "0-0-0-0", 999999999).	
	Bank="Sberbank"	
4	deposit("Ivanov", address("Example", "street", _, _), Bank, _, _) и deposit("Ivanov", address("Example", "street", 0, 0), "VTB", "0-0-0-1", 1).  Bank="VTB"	deposit вычислен, получен результат 2, ищем остальные результаты
5	deposit("Ivanov", address("Example", "street", _, _), Bank, _, _) и deposit("Ivanov", address("City-17", "Gordon street", 0, 0), "Tinkoff", "0-1- 0-1", 987654)  Примеры не совпадают	deposit вычислен, новых результатов нет, ищем остальные
6	deposit("Ivanov", address("Example", "street", _, _), Bank, _, _) и deposit("Petrov", address("St. Petersburg", "Lenina", 24, 42), "Alfa", "1-2-3-4", 999999999)  Примеры не совпадают	deposit вычислен, новых результатов нет, ищем остальные
7	deposit("Ivanov", address("Example", "street", _, _), Bank, _, _) и deposit("Sidorov", address("Los Angeles", "Apple street", 0, 1), "Mavrodi", "6-9-6-9", 1)  Примеры не совпадают	deposit вычислен, новых результатов нет, ищем остальные

8	<p>deposit("Ivanov", address("Example", "street", _, _), Bank, _, _) и deposit("Another", address("One", "Good street", 3, 12), "Bankname", "10-20- 30-40", 40302010)</p> <p>Примеры не совпадают</p>	<p>deposit вычислен, новых результатов нет, все примеры для deposit при Number="000- 000" и Street="street" рассмотрены, ищем новые Number и Street</p>
9	<p>person("Ivanov", Number, address("Example", Street, _, _)) и person("Ivanov", "111- 111", address("City-17", "Gordon street", 0, 0))</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
10	<p>person("Ivanov", Number, address("Example", Street, _, _)) и person("Petrov", "001- 917", address("St. Petersburg", "Lenina", 24, 42))</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
11	<p>person("Ivanov", Number, address("Example", Street, _, _)) и person("Sidorov", "555- 555", address("Los Angeles", "Apple street", 0, 1))</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
12	<p>person("Ivanov", Number, address("Example", Street, _, _)) и person("A", "123-456", address("B", "C avnenuе", 13, 14))</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>

13	<p>person("Ivanov", Number, address("Example", Street, _, _)) и person("Another", "123-321", address("One", "Good street", 3, 12))</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
14	<p>person("Ivanov", Number, address("Example", Street, _, _)) и person("One", "999-666", address("More", "Pioneer street", 3, 4))</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, ищем остальные</p>
15	<p>person("Ivanov", Number, address("Example", Street, _, _)) и person("Not", "987-654", address("Enough", "Bad Fantasy avenue", 9, 9))</p> <p>Примеры не совпадают</p>	<p>person вычислен, новых результатов нет, все примеры с person разобраны. by_surname_city вычислен получено 2 результата</p>