



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»

**Отчет по лабораторной работе №13-14  
по дисциплине  
"Функциональное и логическое программирование"**

Тема Структура программы на Prolog и использование правил

Студент Малышев И. А.

Группа ИУ7-61Б

Оценка (баллы) \_\_\_\_\_

Преподаватель: Толпинская Н. Б.

Москва — 2022 г.

# Лабораторная работа №13

## Задание

Создать базу знаний «Собственники», дополнив базу знаний, хранящую знания (лаб. 13):

- «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.,

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 (пояснять для какого №задания – какой вопрос), 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 пункта 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).

# Решение

```
1 domains
2   surname, phone, city, street, color, bank, name = string
3   home, flat, cost, account, summ, total = integer
4   address = address(city, street, home, flat)
5
6   own = building(name, cost);
7         region(name, cost);
8         car(name, color, cost);
9         water_transport(name, color, cost)
10
11 predicates
12   phone_book(surname, phone, address)
13   deposit(surname, bank, account, summ)
14   owner(surname, own)
15
16   all_owns(surname, name)
17   all_owns_cost(surname, name, cost)
18
19   all_owns_cost_type(surname, symbol, cost)
20   total_cost(surname, total)
21
22 clauses
23   phone_book("Malyshev", "+78005553535", address("Moscow", "Obychnaya", 11, 2)).
24   phone_book("Shatskiy", "+71231421433", address("Saint Peterburg", "Olenevaya",
25         12, 4)).
26   phone_book("Voronin", "+71454663765", address("Saratov", "Bychkovaya", 12, 11)).
27   phone_book("Gribochkov", "+71531432289", address("Tver", "Tomatnaya", 12, 7)).
28   phone_book("Sazonov", "+71766543721", address("Moscow", "Marmeladnaya", 13, 6)).
29   phone_book("Tsetochkin", "+71728332062", address("Tver", "Kabachkovaya", 16,
30         1)).
31
32   owner("Shatskiy", car("Suzuki", "red", 10000000)).
33   owner("Gribochkov", car("BMW", "yellow", 15000000)).
34   owner("Malyshev", car("Highwayman", "gray", 100000000)).
35   owner("Voronin", car("Volga", "black", 20000000)).
36   owner("Malyshev", region("Hell", 666666666)).
37   owner("Tsetochkin", building("Big garden", 550000)).
38   owner("Gribochkov", water_transport("Boatalia", "pink", 55555555)).
39
40   deposit("Sazonov", "Sber", 145464235, 1000).
41   deposit("Shatskiy", "Tinkoff", 585642576, 20000).
42   deposit("Voronin", "Raif", 346536624, 100000).
43   deposit("Malyshev", "Sber", 364562663, 10000).
44
45   all_owns(Surname, Name) :- owner(Surname, car(Name, _, _)).
46   all_owns(Surname, Name) :- owner(Surname, building(Name, _)).
47   all_owns(Surname, Name) :- owner(Surname, region(Name, _)).
48   all_owns(Surname, Name) :- owner(Surname, water_transport(Name, _, _)).
```

```

48 all_owns_cost(Surname, Name, Cost) :- owner(Surname, car(Name, _, Cost)).
49 all_owns_cost(Surname, Name, Cost) :- owner(Surname, building(Name, Cost)).
50 all_owns_cost(Surname, Name, Cost) :- owner(Surname, region(Name, Cost)).
51 all_owns_cost(Surname, Name, Cost) :- owner(Surname, water_transport(Name, _,
    Cost)).
52
53 all_owns_cost_type(Surname, car, Cost) :- owner(Surname, car(_, _, Cost)), !.
54 all_owns_cost_type(Surname, building, Cost) :- owner(Surname, building(_,
    Cost)), !.
55 all_owns_cost_type(Surname, region, Cost) :- owner(Surname, region(_, Cost)), !.
56 all_owns_cost_type(Surname, water_transport, Cost) :- owner(Surname,
    water_transport(_, _, Cost)), !.
57 all_owns_cost_type(_, _, 0).
58
59 total_cost(Surname, Total) :-
60     all_owns_cost_type(Surname, car, Cost1),
61     all_owns_cost_type(Surname, building, Cost2),
62     all_owns_cost_type(Surname, region, Cost3),
63     all_owns_cost_type(Surname, water_transport, Cost4),
64     Total = Cost1 + Cost2 + Cost3 + Cost4.
65
66 goal
67     all_owns("Malyshev", Name).
68     all_owns_cost("Shatskiy", Name, Cost).
69     total_cost("Gribochkov", Total).

```

# Порядок работы

| № шага | Сравниваемые термы; результат; подстановка, если есть   | Дальнейшие действия: прямой ход или откат (к чему приводит?)  |
|--------|---|---|
| 1      | Сравнение:<br>all_owns_cost("Shatskiy", Name, Cost) =<br>phone_book("Malyshev", "+78005553535",<br>address("Moscow", "Obychnaya", 11, 2)).<br>Унификация неуспешна (несовпадение функторов)                 | Откат, переход к следующему предложению   |
| 2-21   | ...   | ...   |
| 22     | Сравнение:<br>all_owns_cost("Shatskiy", Name, Cost) =<br>all_owns_cost(Surname, Name, Cost).<br>Унификация успешна<br>Подстановка: Surname = "Shatskiy", Name = Name,<br>Cost = Cost                        | Новое состояние резольвенты:<br>owner("Shatskiy", car(Name, _, Cost))<br>owner("Shatskiy", building(Name, Cost))<br>owner("Shatskiy", region(Name, Cost))<br>owner("Shatskiy", water_transport(Name, _, Cost))<br>Откат, переход к следующему предложению |
| 23     | Сравнение:<br>owner("Shatskiy", car(Name, _, Cost)) =<br>phone_book("Malyshev", "+78005553535",<br>address("Moscow", "Obychnaya", 11, 2)).<br>Унификация неуспешна (несовпадение функторов)                 | Откат, переход к следующему предложению   |
| 24-28  | ..  | ..  |
| 29     | Сравнение:<br>owner("Shatskiy", car(Name, _, Cost)) =<br>owner("Shatskiy", car("Suzuki", "red", 10000000)).<br>Унификация успешна<br>Подстановка: Surname = "Shatskiy", Name = "Suzuki",<br>Cost = 10000000 | Вывод: Surname = "Shatskiy", Name =<br>"Suzuki", Cost = 10000000<br>Откат, переход к следующему предложению, новая подстановка: Surname =<br>"Shatskiy", Name = Name, Cost = Cost   |
| 30     | Сравнение:<br>owner("Shatskiy", car(Name, _, Cost)) =<br>owner("Gribochkov", car("BMW", "yellow", 15000000)).<br>Унификация неуспешна (несовпадение термов)   | Откат, переход к следующему предложению   |
| 31-35  | ...   | ...   |
| 36     | Сравнение:<br>owner("Shatskiy", car(Name, _, Cost)) =<br>deposit("Sazonov", "Sber", 145464235, 1000).<br>Унификация неуспешна (несовпадение функторов)  | Откат, переход к следующему предложению   |
| 37-53  | ...   | Новое состояние резольвенты:<br>owner("Shatskiy", building(Name, Cost))<br>owner("Shatskiy", region(Name, Cost))<br>owner("Shatskiy", water_transport(Name, _, Cost))<br>Откат, достижение БЗ, переход к следующему предложению относительно шага 23      |

|         |   |   |
|---------|---|---|
| 54      | Сравнение:<br>owner("Shatskiy", building(Name, Cost)) =<br>phone_book("Malyshev", "+78005553535",<br>address("Moscow", "Obychnaya", 11, 2)).<br>Унификация неуспешна (несовпадение функторов) | Откат, переход к следующему предложению   |
| 55-74   | ...   | ...   |
| 75      | Сравнение:<br>owner("Shatskiy", building(Name, Cost)) =<br>owner("Shatskiy", car("Suzuki", "red", 10000000)).<br>Унификация неуспешна (несовпадение термов)                                   | Откат, переход к следующему предложению   |
| 76-81   | ...   | ...   |
| 82      | Сравнение:<br>owner("Shatskiy", building(Name, Cost)) =<br>deposit("Sazonov", "Sber", 145464235, 1000).<br>Унификация неуспешна (несовпадение функторов)                                      | Откат, переход к следующему предложению   |
| 83-99   | ...   | Новое состояние резольвенты:<br>owner("Shatskiy", region(Name, Cost))<br>owner("Shatskiy", water_transport(Name, _, Cost))<br>Откат, достижение БЗ, переход к следующему предложению относительно шага 23 |
| 100     | Сравнение:<br>owner("Shatskiy", building(Name, Cost)) =<br>owner("Shatskiy", car("Suzuki", "red", 10000000)).<br>Унификация неуспешна (несовпадение термов)                                   | Откат, переход к следующему предложению   |
| 101-05  | ...   | ...   |
| 106     | Сравнение:<br>owner("Shatskiy", region(Name, Cost)) =<br>deposit("Sazonov", "Sber", 145464235, 1000).<br>Унификация неуспешна (несовпадение функторов)  | Откат, переход к следующему предложению   |
| 107-123 | ...   | Новое состояние резольвенты:<br>owner("Shatskiy", water_transport(Name, _, Cost))<br>Откат, достижение БЗ, переход к следующему предложению относительно шага 23  |
| 124     | Сравнение:<br>owner("Shatskiy", water_transport(Name, _, Cost)) =<br>owner("Shatskiy", car("Suzuki", "red", 10000000)).<br>Унификация неуспешна (несовпадение термов)                         | Откат, переход к следующему предложению   |
| 125-130 | ...   | ...   |
| 131     | Сравнение:<br>owner("Shatskiy", water_transport(Name, _, Cost)) =<br>deposit("Sazonov", "Sber", 145464235, 1000).<br>Унификация неуспешна (несовпадение функторов)                            | Откат, переход к следующему предложению   |
| 129-145 | ...   | Новое состояние резольвенты: Пустая<br>Достижение БЗ, завершение работы программы   |

# Лабораторная работа №14

## Постановка задачи

Создать базу знаний: «ПРЕДКИ», позволяющую наиболее эффективным способом (за меньшее количество шагов, что обеспечивается меньшим количеством предложений БЗ – правил), и используя разные варианты (примеры) одного вопроса, определить (указать: какой вопрос для какого варианта):

1. По имени субъекта определить всех его бабушек (предки 2-го колена);
2. По имени субъекта определить всех его дедушек (предки 2-го колена);
3. По имени субъекта определить всех его бабушек и дедушек (предки 2-го колена);
4. По имени субъекта определить его бабушку по материнской линии (предки 2-го колена);
5. По имени субъекта определить его бабушку и дедушку по материнской линии (предки 2-го колена).

Минимизировать количество правил и количество вариантов вопросов. Использовать конъюнктивные правила и простой вопрос.

Для одного из вариантов ВОПРОСА и конкретной БЗ составить таблицу, отражающую конкретный порядок работы системы, с объяснениями:

- очередная проблема на каждом шаге и метод ее решения,
- каково новое текущее состояние резольвенты, как получено,
- какие дальнейшие действия? (запускается ли алгоритм унификации? Каких термов? Почему этих?),
- вывод по результатам очередного шага и дальнейшие действия.

Так как резольвента хранится в виде стека, то состояние резольвенты требуется отображать в столбик: вершина – сверху! Новый шаг надо начинать с нового состояния резольвенты!

## Решение

```
1 domains
2   name, sex = string
3   human = human(name, sex)
4
5 predicates
6   is_parent(human, human)
7   is_grandparent(human, sex, name)
8
9 clauses
10  is_grandparent(human(GName, GSex), PSex, Name) :-
11      is_parent(human(GName, GSex), human(TmpName, PSex)),
12      is_parent(human(TmpName, _), human(Name, _)).
13
14  is_parent(human("Elena", f), human("Ivan", m)).
15  is_parent(human("Alexey", m), human("Ivan", m)).
16  is_parent(human("Vera", f), human("Elena", f)).
17  is_parent(human("Evgeny", m), human("Elena", f)).
18  is_parent(human("Nina", f), human("Alexey", m)).
19  is_parent(human("Anatoly", m), human("Alexey", m)).
20
21 goal
22  is_grandparent(human(GName, f), _, "Ivan").
23  is_grandparent(human(GName, m), _, "Ivan").
24  is_grandparent(human(GName, _), _, "Ivan").
25  is_grandparent(human(GName, f), f, "Ivan").
26  is_grandparent(human(GName, _), f, "Ivan").
```



## Порядок работы

| № шага | Состояние резольвенты   | Для каких термов запускается алгоритм унификации: T1=T2 и каков результат (и подстановка)   | Дальнейшие действия: прямой ход или откат  |
|--------|---|---|--|
| 1      | is_parent(human(GName, GSex), human("Ivan", PSex)),<br>is_parent(human(TmpName, PSex), human(Name, _)). | Запуск алгоритма унификации для is_grandparent(human(GName, f), _, "Ivan") и is_grandparent(human(GName, GSex), PSex, Name).<br>Успешная унификация. Подстановка: GSex = f, GName = GName, Name = "Ivan"  | Прямой ход, переход к следующему предложению   |
| 2      | is_parent(human(GName, GSex), human("Ivan", PSex)),<br>is_parent(human(PName, PSex), human(Name, _)).   | Запуск алгоритма унификации для is_parent(human(GrandPName, GSex), human("Ivan", PSex)) и is_grandparent(human(GName, GSex), PSex, Name).<br>Унификация неуспешна.  | Прямой ход, переход к следующему предложению   |
| 3      | is_parent(human(GName, f), human("Elena", f))   | Запуск алгоритма унификации для is_parent(human(GName, GSex), human("Ivan", PSex)) и is_parent(human("Elena", f), human("Ivan", m)).<br>Успешная унификация. Подстановка: GSex = f, Psex = f, GName = GName, TmpName = "Elena", Name = "Ivan"   | Прямой ход, попытка унификации is_parent(human(GName, f), human("Elena", f))                                     |
| 4      | is_parent(human(GName, f), human("Elena", f))   | Запуск алгоритма унификации для is_parent(human(GName, f), human("Elena", f)) и is_grandparent(human(GName, GSex), PSex, Name).<br>Унификация неуспешна.  | Прямой ход, переход к следующему предложению   |
| 5-6    | ...   | ...   | ...  |
| 7      | Пустая  | Запуск алгоритма унификации для is_parent(human(GName, f), human("Elena", f)) и is_parent(human("Vera", f), human("Elena", f)).<br>Унификация успешна. Подстановка: GSex = f, Psex = f, TmpName = "Elena", Name = "Ivan", GName = "Vera".       | Получен результат (GName = "Vera"). Откат (+ откат к следующему предложению относительно 3, т.к. БЗ закончилась) |
| 8      | is_parent(human(GName, f), human("Alexey", m))  | Запуск алгоритма унификации для is_parent(human(GName, GSex), human("Ivan", PSex)) и is_parent(human("Alexey", m), human("Ivan", m)).<br>Успешная унификация. Подстановка: GSex = f, GName = GName, Psex = m, TmpName = "Alexey", Name = "Ivan" | Прямой ход, попытка унификации is_parent(human(GName, f), human("Alexey", m))                                    |

|       |   |  |   |
|-------|---|--|---|
| 9     | is_parent(human(GName, f), human("Alexey", m))  | Запуск алгоритма унификации для is_parent(human(GName, f), human("Alexey", m)) и is_grandparent(human(GName, GSex), PSex, Name).<br>Унификация неуспешна.  | Прямой ход, переход к следующему предложению  |
| 10-15 | ...   | ...  | ...   |
| 16    | Пустая  | Запуск алгоритма унификации для is_parent(human(GName, f), human("Alexey", m)) и is_parent(human("Nina", f), human("Alexey", m)).<br>Успешная унификация. Подстановка: GSex = f, GName = "Nina", Psex = m, TmpName = "Alexey", Name = "Ivan" | Получен результат (GName = "Nina").<br>Откат. |
| 17    | is_parent(human(GName, GSex), human("Ivan", PSex)),<br>is_parent(human(TmpName, PSex), human(Name, _)). | Запуск алгоритма унификации для is_parent(human(GName, f), human("Alexey", m)) и is_parent(human("Anatoly", m), human("Alexey", m)).<br>Унификация неуспешна.  | Откат, следующее предложение относительно 1   |
| 18-22 | ...   | ...  | ...   |
| 23    | is_grandparent(human(GName, f), _, "Ivan").   | Запуск алгоритма унификации для is_grandparent(human(GName, f), _, "Ivan"). и is_parent(human("Elena", f), human("Ivan", m)).<br>Унификация неуспешна.   | Прямой ход, переход к следующему предложению  |
| 24-27 | ...   | ...  | ...   |
| 28    | Пустая  | Запуск алгоритма унификации для is_grandparent(human(GName, f), _, "Ivan"). и is_parent(human("Anatoly", m), human("Alexey", m)).<br>Унификация неуспешна.   | Завершение работы.<br>Найдено 2 ответа.       |