



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 №15

*По предмету: «Функциональное и логическое
программирование»*

Преподаватель: Толпинская Н.Б

Студент: Турсунов Ж.Р.,

Группа: ИУ7-66Б

Москва, 2021 г.

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 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)

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1...	-попытка унификации: T1=T2 ... -результат: Успех и подстановка, или Нет	Комментарий, вывод...
2

При желании, можно усложнить свою базу знаний, введя варианты: **строение**: (Дом, офис, торговый центр), **участок**: (садовый, территория под застройку, территория под агро-работы), **Водный_транспорт**: варианты названий.

Текст программы

domains

```
address = address(symbol City, symbol Street, integer House_num, integer Flat_num).

property = auto_own(symbol Brand, symbol Color, integer Cost);
building_own(symbol Type, integer Floors, integer Cost);
land_own(symbol Type, integer Plottage, integer Cost);
water_transport_own(symbol Type, integer Cost).
```

predicates

```
phone_book(symbol Surname, symbol Phone_num, address)
auto(symbol Surname, symbol Brand, symbol Color, integer Cost)
deposit(symbol Surname, symbol Bank, integer Bill, integer Sum)
```

```
owner(symbol Surname, property Property)
```

```
rule_1_2(symbol Surname, symbol Own_name, integer Cost)
```

clauses

```
phone_book(ivanov, "32-17-77", address(moscow, ilmenskiy, 14, 1018)).
phone_book(petrov, "37-87-70", address(pyatigosk, lenina, 73, 527)).
phone_book(sidorov, "98-36-83", address(moscow, tverskaya, 103, 17)).
phone_book(vasilev, "59-60-55", address(samara, lesnaya, 10, 7)).
```

```
auto(ivanov, bmw, white, 7800000).
auto(petrov, volvo, red, 1300000).
auto(sidorov, audi, green, 1600000).
```

```
deposit(ivanov, vtb, 00232973, 300000).
deposit(petrov, sberbank, 00101023, 36000).
deposit(sidorov, alfabank, 00014584, 200000).
```

```
owner(Surname, auto_own(Brand, Color, Cost)):- auto(Surname, Brand, Color, Cost).
owner(ivanov, building_own(supermarket, 1, 20000000)).
owner(petrov, building_own(townhouse, 3, 15000000)).
owner(petrov, land_own(field, 40, 4000000)).
owner(petrov, water_transport_own(hydrocycle, 200000)).
owner(vasiliev, water_transport_own(boat, 100000)).
```

```
rule_1_2(Surname, Brand, Cost):-owner(Surname, auto_own(Brand, _, Cost)).
rule_1_2(Surname, Type, Cost):-owner(Surname, building_own(Type, _, Cost)).
rule_1_2(Surname, Type, Cost):-owner(Surname, land_own(Type, _, Cost)).
rule_1_2(Surname, Type, Cost):-owner(Surname, water_transport_own(Type, Cost)).
```

goal

```
%rule_1_2(ivanov, Own_name, _).
%rule_1_2(petrov, Own_name, Cost).
```

Примеры работы программы

Задание 1. Поиск названий всех объектов собственности заданного субъекта.

```
goal  
    rule_1_2(ivanov, Own_name, _).
```

Вывод:

Own_name=bmw

Own_name=supermarket

2 Solutions

Задание 2. Поиск названий и стоимости всех объектов собственности заданного субъекта.

```
goal  
    rule_1_2(petrov, Own_name, Cost).
```

Вывод:

Own_name=volvo, Cost=1300000

Own_name=townhouse, Cost=15000000

Own_name=field, Cost=4000000

Own_name=hydrocycle, Cost=200000

4 Solutions

Описание порядка поиска объектов

Поиск названий и стоимости всех объектов собственности заданного субъекта.

goal

```
rule_1_2(petrov, Own_name, Cost) .
```

Вывод:

Own_name=volvo, Cost=1300000

Own_name=townhouse, Cost=15000000

Own_name=field, Cost=4000000

Own_name=hydrocycle, Cost=200000

4 Solutions

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	<p>Попытка унификации:</p> <pre>rule_1_2(petrov, Own_name, Cost) = phone_book(ivanov, "32-17-77", address(moscow, ilmenskiy, 14, 1018))</pre> <p>Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
2	<p>Попытка унификации:</p> <pre>rule_1_2(petrov, Own_name, Cost) = phone_book(petrov, "37-87-70", address(pyatigorsk, lenina, 73, 527))</pre> <p>Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
3	<p>Попытка унификации:</p> <pre>rule_1_2(petrov, Own_name, Cost) = phone_book(sidorov, "98-36-83", address(moscow, tverskaya, 103, 17))</pre> <p>Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
4	<p>Попытка унификации:</p> <pre>rule_1_2(petrov, Own_name, Cost) = phone_book(vasilev, "59-60-55", address(samara, lesnaya, 10, 7))</pre> <p>Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
5	<p>Попытка унификации:</p> <pre>rule_1_2(petrov, Own_name, Cost) = auto(ivanov, bmw, white, 7800000)</pre> <p>Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
6	<p>Попытка унификации:</p> <pre>rule_1_2(petrov, Own_name, Cost) = auto(petrov, volvo, red, 1300000)</pre>	Прямой ход, переход к следующему предложению

	Результат: Неудача (разные функторы).	
7	Попытка унификации: <code>rule_1_2(petrov, Own_name, Cost)</code> <code>=</code> <code>auto(sidorov, audi, green, 1600000)</code> Результат: Неудача (разные функторы).	Прямой ход, переход к следующему предложению
8	Попытка унификации: <code>rule_1_2(petrov, Own_name, Cost)</code> <code>=</code> <code>deposit(ivanov, vtbbank, 00232973, 300000)</code> Результат: Неудача (разные функторы).	Прямой ход, переход к следующему предложению
9	Попытка унификации: <code>rule_1_2(petrov, Own_name, Cost)</code> <code>=</code> <code>deposit(petrov, sberbank, 00101023, 36000)</code> Результат: Неудача (разные функторы).	Прямой ход, переход к следующему предложению
10	Попытка унификации: <code>rule_1_2(petrov, Own_name, Cost)</code> <code>=</code> <code>deposit(sidorov, alfabank, 00014584, 200000)</code> Результат: Неудача (разные функторы).	Прямой ход, переход к следующему предложению
11	Попытка унификации: <code>rule_1_2(petrov, Own_name, Cost)</code> <code>=</code> <code>owner(Surname, auto_own(Brand, Color, Cost))</code> Результат: Неудача (разные функторы).	Прямой ход, переход к следующему предложению
...	Производится попытка унификации со всеми предложениями <code>owner(...)</code> Результат: Неудача (разные функторы).	Прямой ход, переход к следующему предложению
17	Попытка унификации: <code>rule_1_2(petrov, Own_name, Cost)</code> <code>=</code> <code>rule_1_2(Surname, Brand, Cost)</code> Результат: Успех (выполняется подстановка) <code>{ Surname = petrov, Brand = Own_name, Cost = Cost}.</code>	Прямой ход. Сравнение начинается с начала с предложением <code>owner(petrov, auto_own(Own_name, _, Cost)).</code>
18	Попытка унификации: <code>owner(petrov, auto_own(Own_name, _, Cost))</code> <code>=</code> <code>phone_book(ivanov, "32-17-77", address(moscow, ilmenskiy, 14, 1018))</code> Результат: Неудача (разные функторы).	Прямой ход, переход к следующему предложению
...	Производится попытка унификации со всеми предложениями <code>phone_book(...)</code> Результат: Неудача (разные функторы).	Прямой ход, переход к следующему предложению
22	Попытка унификации: <code>owner(petrov, auto_own(Own_name, _, Cost))</code> <code>=</code> <code>auto(ivanov, bmw, white, 7800000)</code> Результат: Неудача (разные функторы).	Прямой ход, переход к следующему предложению

...	<p>Производится попытка унификации со всеми предложениями <code>auto (...)</code> Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
25	<p>Попытка унификации: <code>owner(petrov, auto_own(Own_name, _, Cost))</code> <code>=</code> <code>deposit(ivanov, vtb, 00232973, 300000)</code> Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
...	<p>Производится попытка унификации со всеми предложениями <code>deposit (...)</code> Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
28	<p>Попытка унификации: <code>owner(petrov, auto_own(Own_name, _, Cost))</code> <code>=</code> <code>owner(Surname, auto_own(Brand, Color, Cost))</code> Результат: Успех (выполняется подстановка) <code>{ Surname = petrov, Brand = Own_name, Cost = Cost }.</code></p>	Прямой ход. Сравнение начинается с начала с предложением <code>auto(petrov, Own_name, _, Cost).</code>
29	<p>Попытка унификации: <code>auto(petrov, Own_name, _, Cost).</code> <code>=</code> <code>phone_book(ivanov, "32-17-77", address(moscow, ilmenskiy, 14, 1018))</code> Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
...	<p>Производится попытка унификации со всеми предложениями <code>phone_book (...)</code> Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
33	<p>Попытка унификации: <code>auto(petrov, Own_name, _, Cost).</code> <code>=</code> <code>auto(ivanov, bmw, white, 7800000).</code> Результат: Неудача (<code>petrov ≠ ivanov</code>).</p>	Прямой ход, переход к следующему предложению
34	<p>Попытка унификации: <code>auto(petrov, Own_name, _, Cost).</code> <code>=</code> <code>auto(petrov, volvo, red, 1300000).</code> Результат: Успех (выполняется подстановка) <code>{ Own_name = volvo, Cost = 1300000 }.</code></p>	Вывод: <code>Own_name = volvo,</code> <code>Cost = 1300000</code> Откат к <code>auto(petrov, Own_name, _, Cost).</code>
35	<p>Попытка унификации: <code>auto(petrov, Own_name, _, Cost).</code> <code>=</code> <code>auto(sidorov, audi, green, 1600000).</code> Результат: Неудача (<code>petrov ≠ sidorov</code>).</p>	Прямой ход, переход к следующему предложению
...	<p>Производится попытка унификации со всеми предложениями <code>deposit(...)</code>, <code>owner(...)</code> и <code>rule_1_2(...)</code> Результат: Неудача (разные функторы).</p>	Откат к шагу 28

49	<p>Попытка унификации: <code>owner(petrov, auto_own(Own_name, _, Cost))</code> <code>=</code> <code>owner(ivanov, building_own(supermarket, 1, 20000000))</code> Результат: Неудача (<code>petrov ≠ ivanov</code>).</p>	Прямой ход, переход к следующему предложению
...	<p>Производится попытка унификации со всеми предложениями <code>owner(...)</code> и <code>rule_1_2(...)</code> Результат: Неудача</p>	Откат к шагу 17
58	<p>Попытка унификации: <code>rule_1_2(petrov, Own_name, Cost)</code> <code>=</code> <code>rule_1_2(Surname, Type, Cost)</code> Результат: Успех (выполняется подстановка) <code>{ Surname = petrov, Type = Own_name, Cost = Cost}</code>.</p>	Прямой ход. Сравнение начинается с начала с предложением <code>owner(petrov, building_own(Own_name, _, Cost))</code> .
59	<p>Попытка унификации: <code>owner(petrov, building_own(Own_name, _, Cost))</code> <code>=</code> <code>phone_book(ivanov, "32-17-77", address(moscow, ilmenskiy, 14, 1018))</code> Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
...	<p>Производится попытка унификации со всеми предложениями <code>phone_book(...)</code>, <code>auto(...)</code> и <code>deposit(...)</code> Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению
69	<p>Попытка унификации: <code>owner(petrov, building_own(Own_name, _, Cost))</code> <code>=</code> <code>owner(Surname, auto_own(Brand, Color, Cost))</code> Результат: Неудача (<code>building_own ≠ auto_own</code>).</p>	Прямой ход, переход к следующему предложению
70	<p>Попытка унификации: <code>owner(petrov, building_own(Own_name, _, Cost))</code> <code>=</code> <code>owner(ivanov, building_own(supermarket, 1, 20000000))</code> Результат: Неудача (<code>petrov ≠ ivanov</code>).</p>	Прямой ход, переход к следующему предложению
71	<p>Попытка унификации: <code>owner(petrov, building_own(Own_name, _, Cost))</code> <code>=</code> <code>owner(petrov, building_own(townhouse, 3, 15000000))</code> Результат: Успех (выполняется подстановка) <code>{ Own_name = townhouse, Cost = 15000000}</code>.</p>	Вывод: <code>Own_name = townhouse,</code> <code>Cost = 15000000</code> Откат к <code>owner(petrov, building_own(Own_name, _, Cost))</code>
...	<p>Попытка унификации: Результат: Неудача при сравнении со следующими термами до конца БЗ.</p>	Откат к 58

78	<p>Попытка унификации: <code>rule_1_2(petrov, Own_name, Cost)</code> <code>=</code> <code>rule_1_2(Surname, Type, Cost)</code> Результат: Успех (выполняется подстановка) <code>{ Surname = petrov, Type = Own_name, Cost = Cost}.</code></p>	<p>Прямой ход. Сравнение начинается с начала с предложением <code>owner(petrov, land_own(Own_name, _, Cost)).</code></p>
79	<p>Попытка унификации: <code>owner(petrov, land_own(Own_name, _, Cost))</code> <code>=</code> <code>phone_book(ivanov, "32-17-77", address(moscow, ilmenskiy, 14, 1018)).</code> Результат: Неудача (разные функторы).</p>	<p>Прямой ход, переход к следующему предложению</p>
...	<p>Производится попытка унификации со всеми предложениями <code>phone_book(...)</code>, <code>auto(...)</code> и <code>deposit(...)</code> Результат: Неудача (разные функторы).</p>	<p>Прямой ход, переход к следующему предложению</p>
89	<p>Попытка унификации: <code>owner(petrov, land_own(Own_name, _, Cost))</code> <code>=</code> <code>owner(Surname, auto_own(Brand, Color, Cost))</code> Результат: Неудача (<code>land_own</code> \neq <code>auto_own</code>).</p>	<p>Прямой ход, переход к следующему предложению</p>
90	<p>Попытка унификации: <code>owner(petrov, land_own(Own_name, _, Cost))</code> <code>=</code> <code>owner(ivanov, building_own(supermarket, 1, 20000000))</code> Результат: Неудача (<code>land_own</code> \neq <code>building_own</code>).</p>	<p>Прямой ход, переход к следующему предложению</p>
91	<p>Попытка унификации: <code>owner(petrov, land_own(Own_name, _, Cost))</code> <code>=</code> <code>owner(petrov, land_own(field, 40, 4000000))</code> Результат: Успех (выполняется подстановка) <code>{ Own_name = field, Cost = Cost}.</code></p>	<p>Вывод: <code>Own_name = field,</code> <code>Cost = 4000000</code> Откат к <code>owner(petrov, land_own(Own_name, _, Cost))</code></p>
...	<p>Попытка унификации: Результат: Неудача при сравнении со следующими термами до конца БЗ.</p>	<p>Откат к 78</p>
98	<p>Попытка унификации: <code>rule_1_2(petrov, Own_name, Cost)</code> <code>=</code> <code>rule_1_2(Surname, Type, Cost)</code> Результат: Успех (выполняется подстановка) <code>{ Surname = petrov, Type = Own_name, Cost = Cost}.</code></p>	<p>Прямой ход. Сравнение начинается с начала с предложением <code>owner(petrov, water_transport_own(Own_name, Cost)).</code></p>
99	<p>Попытка унификации: <code>owner(petrov, water_transport_own(Own_name, Cost))</code> <code>=</code> <code>phone_book(ivanov, "32-17-77", address(moscow, ilmenskiy, 14, 1018))</code> Результат: Неудача (разные функторы).</p>	<p>Прямой ход, переход к следующему предложению</p>
...	<p>Производится попытка унификации со всеми предложениями <code>phone_book(...)</code>, <code>auto(...)</code> и <code>deposit(...)</code> Результат: Неудача (разные функторы).</p>	<p>Прямой ход, переход к следующему предложению</p>

110	<p>Попытка унификации: <code>owner(petrov, water_transport_own(Own_name, Cost))</code> <code>=</code> <code>owner(Surname, auto_own(Brand, Color, Cost))</code> Результат: Неудача (<code>water_transport_own ≠ auto_own</code>).</p>	Прямой ход, переход к следующему предложению
111	<p>Попытка унификации: <code>owner(petrov, water_transport_own(Own_name, Cost))</code> <code>=</code> <code>owner(ivanov, building_own(supermarket, 1, 20000000))</code> Результат: Неудача (<code>petrov ≠ ivanov</code>).</p>	Прямой ход, переход к следующему предложению
112	<p>Попытка унификации: <code>owner(petrov, water_transport_own(Own_name, Cost))</code> <code>=</code> <code>owner(petrov, building_own(townhouse, 3, 15000000))</code> Результат: Неудача (<code>water_transport_own ≠ building_own</code>).</p>	Прямой ход, переход к следующему предложению
113	<p>Попытка унификации: <code>owner(petrov, water_transport_own(Own_name, Cost))</code> <code>=</code> <code>owner(petrov, land_own(field, 40, 4000000))</code> Результат: Неудача (<code>water_transport_own ≠ land_own</code>).</p>	Прямой ход, переход к следующему предложению
114	<p>Попытка унификации: <code>owner(petrov, water_transport_own(Own_name, Cost))</code> <code>=</code> <code>owner(petrov, water_transport_own(hydrocycle, 200000))</code> Результат: Успех (выполняется подстановка) <code>{ Own_name = hydrocycle, Cost = 200000 }.</code></p>	Вывод: <code>Own_name = hydrocycle,</code> <code>Cost = 4000000</code> Откат к <code>owner(petrov, water_transport_own(Own_name, Cost))</code>
...	<p>Попытка унификации: <code>owner(petrov, water_transport_own(Own_name, Cost))</code> и <code>rule_1_2(...)</code> Результат: Неудача (разные функторы).</p>	Прямой ход, переход к следующему предложению.
118	<p>Попытка унификации: <code>owner(petrov, water_transport_own(Own_name, Cost))</code> и <code>rule_1_2(Surname, Type, Cost)</code> Результат: Неудача (разные функторы).</p>	Откат к шагу 98, а затем достижение конца БЗ. Завершение программы.

Теоретические вопросы.

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

Знания сформулированы в разделе CLAUSES в виде предложений (факты и правила). Они представляют собой знания о некоторой предметной области, формально – отношения между различными объектами.

2. Что содержит тело правила?

Тело правила содержит условие истинности заголовка правила.

- 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?**

Использование переменных в формулировании знаний позволяют уточнять значения и переносить их «в пространстве и времени». Формулировка знаний с использованием переменных носит более общий характер по отношению к знанию, состоящему только лишь из констант. Использование знаний с одинаковой арностью при использовании одной переменной носит менее общий характер по отношению к знанию с использованием нескольких переменных.

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

Переменные входят в правило с квантором всеобщности («для любого»). Именованные переменные уникальны в пределах одного предложения, анонимные уникальны везде.

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

Предложения в разделе DOMAINS используются для объявления используемых доменов, неявляющимися стандартными доменами в Prolog. Раздел доменов используется для описания структур (вариантных доменов).

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

В разделе PREDICATES описываются предикаты, их арность (местность) и домены (типы и природа аргументов). С помощью описанных предикатов, можно создавать предложения в базе знаний. Предикаты используются для представления, как фактов, так и правил.

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

На первом шаге работы происходит унификация вопроса и первого предложения базы знаний. Работа алгоритма унификации заключается в попарном сопоставлении термов и попытке построить для них общий пример. Назначение -

подбор нужного в данный момент правила. Унификация может завершаться успехом или тупиковой ситуацией (неудачей).

8. В каком случае запускается механизм отката?

Механизм отката запускается, если возникла тупиковая ситуация (достигнут конец БЗ) либо резольвента пуста. В таких случаях происходит откат к предыдущему состоянию резольвенты.