



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» (ИУ7)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.04 Программная инженерия

О Т Ч Е Т

по лабораторной работе № 1 5

Дисциплина: Функциональное и логическое программирование

Студент

ИУ7-62Б

(Группа)

(Подпись, дата)

Е.В. Брянская

(И.О. Фамилия)

Преподаватель

Н.Б.Толпинская

Ю.В.Строганов

(Подпись, дата)

(И.О. Фамилия)

Москва, 2021

Задание

Создать базу знаний «Собственники», переформатировать имеющуюся, используя вариантный домен.

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

Вид собственности:

- Строение, стоимость и др.
- Участок, стоимость и др.
- Водный_транспорт, стоимость и др.

Владелец может иметь только один объект каждого вида собственности, 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 2ого пункта и одной фамилии составить таблицу, отражающую конкретный порядок работы системы, с объяснениями порядка работы и особенностей использования доменов (указать конкретные T1, T2 и полную подстановку на каждом шаге)

При желании можно усложнить свою БЗ, введя варианты:

Строение: (Дом, офис, торговый центр)

Участок: (садовый, территория под застройку, территория под агро-работы)

Водный транспорт: (варианты названий)

DOMAINS

surname, phone = symbol.

city, street = symbol.

home, flat = integer.

brand, color = symbol.

price = real.

years = integer.

bank, account = symbol.

sum = real.

address = address(city, street, home, flat).

s = real.

property = building(price, s);

area(price, s);

water_transport(price, years);

phone_book(phone, address);

car(brand, color, price, years).

cur_property = symbol.

PREDICATES

investor(surname, bank, account, sum).

phone_book(surname, phone, address).

own(surname, property).

property_price_by_owner(surname, cur_property, price).

get_price_by_owner(surname, cur_property, price).

sum_price(Surname, sum)

CLAUSES

investor("Birukova", "New_1", deposit, 1000000).

investor("Mironova", "Old_bank", deposit, 5000).

investor("Nikiforov", "VTB", deposit, 5000).

phone_book("Birukova", "+123456", address("Moscow", "Zhukovsky Street", 12, 145)).

phone_book("Mironova", "+333333", address("Ramenskoe", "New Street", 77, 77)).

own("Birukova", building(1000000, 50)).

own("Birukova", area(750000, 120)).

own("Birukova", water_transport(5000000, 1)).

own("Filin", water_transport(55000000, 2)).

own("Birukova", car("BMW", black, 4500000, 2)).

own("Filin", car("Honda", grey, 2300000, 4)).

own("Nikiforov", car("Honda", grey, 2000000, 6)).

property_price_by_owner(Surname, building, Price) :- own(Surname, building(Price, _)).

property_price_by_owner(Surname, area, Price) :- own(Surname, area(Price, _)).

property_price_by_owner(Surname, water_transport, Price) :- own(Surname, water_transport(Price, _)).

property_price_by_owner(Surname, car, Price) :- own(Surname, car(_, _, Price, _)).

get_price_by_owner(Surname, building, Price) :- own(Surname, building(Price, _)), !.

<pre> get_price_by_owner(Surname, area, Price) :- own(Surname, area(Price, _)), !. get_price_by_owner(Surname, water_transport, Price) :- own(Surname, water_transport(Price, _)), !. get_price_by_owner(Surname, car, Price) :- own(Surname, car(_, _, Price, _)), !. get_price_by_owner(_, _, 0). sum_price(Surname, Sum) :- get_price_by_owner(Surname, building, Price1), get_price_by_owner(Surname, area, Price2), get_price_by_owner(Surname, water_transport, Price3), get_price_by_owner(Surname, car, Price4), Sum = Price1 + Price2 + Price3 + Price4. </pre>
<p>GOAL</p> <pre> %property_price_by_owner("Filin", Property, _). %property_price_by_owner("Filin", Property, Price). %sum_price("Nikiforov", Sum). </pre>

1. Найти названия всех объектов собственности заданного субъекта

property_price_by_owner("Birukova ", Property, _).	Property=building Property=area Property=water_transport Property=car 4 Solutions
property_price_by_owner("Filin", Property, _).	Property=water_transport Property=car 2 Solutions
property_price_by_owner("Mishina", Property, _).	No Solution

2. Найти названия и стоимости всех объектов собственности заданного субъекта

property_price_by_owner("Birukova", Property, Price).	Property=building, Price=1000000 Property=area, Price=750000 Property=water_transport, Price=5000000 Property=car, Price=4500000 4 Solutions
property_price_by_owner("Filin", Property, Price).	Property=water_transport, Price=55000000 Property=car, Price=2300000 2 Solutions
property_price_by_owner("Mishina", Property, Price).	No Solution

3. Разработать правило, позволяющее найти суммарную стоимость всех объектов собственности заданного субъекта

sum_price("Birukova", Sum).	Sum=11250000 1 Solution
sum_price("Filin", Sum).	Sum=57300000 1 Solution
sum_price("Mishina", Sum).	Sum=0 1 Solution

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

№	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действие: прямой ход или откат (к чему приводит?)
0.		Начальное состояние резольвенты: property_price_by_owner("Filin", Propertyt, Pricet)
1.	property_price_by_owner("Filin", Propertyt, Pricet) = investor("Birukova", "New_1", deposit, 1000000)	Неудача (не совпали главные функторы)
2.	***	***
3.	property_price_by_owner("Filin", Propertyt, Pricet) = phone_book("Birukova", "+123456", address("Moscow", "Zhukovsky Street", 12, 145))	Неудача (не совпали главные функторы)
4.	***	***
5.	property_price_by_owner("Filin", Propertyt, Pricet) = own("Birukova", building(1000000, 50))	Неудача (не совпали главные функторы)
6.	***	***
7.	property_price_by_owner("Filin", Propertyt, Pricet) = property_price_by_owner(Surname, building, Price) Постановка: {Surname="Filin", Propertyt=building, Price=Pricet}	Удача Изменение резольвенты: Новое состояние резольвенты: own("Filin", building(Pricet, _))
8.	own("Filin", building(Pricet, _)) = investor("Birukova", "New_1", deposit, 1000000)	Неудача (не совпали главные функторы)
9.	***	***
10.	own("Filin", building(Pricet, _)) = own("Birukova", building(1000000, 50))	Неудача (не совпали главные функторы)

11.	***	***
12.	own("Filin", building(Pricet, _)) = own("Filin", water_transport(55000000, 2))	Неудача (не совпали функторы)
13.	own("Filin", building(Pricet, _)) = own("Birukova", car("BMW", black, 4500000, 2))	Неудача (не совпали константы)
14.	own("Filin", building(Pricet, _)) = own("Filin", car("Honda", grey, 2300000, 4))	Неудача (не совпали функторы)
15.	***	***
16.	own("Filin", building(Pricet, _)) = sum_price(Surname, Sum)	Неудача (не совпали функторы) Откат (просмотрена вся БЗ)
17.	Подстановка: {}	Изменение состояния резольвенты: property_price_by_owner("Filin", Propertyt, Pricet)
18.	property_price_by_owner("Filin", Propertyt, Pricet) = property_price_by_owner(Surname, area, Price) Постановка: {Surname="Filin", Propertyt=area, Price=Pricet}	Удача Изменение резольвенты: Новое состояние резольвенты: own("Filin", area(Pricet, _))
19.	own("Filin", area(Pricet, _)) = investor("Birukova", "New_1", deposit, 1000000)	Неудача (не совпали главные функторы)
20.	***	***
21.	own("Filin", area(Pricet, _)) = sum_price(Surname, Sum)	Неудача (не совпали главные функторы) Откат (просмотрена вся БЗ)
22.	Подстановка: {}	Изменение состояния резольвенты: property_price_by_owner("Filin", Propertyt, Pricet)
23.	property_price_by_owner("Filin", Propertyt, Pricet)	Удача

	<p>= property_price_by_owner(Surname, water_transport, Price)</p> <p>Постановка: {Surname="Filin", Propertyt= water_transport, Price=Pricet}</p>	<p>Изменение резольвенты: Новое состояние резольвенты: own("Filin", water_transport(Pricet, _))</p>
24.	<p>own("Filin", water_transport(Pricet, _)) = investor("Birukova", "New_1", deposit, 1000000)</p>	<p>Неудача (не совпали главные функторы)</p>
25.	***	***
26.	<p>own("Filin", water_transport(Pricet, _)) = own("Filin", water_transport(55000000, 2))</p> <p>Постановка: {Surname="Filin", Propertyt= water_transport, Price=Pricet, Pricet=55000000}</p>	<p>Удача</p> <p>Изменение резольвенты: Новое состояние резольвенты: Резольвента пуста</p> <p>Выводится: Propertyt= water_transport, Pricet=55000000</p> <p>Откат</p>
27.	<p>Подстановка: {Surname="Filin", Propertyt= water_transport, Price=Pricet}</p>	<p>Изменение состояния резольвенты: own("Filin", water_transport(Pricet, _))</p>
28.	<p>own("Filin", water_transport(Pricet, _)) = own("Birukova", car("BMW", black, 4500000, 2))</p>	<p>Неудача (не совпали константы)</p>
29.	***	***
30.	<p>own("Filin", water_transport(Pricet, _)) = sum_price(Surname, Sum)</p>	<p>Неудача (не совпали главные функторы)</p> <p>Откат (просмотрена вся БЗ)</p>
31.	<p>Подстановка: {}</p>	<p>Изменение состояния резольвенты: property_price_by_owner("Filin", Propertyt, Pricet)</p>
32.	<p>property_price_by_owner("Filin", Propertyt, Pricet) = property_price_by_owner(Surname, car, Price)</p>	<p>Удача</p> <p>Изменение резольвенты: Новое состояние резольвенты: own("Filin", car(_, _, Pricet, _))</p>

	Постановка: {Surname="Filin", Propertyt= car, Price=Pricet}	
33.	own("Filin", car(_ , Pricet, _)) = investor("Birukova", "New_1", deposit, 1000000)	Неудача (не совпали главные функторы)
34.	***	***
35.	own("Filin", car(_ , Pricet, _)) = own("Filin", car("Honda", grey, 2300000, 4)) Постановка: {Surname="Filin", Propertyt=car, Price=Pricet, Pricet=2300000}	Удача Изменение резольвенты: Новое состояние резольвенты: Резольвента пуста Выводится: Propertyt=car, Pricet=2300000 Откат
36.	Подстановка: {Surname="Filin", Propertyt=car, Price=Pricet}	Изменение состояния резольвенты: own("Filin", car(Pricet, _))
37.	own("Filin", car(Pricet, _)) = own("Nikiforov", car("Honda", grey, 2000000, 6))	Неудача (не совпали константы)
38.	***	***
39.	own("Filin", car(Pricet, _)) = sum_price(Surname, Sum)	Неудача (не совпали главные функторы) Откат (просмотрена вся БЗ)
40.	Подстановка: {}	Изменение состояния резольвенты: property_price_by_owner("Filin", Propertyt, Pricet)
41.	property_price_by_owner("Filin", Propertyt, Pricet) = get_price_by_owner(Surname, building, Price)	Неудача (не совпали главные функторы)
42.	***	***
43.	property_price_by_owner("Filin", Propertyt, Pricet) = 	Неудача (не совпали главные функторы)

	sum_price(Surname, Sum)	Просмотрена вся БЗ Завершение работы
--	-------------------------	---

Вопросы:

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

Знание формулируется в разделе CLAUSES, это знания о предметной области.

2. Что содержит тело правила?

Тело правила содержит условие истинности.

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?

Переменные нужны для обобщения, в программе же необходимы для передачи значения в пространстве и во времени. Чем больше переменных, тем сильнее обобщённость, абстракция.

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

Переменные в правило входят с квантором всеобщности. Именованная переменная уникальна в рамках одного предложения, анонимная же уникальна всегда.

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

DOMAINS - раздел описания доменов, используется для определения структуры, позволяют описать природу аргументов.

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

PREDICATES - раздел описания предикатов, используется для отслеживания природы аргументов.

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

На самом первом шаге запускается унификация цели и первого терма. Унификация позволяет формализовать процесс логического вывода и обеспечивает двустороннюю передачу параметров процедурам, неразрушающее присваивание, проверку условий. Результат унификации – ответ да/нет.

8. В каком случае запускается механизм отката?

Механизм отката запускается, в случае если было найдено подходящее знание, позволяющее ответить «да» на поставленный вопрос (в таком случае система проверяет, нет ли еще решения), либо в случае, когда его, наоборот, найти не удалось, просмотрена вся база знаний.