



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 по курсу "Функциональное и логическое программирование"

Тема Структура программы на Prolog и ее реализация

Студент Цветков И.А.

Группа ИУ7-63Б

Оценка (баллы) _____

Преподаватели Толпинская Н. Б., Строганов Ю. В.

Москва — 2022 г.

1 Практические задания

Условие: Создать базу знаний «Собственники», дополнив (и минимально изменив) базу знаний, хранящую знания:

- **«Телефонный справочник»:** Фамилия, №тел, Адрес – структура(Город, Улица, №дома, №квартиры);
- **«Автомобили»:** Фамилия_владельца, Марка, Цвет, Стоимость, и др.;
- **«Вкладчики банков»:** Фамилия, Банк, счет, сумма, др.

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 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.

Листинг программы

```
1 DOMAINS
2     surname = symbol.
3
4     % Telephone
5     phoneNumber = symbol.
6     city = symbol.
7     street = symbol.
8     houseNum = integer.
9     apartmentNum = integer.
10    addressStruct = address(city, street, houseNum, apartmentNum).
11
12    % Car
13    brand = symbol.
14    color = symbol.
15    price = integer.
16
17    % Bank
18    bank = symbol.
19    bankAccount = integer.
20    sum = integer.
21
22    property = building(price, addressStruct);
23                area(price, integer);
24                car(brand, color, price);
25                waterTransport(price, symbol).
26
27 PREDICATES
28    phonebook(surname, phoneNumber, addressStruct).
29    bankDepositor(surname, city, bank, bankAccount, sum).
30    owner(surname, property).
31
32    ownObjects(surname, property).
33    ownObjectsPrice(surname, property, price).
34    ownPriceCount(surname, property, price).
35    ownFullPrice(surname, price).
36
37 CLAUSES
38    phonebook("Petrov", "8 (999) 888-77-66", address("Moscow", "Baumanskaya",
39                50, 23)).
40    phonebook("Ivanov", "8 (888) 777-66-55", address("Moscow", "Central", 23,
41                50)).
42    phonebook("Smirnov", "8 (777) 123-23-32", address("Moscow", "Baumanskaya",
43                50, 23)).
44    phonebook("Sergeev", "8 (999) 456-45-54", address("St. Petersburg",
45                "Petrovskaya", 45, 45)).
```

```

42  phonebook("Ivanov", "8 (888) 135-53-15", address("St. Petersburg",
    "Petrovskaya", 50, 23)).
43  phonebook("Smirnov", "8 (555) 555-55-55", address("St. Petersburg",
    "Dvorcovaya", 1, 5)).
44  phonebook("Petrov", "8 (123) 456-78-99", address("St. Petersburg",
    "Dvorcovaya", 5, 7)).
45
46  bankDepositor("Petrov", "Moscow", "Sberbank", 100000, 200000).
47  bankDepositor("Petrov", "St. Petersburg", "Tinkoff", 50000, 100000).
48  bankDepositor("Ivanov", "Moscow", "Sberbank", 25000, 300000).
49  bankDepositor("Ivanov", "St. Petersburg", "Alpha", 100000, 150000).
50  bankDepositor("Smirnov", "St. Petersburg", "Alpha", 300000, 500000).
51
52  owner("Ivanov", building(500000, address("Moscow", "Central", 23, 23))).
53  owner("Ivanov", car("Mercedes", "Black", 100000)).
54  owner("Ivanov", area(500000, 350)).
55  owner("Ivanov", waterTransport(300000, "Ship")).
56
57  owner("Petrov", building(500000, address("Moscow", "Baumanskaya", 50, 23))).
58  owner("Petrov", car("Mercedes", "red", 100000)).
59  owner("Petrov", waterTransport(10000, "Watercycle")).
60
61  owner("Smirnov", car("BMW", "Black", 50000)).
62  owner("Smirnov", area(250000, 200)).
63  owner("Smirnov", waterTransport(150000, "Ship")).
64
65  ownObjects(Surname, building) :- owner(Surname, building(_, _)).
66  ownObjects(Surname, car) :- owner(Surname, car(_, _, _)).
67  ownObjects(Surname, waterTransport) :- owner(Surname, waterTransport(_, _)).
68  ownObjects(Surname, area) :- owner(Surname, area(_, _)).
69
70  ownObjectsPrice(Surname, building, Price) :- owner(Surname, building(Price,
    _)).
71  ownObjectsPrice(Surname, car, Price) :- owner(Surname, car(_, _, Price)).
72  ownObjectsPrice(Surname, waterTransport, Price) :- owner(Surname,
    waterTransport(Price, _)).
73  ownObjectsPrice(Surname, area, Price) :- owner(Surname, area(Price, _)).
74
75  ownPriceCount(Surname, building, Price) :- owner(Surname, building(Price,
    _)), !.
76  ownPriceCount(Surname, car, Price) :- owner(Surname, car(_, _, Price)), !.
77  ownPriceCount(Surname, waterTransport, Price) :- owner(Surname,
    waterTransport(Price, _)), !.
78  ownPriceCount(Surname, area, Price) :- owner(Surname, area(Price, _)), !.
79  ownPriceCount(_, _, 0).
80
81  ownFullPrice(Surname, Price) :-
82      ownPriceCount(Surname, building, Price1),

```

```

83      ownPriceCount(Surname, car, Price2),
84      ownPriceCount(Surname, area, Price3),
85      ownPriceCount(Surname, waterTransport, Price4),
86      Price = Price1 + Price2 + Price3 + Price4.
87
88 GOAL
89      % Task 1
90      %% ownObjects("Ivanov", Object).
91      % Task 2
92      %% ownObjectsPrice("Petrov", Object, Price).
93      % Tssk 3
94      ownFullPrice("Smirnov", Price).

```

Выполнение заданий

Таблицы приложены в конце отчета

Таблица к заданию 2

Вопрос: *ownObjectsPrice("Petrov", Object, Price)*

№ шага	Сравнение термы, результат, подстановка, если есть	Дальнейшие действия, прямой ход или откат (к чему приводит?)
0		Состояние резольвенты: <i>ownObjectsPrice("Petrov", Object, Price)</i>
1	Сравнение: <i>ownObjectsPrice("Petrov", Object, Price) == phonebook("Petrov", "8 (999) 888-77-66", address("Moscow", "Baumanskaya", 50, 23)).</i> Унификация: неуспешна (несовпадение функторов)	Прямой ход Переход к следующему предложению
2-23
28	Сравнение: <i>ownObjectsPrice("Petrov", Object, Price) == ownObjectsPrice(Surname, building, Price)</i> Унификация: успешна Подстановка: {Surname = "Petrov", Object = building, Price = Price}	Образование новой резольвенты: 1. Редукция верхней подцели: замена <i>ownObjectsPrice("Petrov", Object, Price)</i> телом найденного правила Получена конъюнкция целей: <i>owner(Surname, building(Price, _)).</i> 2. Применение подстановки к полученной конъюнкции целей. Новое состояние резольвенты: <i>owner("Petrov", building(Price, _))</i>
29	Сравнение: <i>owner("Petrov", building(Price, _)) == phonebook("Petrov", "8 (999) 888-77-66", address("Moscow", "Baumanskaya", 50, 23))</i> Унификация: неуспешна (несовпадение функторов)	Прямой ход Переход к следующему предложению
21-44
45	Сравнение: <i>owner("Petrov", building(Price, _)) == owner("Petrov", building(500000, address("Moscow", "Baumanskaya", 50, 23))).</i> Унификация: успешна Подстановка: {Price = 500000}	Образование новой резольвенты: 1. Редукция верхней подцели: удаление <i>owner("Petrov", building(Price, _))</i> , так как найденное правило – факт Новое состояние резольвенты: пуста Решение найдено: формирование подстановки в качестве побочного эффекта: {Object = building, Price = 500000} Система должна найти все возможные ответы
		Обратная трассировка: 1) Отмена крайней редукции (на шаге 45)

		<p>2) Восстановление предыдущего состояния резольвенты (с шага 28): <code>owner("Petrov", building(Price, _)).</code></p> <p>3) Реконкретизация переменных с шага 45: <code>{Price = 500000}</code></p> <p>Переход к следующему предложению относительно шага 45</p>
46	<p>Сравнение: <code>owner("Petrov", building(Price, _)) == owner("Petrov", car("Mercedes", "red", 100000))</code></p> <p>Унификация: неуспешна (несовпадение термов)</p>	<p>Прямой ход Переход к следующему предложению</p>
47-69
70		<p>Конец БЗ Обратная трассировка: 1) Отмена крайней редукции (на шаге 28)</p> <p>2) Восстановление предыдущего состояния резольвенты: <code>ownObjectsPrice("Petrov", Object, Price)</code></p> <p>3) Реконкретизация переменных с шага 28: <code>{Surname = "Petrov", Object = building}</code></p> <p>Переход к следующему предложению относительно шага 28</p>
71	<p>Сравнение: <code>ownObjectsPrice("Petrov", Object, Price) == ownObjectsPrice(Surname, car, Price)</code></p> <p>Унификация: успешна</p> <p>Подстановка: <code>{Surname = "Petrov", Object = car, Price = Price}</code></p>	<p>Образование новой резольвенты:</p> <p>1. Редукция верхней подцели: замена <code>ownObjectsPrice("Petrov", Object, Price)</code> телом найденного правила</p> <p>Получена конъюнкция целей: <code>owner(Surname, car(_, _, Price))</code></p> <p>2. Применение подстановки к полученной конъюнкции целей.</p> <p>Новое состояние резольвенты: <code>owner("Petrov", car(_, _, Price))</code></p>
72	<p>Сравнение: <code>owner("Petrov", car(_, _, Price)) == phonebook("Petrov", "8 (999) 888-77-66", address("Moscow", "Baumanskaya", 50, 23))</code></p> <p>Унификация: неуспешна (несовпадение функторов)</p>	<p>Прямой ход Переход к следующему предложению</p>
73-89
90	<p>Сравнение: <code>owner("Petrov", car(_, _, Price)) == owner("Petrov", car("Mercedes", "red", 100000)).</code></p> <p>Унификация: успешна</p> <p>Подстановка: <code>{Price = 100000}</code></p>	<p>Образование новой резольвенты:</p> <p>1. Редукция верхней подцели: удаление <code>owner("Petrov", car(_, _, Price))</code>, так как найденное правило – факт</p> <p>Новое состояние резольвенты: пуста</p> <p>Решение найдено: формирование</p>

		<p>подстановки в качестве побочного эффекта: $\{Object = car, Price = 100000\}$</p> <p>Система должна найти все возможные ответы</p>
		<p>Обратная трассировка:</p> <p>1) Отмена крайней редукции (на шаге 90)</p> <p>2) Восстановление предыдущего состояния резольвенты (с шага 71): $owner("Petrov", car(_, _, Price))$</p> <p>3) Реконкретизация переменных с шага 90: $\{Price = 100000\}$</p> <p>Переход к следующему предложению относительно шага 90</p>
91	<p>Сравнение: $owner("Petrov", car(_, _, Price)) == owner("Petrov", waterTransport(10000, "Watercycle"))$</p> <p>Унификация: неуспешна (несовпадение термов)</p>	<p>Прямой ход</p> <p>Переход к следующему предложению</p>
92-108
109		<p>Конец БЗ</p> <p>Обратная трассировка:</p> <p>1) Отмена крайней редукции (на шаге 71)</p> <p>2) Восстановление предыдущего состояния резольвенты: $ownObjectsPrice("Petrov", Object, Price)$</p> <p>3) Реконкретизация переменных с шага 71: $\{Surname = "Petrov", Object = car\}$</p> <p>Переход к следующему предложению относительно шага 71</p>
110	<p>Сравнение: $ownObjectsPrice("Petrov", Object, Price) == ownObjectsPrice(Surname, waterTransport, Price)$</p> <p>Унификация: успешна</p> <p>Подстановка: $\{Surname = "Petrov", Object = waterTransport, Price = Price\}$</p>	<p>Образование новой резольвенты:</p> <p>1. Редукция верхней подцели: замена $ownObjectsPrice("Petrov", Object, Price)$ телом найденного правила</p> <p>Получена конъюнкция целей: $owner(Surname, waterTransport(Price, _))$</p> <p>2. Применение подстановки к полученной конъюнкции целей.</p> <p>Новое состояние резольвенты: $owner("Petrov", waterTransport(Price, _))$</p>
111	<p>Сравнение: $owner("Petrov", waterTransport(Price, _)) == phonebook("Petrov", "8 (999) 888-77-66", address("Moscow", "Baumanskaya", 50, 23))$</p> <p>Унификация: неуспешна (несовпадение функторов)</p>	<p>Прямой ход</p> <p>Переход к следующему предложению</p>
112-129

130	<p>Сравнение: <code>owner("Petrov", waterTransport(Price, _)) == owner("Petrov", waterTransport(10000, "Watercycle"))</code></p> <p>Унификация: успешна</p> <p>Подстановка: <code>{Price = 10000}</code></p>	<p>Образование новой резолюенты:</p> <p>1. Редукция верхней подцели: удаление <code>owner("Petrov", waterTransport(Price, _))</code>, так как найденное правило – факт</p> <p>Новое состояние резолюенты: пуста</p> <p>Решение найдено: формирование подстановки в качестве побочного эффекта: <code>{Object = waterTransport, Price = 10000}</code></p> <p>Система должна найти все возможные ответы</p>
		<p>Обратная трассировка:</p> <p>1) Отмена крайней редукции (на шаге 130)</p> <p>2) Восстановление предыдущего состояния резолюенты (с шага 110): <code>owner("Petrov", waterTransport(Price, _))</code></p> <p>3) Реконкретизация переменных с шага 110: <code>{Price = 10000}</code></p> <p>Переход к следующему предложению относительно шага 130</p>
131	<p>Сравнение: <code>owner("Petrov", waterTransport(Price, _)) == owner("Smirnov", car("BMW", "Black", 50000))</code></p> <p>Унификация: неуспешна (несовпадение термов)</p>	<p>Прямой ход</p> <p>Переход к следующему предложению</p>
132-147
148		<p>Конец БЗ</p> <p>Обратная трассировка:</p> <p>1) Отмена крайней редукции (на шаге 110)</p> <p>2) Восстановление предыдущего состояния резолюенты: <code>ownObjectsPrice("Petrov", Object, Price)</code></p> <p>3) Реконкретизация переменных с шага 110: <code>{Surname = "Petrov", Object = waterTransport}</code></p> <p>Переход к следующему предложению относительно шага 110</p>
149	<p>Сравнение: <code>ownObjectsPrice("Petrov", Object, Price) == ownObjectsPrice(Surname, area, Price)</code></p> <p>Унификация: успешна</p> <p>Подстановка: <code>{Surname = "Petrov", Object = area, Price = Price}</code></p>	<p>Образование новой резолюенты:</p> <p>1. Редукция верхней подцели: замена <code>ownObjectsPrice("Petrov", Object, Price)</code> телом найденного правила</p> <p>Получена конъюнкция целей: <code>owner(Surname, area(Price, _))</code></p> <p>2. Применение подстановки к полученной конъюнкции целей.</p> <p>Новое состояние резолюенты: <code>owner("Petrov", area(Price, _))</code></p>

150	<p>Сравнение: <code>owner("Petrov", area(Price, _)) == phonebook("Petrov", "8 (999) 888-77-66", address("Moscow", "Baumanskaya", 50, 23))</code></p> <p>Унификация: неуспешна (несовпадение функторов)</p>	<p>Прямой ход Переход к следующему предложению</p>
151-186
187		<p>Конец БЗ Обратная трассировка: 1) Отмена крайней редукции (на шаге 149) 2) Восстановление предыдущего состояния резольвенты: <code>ownObjectsPrice("Petrov", Object, Price)</code> 3) Реконкретизация переменных с шага 149: <code>{Surname = "Petrov", Object = area}</code></p> <p>Переход к следующему предложению относительно шага 149</p>
188	<p>Сравнение: <code>ownObjectsPrice("Petrov", Object, Price) == ownPriceCount(Surname, building, Price)</code></p> <p>Унификация: неуспешна (несовпадение функторов)</p>	<p>Прямой ход Переход к следующему предложению</p>
189-193
194		<p>Конец БЗ Обратная трассировка: 1) Отмена крайней редукции (на шаге 0) 2) Восстановление предыдущего состояния резольвенты: резольвента пуста</p> <p>Завершение работы На вопрос удалось ответить утвердительно 3 подстановки были возвращены в качестве побочного эффекта</p>