|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **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**

*к лабораторной работе №13*

*По курсу: «Функциональное и логическое программирование»*

**Темы: «**Структура программы на Prolog и ее реализация»

Студент: Зайцева А. А.

Группа: ИУ7-62Б

Преподаватели: Толпинская Н. Б.,

Строганов Ю. В.

Москва, 2022 г.

# Практическая часть

Задание. Создать базу знаний **«Собственники»**, дополнив базу знаний, хранящую знания (лаб. 12):

* **«Телефонный справочник»:** Фамили, №тел, Адрес – структура (Город, Улица, №дома, №кв),
* **«Автомобили»:** Фамилия\_владельца, Марка, Цвет, Стоимость, и др.,
* **«Вкладчики банков»:** Фамилия, Банк, счёт, 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)

|  |
| --- |
| domains  city, street = symbol.  house, flat = integer.  address\_struct = address(city, street, house, flat).  surname = symbol.  phone = integer.  model, color = symbol.  price, year = integer.  bank = symbol.  sum, account = integer.  size = integer.    ownnership = building(price, address\_struct);  area(price, size);  water\_transport(price, color);  car(price, model, color).    predicates  phone\_record(surname, phone, address\_struct).  depositor(surname, bank, account, sum).    own(surname, ownnership).    ownershipsNamePrice(surname, symbol, price).  sumCostInner(surname, symbol, price).  sumCost(surname, price).  clauses  phone\_record(rich, 7777772, address(london, green, 1, 10)).  phone\_record(rich, 7777771, address(london, green, 1, 10)).  phone\_record(rich, 1111111, address(moscow, zelenaya, 2, 20)).  phone\_record(middle, 9999999, address(moscow, ivanovskaya, 3, 2)).  phone\_record(poor, 3333331, address(karaganda, pit, 23, 5)).  phone\_record(poor, 3333332, address(perm, pit, 36, 7)).  phone\_record(poor, 3333333, address(kop, leet, 2, 53)).      depositor(rich, gosbank, 10, 10000000).  depositor(rich, mosbank, 15, 9000000).  depositor(middle, mosbank, 17, 20000).  depositor(middle, newbank, 345, 0).      own(rich, building(100, address(kop, leet, 2, 53))).  own(rich, area(10, 500)).  own(rich, water\_transport(1, green)).  own(rich, car(5, model1, red)).    own(middle, car(3, model2, red)).  own(middle, building(90, address(moscow, leninskaya, 2, 53))).    % 1, 2  ownershipsNamePrice(Surname\_, building, Price\_) :- own(Surname\_, building(Price\_, \_)).  ownershipsNamePrice(Surname\_, area, Price\_) :- own(Surname\_, area(Price\_, \_)).  ownershipsNamePrice(Surname\_, water\_transport, Price\_) :- own(Surname\_, water\_transport(Price\_, \_)).  ownershipsNamePrice(Surname\_, car, Price\_) :- own(Surname\_, car(Price\_, \_, \_)).    % 3  sumCostInner(Surname\_, building, Price\_) :- own(Surname\_, building(Price\_, \_)), !.  sumCostInner(Surname\_, area, Price\_) :- own(Surname\_, area(Price\_, \_)), !.  sumCostInner(Surname\_, water\_transport, Price\_) :- own(Surname\_, water\_transport(Price\_, \_)), !.  sumCostInner(Surname\_, car, Price\_) :- own(Surname\_, car(Price\_, \_, \_)), !.  sumCostInner(\_, \_, 0). % if Surname\_ has no some kind of ownership    sumCost(Surname\_, Price\_) :-  sumCostInner(Surname\_, building, Price1\_),  sumCostInner(Surname\_, area, Price2\_),  sumCostInner(Surname\_, water\_transport, Price3\_),  sumCostInner(Surname\_, car, Price4\_),  Price\_ = Price1\_ + Price2\_ + Price3\_ + Price4\_.  goal  % 1  % ownershipsNamePrice(rich, Ownnership, \_).  % Ownnership=building  % Ownnership=area  % Ownnership=water\_transport  % Ownnership=car  % 4 Solutions    % ownershipsNamePrice(Surname, car, \_).  % Surname=rich  % Surname=middle  % 2 Solutions      % 2  ownershipsNamePrice(middle, Ownnership, Price).  % Ownnership=building, Price=90  % Ownnership=car, Price=3  % 2 Solutions  % ownershipsNamePrice(Surname, building, Price).  % Surname=rich, Price=100  % Surname=middle, Price=90  % 2 Solutions      % 3  % sumCost(middle, Price). % =sumCost("middle", Price).  % Price=93  % 1 Solution    % sumCost(rich, Price).  % Price=116  % 1 Solution  % sumCost(poor, Price).  % Price=0  % 1 Solution |

ownershipsNamePrice(middle, Ownnership, Price).

|  |  |  |
| --- | --- | --- |
| № шага | Сравниваемые термы; результат; подстановка, если есть | Дальнейшие действия: прямой ход или откат (к чему приводит?) |
| 0 | Результирующая ячейка пуста | Состояние резольвенты:  ownershipsNamePrice(middle, Ownnership, Price).  Поиск способа доказательства подцели с начала БЗ |
| 1 | **Сравниваемые термы:**  ownershipsNamePrice(middle, Ownnership, Price).  phone\_record(rich, 7777772, address(london, green, 1, 10)).**Результат:**  Унификация неуспешна | Прямой ход, переход к следующему предложению |
| 2-17 | … | … |
| 18 | **Сравниваемые термы:**  ownershipsNamePrice(middle, Ownnership, Price).  ownershipsNamePrice(Surname\_, building, Price\_)  **Результат:**  Унификация успешна  **Подстановка:**  {Ownnership= building, Surname\_= middle}  Результирующая ячейка:  {Ownnership= building} | Образование новой резольвенты:  1. Редукция верхней подцели: замена ownershipsNamePrice(middle, Ownnership, Price) телом найденного правила.  Полученная конъюнкция целей:  own(Surname\_, building(Price\_, \_)).  2. Применение подстановки к полученной  конъюнкции целей.  ***Новое состояние резольвенты:***  own(middle, building(Price\_, \_)).  Поиск способа доказательства новой подцели (верхней) с начала БЗ |
| 19 | **Сравниваемые термы:**  own(middle, building(Price\_, \_)).  phone\_record(rich, 7777772, address(london, green, 1, 10)).  **Результат:**  Унификация неуспешна | Прямой ход, переход к следующему предложению |
| 20-34 | … | … |
| 35 | **Сравниваемые термы:**  own(middle, building(Price\_, \_)).  own(middle, building(90, address(moscow, leninskaya, 2, 53))).  **Результат:**  Унификация успешна  **Подстановка:**  {Price\_=90}  Результирующая ячейка:  {Ownnership= building, Price=90} | Образование новой резольвенты:  Редукция верхней подцели: удаление own(middle, building(Price\_, \_)), так как найденное правило – факт (тело пустое).  ***Новое состояние резольвенты:***  пуста  Решение найдено: формирование подстановки {Ownnership= building, Price=90} в качестве побочного эффекта.  Система должна получить все возможные ответы |
|  |  | Откат  Отмена последней редукции (на шаге 35),  восстановление предыдущего состояния резольвенты:  own(middle, building(Price\_, \_)).  реконкретизация переменных, которые были конкретизированы на предыдущем шаге ({Price\_=90})  переход к следующему предложению относительно шага 35 |
| 36 | **Сравниваемые термы:**  own(middle, building(Price\_, \_)).  ownershipsNamePrice(Surname\_, building, Price\_)  **Результат:**  Унификация неуспешна | Прямой ход, переход к следующему предложению |
| 37-45 | … | … |
|  |  | Конец БЗ. Из данного состояния невозможен переход в новое состояние.  Откат  Отмена последней редукции (на шаге 18),  восстановление предыдущего состояния резольвенты  ownershipsNamePrice(middle, Ownnership, Price).  реконкретизация переменных, которые были конкретизированы на предыдущем шаге ({Ownnership= building})  переход к следующему предложению относительно шага 18 |
| 46 | **Сравниваемые термы:**  ownershipsNamePrice(middle, Ownnership, Price).  ownershipsNamePrice(Surname\_, area, Price\_)  **Результат:**  Унификация успешна  **Подстановка:**  {Ownnership= area, Surname\_= middle}  Результирующая ячейка:  {Ownnership= area} | Образование новой резольвенты:  1. Редукция верхней подцели: замена ownershipsNamePrice(middle, Ownnership, Price) телом найденного правила.  Полученная конъюнкция целей:  own(Surname\_, area(Price\_, \_))  2. Применение подстановки к полученной  конъюнкции целей.  ***Новое состояние резольвенты:***  own(middle, area(Price\_, \_))  Поиск способа доказательства новой подцели (верхней) с начала БЗ |
| 47 | **Сравниваемые термы:**  own(middle, area(Price\_, \_)).  phone\_record(rich, 7777772, address(london, green, 1, 10)).  **Результат:**  Унификация неуспешна | Прямой ход, переход к следующему предложению |
| 48-73 | **…** | … |
|  |  | Конец БЗ. Из данного состояния невозможен переход в новое состояние.  Откат  Отмена последней редукции (на шаге 46),  восстановление предыдущего состояния резольвенты:  ownershipsNamePrice(middle, Ownnership, Price)  реконкретизация переменных, которые были конкретизированы на предыдущем шаге ({Ownnership= area})  переход к следующему предложению относительно шага 46 |
|  |  |  |
|  | **…** | … |