|  |  |
| --- | --- |
| **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 = string.  street = string.  house = integer.  flat = integer.  address = address(city, street, house, flat).  surname = string.  phone = integer.  model = string.  color = string.  price = integer.  year = integer.  bank = string.  sum = integer.  account = integer.  size = integer.    ownnership = building(price, address);  area(price, size);  water\_transport(price, color);  car(price, model, color).    predicates  phone\_record(surname, phone, address).  depositor(surname, bank, account, sum).  own(surname, ownnership).    ownershipsName(surname, symbol).  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  ownershipsName(Surname, building) :- own(Surname, building(\_, \_)).  ownershipsName(Surname, area) :- own(Surname, area(\_, \_)).  ownershipsName(Surname, water\_transport) :- own(Surname, water\_transport(\_, \_)).  ownershipsName(Surname, car) :- own(Surname, car(\_, \_, \_)).  % 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).    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  % ownershipsName("Rich", Ownnership).  % Ownnership=building  % Ownnership=area  % Ownnership=water\_transport  % Ownnership=car  % 4 Solutions    % ownershipsName(Surname, car).  % Surname=Rich  % Surname=Middle  % 2 Solutions    % 2  % ownershipsNamePrice(Surname, building, Price).  % Surname=Rich, Price=100  % Surname=Middle, Price=90  % 2 Solutions    % ownershipsNamePrice("Middle", Ownnership, Price).  % Ownnership=building, Price=90  % Ownnership=car, Price=3  % 2 Solutions      % 3  % sumCost("Middle", Price).  % Price=93  % 1 Solution    % sumCost("Rich", Price).  % Price=116  % 1 Solution  sumCost("Poor", Price).  % Price=0  % 1 Solution |