



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»

---

## Отчёт

### по лабораторной работе №13

Название «Структура программы на Prolog и ее реализация»

---

Дисциплина «Функциональное и логическое программирование»

---

Студент ИУ7-65Б

---

(подпись, дата)

Клименко А.К.

---

(Фамилия И.О.)

Преподаватель

---

(подпись, дата)

Толпинская Н.Б.

---

(Фамилия И.О.)

Москва, 2022

## Введение

**Цель работы** – изучить структуру, особенности и принципы оформления программы, и способ выполнения программы на Prolog.

Для достижения поставленной цели необходимо решить следующие задачи:

- приобрести навыки декларативного описания предметной области с использованием фактов, правил и некоторых специальных разделов программы.
- Изучить порядок использования фактов и правил в программе на Prolog, принципы и особенности сопоставления и отождествления термов, на основе механизма унификации.

## Теоритические вопросы

1) *В каком фрагменте программы сформулировано знание? Это знание о чем на формальном уровне?*

**Ответ:** знание сформулировано в заголовках правил. Каждое знание на формальном уровне представляет факт наличия связи между объектами.

2) *Что содержит тело правила?*

**Ответ:** тело правила содержит конъюнкцию термов, которые предствляют условие истинности заголовка.

3) *Что дает использование переменных при формулировании знаний? В чем отличие формулировки знания с помощью термов с одинаковой арностью при использовании одной переменной и при использовании нескольких переменных?*

**Ответ:** использование переменных позволяет повысить уровень абстракции и обобщить некоторое знание на группы объектов. При формулировании знания с помощью термов с одинаковой арностью при использовании нескольких переменных фиксируется знание о связи между объектами различных (возможно совпадающих) групп, которым позиционно соответствуют аргументы в терме. При этом будут учитываться все возможные комбинации значений переменных.

4) *С каким квантором переменные входят в правило, в каких пределах переменная уникальна?*

**Ответ:** в правило переменные входят с квантором общности. Именованная переменная уникальна в пределах одного предложения, тогда как анонимная переменная уникальна всегда.

5) *Какова семантика (смысл) предложений раздела DOMAINS? Когда, где и с какой целью используется это описание?*

**Ответ:** семантика предложений раздела DOMAINS заключается в описании структуры нестандартных доменов, использующихся при работе с предикатами. При определении доменов используются стандартные и определенные ранее идентификаторы доменов. Идентификаторы доменов условны и воспринимаются системой формально – не влияют на распределение памяти.

6) *Какова семантика (смысл) предложений раздела PREDICATES? Когда, и где используется это описание? С какой целью?*

**Ответ:** в разделе PREDICATES указывается важная во время работы системы информация о природе и структуре объектов, обозначенных аргументами, между которыми устанавливается отношение в предикате.

7) *Унификация каких термов запускается на самом первом шаге работы системы? Каковы назначение и результат использования алгоритма унификации?*

**Ответ:** на первом шаге работы системы запускается алгоритм унификации для терма вопроса и заголовка первого правила в базе знаний. Результат использования алгоритма унификации представляет собой ответ да или нет на вопрос «унифицируются ли два терма?». При ответе да результатом также является подстановка, сформированная в процессе работы алгоритма унификации.

8) *В каком случае запускается механизм отката?*

**Ответ:** механизм отката запускается в случае, когда система попадает в тупиковое состояние – резольвента не пуста, но вся база знаний уже была просмотрена с целью подбора знания для текущей цели доказательства.

## Задание

Необходимо Создать базу знаний «**Собственники**»:

- «**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.

Дополнить (минимально изменив) базу знаниями о дополнительной собственности владельца. Преобразовать знания об автомобиле к форме знаний о собственности.

Вид собственности (кроме автомобиля):

- **Строение**, стоимость и другие его характеристики;
- **Участок**, стоимость и другие его характеристики;
- **Водный транспорт**, стоимость и другие его характеристики.

Описать и использовать вариантный домен: **Собственность**. Владелец может иметь, но только один объект каждого вида собственности (это касается и автомобиля), или не иметь некоторых видов собственности.

Используя конъюнктивное правило и разные формы задания одного вопроса, обеспечить возможность поиска:

1. Названий всех объектов собственности заданного субъекта,
2. Названий и стоимости всех объектов собственности заданного субъекта,
3. \* Разработать правило, позволяющее найти суммарную стоимость всех объектов собственности заданного субъекта.

Для 2-го пункт и одной фамилии составить таблицу, отражающую конкретный порядок работы системы, с объяснениями порядка работы и особенностей использования доменов (указать конкретные T1 и T2 и полную подстановку на каждом шаге).

## Текст программы

```
1 domains
2     address = address(symbol City, symbol Street, integer HouseNum, integer AppartNum).
3     property = car(symbol Mark, integer Cost);
4                 building(symbol Name, integer Cost);
5                 sector(symbol Name, integer Cost);
6                 ship(symbol Name, integer Cost).
7     integerList = integer*.
8
9 predicates
10    telephone(symbol Surname, symbol TelNum, address).
11    car(symbol Surname, symbol Mark, symbol Color, integer Cost).
12    account(symbol Surname, symbol Bank, symbol AccNum, integer Cash).
13    own(symbol Surname, property).
14    getOwningInfo(symbol Surname, symbol Name, integer Cost).
15    getListSum(integerList, integer Sum).
16    getTotalCost(symbol Surname, integer Cost).
17
18 clauses
19    telephone(ivanov, "+79162694425", address(smolensk, baumanskaya, 9, 38)).
20    telephone(sidorov, "+79578163207", address(moscow, pushkinskaya, 69, 29)).
21    telephone(petrova, "+79690758483", address(omsk, lermontova, 65, 6)).
22    telephone(sidorova, "+79917012024", address(omsk, baumanskaya, 82, 79)).
23    telephone(petrova, "+79533641292", address(moscow, leninskaya, 33, 79)).
24    car(petrov, skoda, white, 900000).
25    car(petrova, hynday, blue, 720000).
26    car(sidorova, skoda, gray, 900000).
27    car(ivanova, opel, black, 800000).
28    car(sidorov, hynday, gray, 1000000).
29    account(ivanova, alphabank, a2394, 71000).
30    account(ivanov, tinkoff, a0064, 10000).
31    account(sidorov, sberbank, a0020, 48000).
32    account(sidorova, sberbank, a3564, 85000).
33    account(petrov, tinkoff, a5992, 81000).
34    own(ivanov, building(house1, 1800000)).
35    own(ivanova, building(house2, 1900000)).
36    own(petrova, building(house3, 1000000)).
37    own(sidorov, building(house4, 1100000)).
38    own(sidorova, sector(sector1, 490000)).
39    own(petrov, sector(sector2, 480000)).
40    own(ivanova, sector(sector3, 470000)).
41    own(invanov, sector(sector4, 460000)).
42    own(petrova, sector(sector5, 450000)).
43    own(Surname, car(Mark, Cost)) :-
44        car(Surname, Mark, _, Cost).
```

```

45     getOwningInfo(Surname, Name, Cost) :-
46         own(Surname, car(Name, Cost));
47         own(Surname, building(Name, Cost));
48         own(Surname, sector(Name, Cost)).
49
50     getListSum([], 0) :- !.
51     getListSum([Res], Res) :- !.
52     getListSum([Head,Next|Tail], Res) :-
53         TmpRes = Head + Next,
54         getListSum([TmpRes|Tail], Res).
55
56     getTotalCost(Surname, TotalCost) :-
57         findall(Cost, getOwningInfo(Surname, _, Cost), CostList),
58         getListSum(CostList, TotalCost).
59
60 goal
61     %1 getOwningInfo(petrov, Name, _).
62     %2 getOwningInfo(petrov, Name, Cost).
63     getTotalCost(sidorov, TotalCost).

```

## Порядок поиска ответа для задания 2

№ шаг	Сравниваемые термы; результат; подстановка	Дальнейшие действия	Резольвента
1	T1=getOwningInfo(petrov, Name, Cost) T2=getOwningInfo(Surname, Name, Cost) унифицированы theta={Surname=petrov}	Прямой ход, замена терма вопроса в резольvente на тело правила.	getOwningInfo(petrov, Name, Cost)
2	T1=own(petrov, car(Name, Cost)) T2=own(Surname, car(Mark, Cost)) унифицированы theta={Surname=petrov, Name=Mark}	Прямой ход, замена терма резольвенты на тело правила.	own(petrov, car(Name, Cost)); own(petrov, building(Name, Cost)); own(petrov, sector(Name, Cost))
3	T1=car(petrov, Name, _, Cost) T2=car(petrov, shkola, white, 900000) унифицированы theta={Name=shkola, Cost=900000}	Вывод найденного решения. Продолжение поиска других решений.	car(petrov, Name, _, Cost); own(petrov, building(Name, Cost)); own(petrov, sector(Name, Cost))
4-7	T1=car(petrov, Name, _, Cost) T2=car(petrova, hynday, build, 720000) не унифицированы	Переход к следующему правилу в базе знаний.	car(petrov, Name, _, Cost); own(petrov, building(Name, Cost)); own(petrov, sector(Name, Cost))
8	Достижение конца базы знаний	Резольвента не пуста – откат.	car(petrov, Name, _, Cost); own(petrov, building(Name, Cost)); own(petrov, sector(Name, Cost))
9	Достижение конца базы знаний	Резольвента не пуста – откат. Переход к следующему терму в дизъюнкции	own(petrov, car(Name, Cost)); own(petrov, building(Name, Cost)); own(petrov, sector(Name, Cost))
10	T1=own(petrov, building(Name, Cost)) T2=own(petrov, building(house5, 1200000)) унифицированы theta={Name=house5, Cost=1200000}	Подобран факт. Вывод найденного решения. Продолжение поиска других решений.	own(petrov, building(Name, Cost)); own(petrov, sector(Name, Cost))
11-14	T1=own(petrov, building(Name, Cost)) T2=own(ivanov, building(house1, 1800000)) не унифицированы	Переход к следующему правилу в базе знаний.	own(petrov, building(Name, Cost)); own(petrov, sector(Name, Cost))
15	Достижение конца базы знаний	Переход к следующему терму дизъюнкции	own(petrov, sector(Name, Cost))
16	T1=own(petrov, sector(Name, Cost)) T2=own(petrov, sector(sector2, 480000)) унифицированы theta={Name=sector2, Cost=480000}	Подобран факт. Вывод найденного решения. Продолжение поиска других решений.	own(petrov, sector(Name, Cost))
17-20	T1=own(petrov, sector(Name, Cost)) T2=own(sidorova, sector(sector1, 490000)) не унифицированы	Переход к следующему правилу в базе знаний	own(petrov, sector(Name, Cost))
21	Достижение конца базы знаний	Окончание работы алгоритма. Выведены все решения.	own(petrov, sector(Name, Cost))

## **Заключение**

В ходе работы были приобретены навыки построения предметной области, разработки и оформления программы на Prolog. Были изучены принципы, логика формирования программы и отдельные шаги выполнения программы написанной на языке Prolog. Были освоены принципы и правила сопоставления, отождествления и унификации.