|  |  |
| --- | --- |
| 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**

**по лабораторной работе № 15**

**Дисциплина: «Функциональное и логическое программирование»**

Выполнила: Овчинникова А. П.

Группа: ИУ7-65Б

Преподаватель: Толпинская Н. Б.

Строганов Ю. В.

Москва, 2020

## 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 Т1 и Т2 и полную подстановку на каждом шаге).

## Теоретическая часть

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

Заголовок содержит отдельное знание о предметной области (составной терм). Заголовок, как составной терм f(t1, t2, …,tm), содержит знание о том, что между аргументами t1, t2, …, tm существует отношение (взаимосвязь, взаимозависимость). А имя этого отношения – это f.

2. Что содержит тело правила?

Заголовок содержит отдельное знание о предметной области (составной терм), а тело содержит условия истинности этого знания.

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?

Чем больше переменных в правиле, тем более оно общее. При сравнении правил с одинаковой арностью и разным количеством переменных, более общим будет то, в котором больше переменных.

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

Переменные в факты и правила входят только с квантором всеобщности. В вопросы – только с квантором существования. Отметим, что именованные переменные уникальны в рамках предложения, а анонимная переменная – любая уникальна.

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

DOMAINS – раздел описания доменов. В разделе описания доменов объявляются любые нестандартные домены, используемые в качестве аргументов предикатов. В этом разделе определяются имена и семантика дополнительных доменов, если структура каких-либо объектов не может быть представлена с помощью стандартных доменов.

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

PREDICATES – раздел описания предикатов. Содержит описания определяемых пользователем предикатов. Домены аргументов предикатов должны быть либо стандартными, либо объявленными в разделе описания доменов. В раздел PREDICATED попадают только предикаты, не являющиеся стандартными, то есть те, которые отсутствуют в самом языке программирования.

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

Назначение унификации – подобрать нужное в данный момент правило. Система подбирает сопоставимые с целью правила с помощью алгоритма унификации. В результате унификации формируются подстановки. Унификация запускается, если резольвента не пуста.

Пролог берет начинает последовательно сверху-вниз сравнивать вопрос с фактами и правилами базы знаний. Там, где обнаруживается предикат с таким же идентификатором, как и у вопроса и с таким же количеством аргументов, происходит сопоставление.

8. В каком случае запускается механизм отката?

В ситуации, когда решение не найдено, и из данного состояния невозможен переход в новое состояние, автоматически включается бэктрэкинг. Происходит возврат к моменту, где еще можно сделать другой альтернативный выбор, то есть к предыдущему состоянию резольвенты. Бэктрэкинг возможен только при наличии альтернативных путей унификации цели.

## Код программы

domains

address = address(string, string, integer, integer). % city, street, house, flat

abonent = abonent(symbol, string, address). % surname, phone, address

investor = investor(symbol, string, string, integer). % surname, bank, deposit number, amount

property = car(string, string, integer); % model, color, cost

water\_tr(symbol, integer); %type, price

house(string, address, integer); %name, address, price

area(string, integer, string, integer). %name, area, cadastral number, cost

predicates

owns(symbol, property). %surname, property

names\_and\_cost(symbol, string, integer).

clauses

owns(ivanov, car("Land Rover", "Dark blue", 1000000)).

owns(ivanov, water\_tr("Yacht", 2000000)).

owns(ivanov, house("Flat", address("Moscow", "Lenina", 1, 3), 300000)).

owns(ivanov, area("Dacha area", 300, "12:34:567890:123", 50000)).

owns(petrov, car("Ford", "Black", 400000)).

names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, car(NAME, \_, PRICE)).

names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, water\_tr(NAME, PRICE)).

names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, house(NAME, \_, PRICE)).

names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, area(NAME, \_, \_, PRICE)).

goal

%names\_and\_cost(petrov, Name, Price).

names\_and\_cost(petrov, PROPERTY, \_).

## Примеры работы программы

1. Поиск названий всех объектов собственности заданного субъекта.

Вопрос: names\_and\_cost(ivanov, PROPERTY, \_).

Результат (рис. 1):

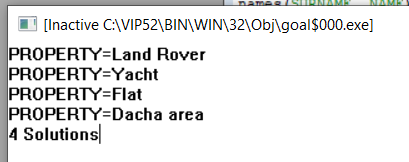


Рисунок 1. Задание 1 (часть 1).

Вопрос: names\_and\_cost(petrov, PROPERTY, \_).

Результат (рис. 2):

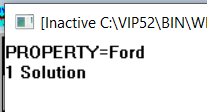


Рисунок 2. Задание 1 (часть 2).

Вопрос: names\_and\_cost(sidorov, PROPERTY, \_).

Результат (рис. 3):

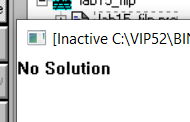


Рисунок 3. Задание 1 (часть 3).

1. Поиск названий и стоимости всех объектов собственности заданного субъекта.

Вопрос: names\_and\_cost(ivanov, Name, Price).

Результат (рис. 4):

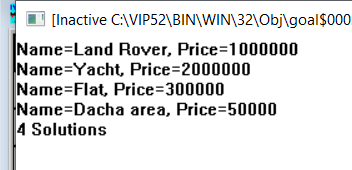


Рисунок 4. Задание 2 (часть 1).

Вопрос: names\_and\_cost(petrov, Name, Price).

Результат (рис. 5):

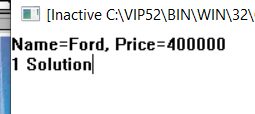


Рисунок 5. Задание 2 (часть 2).

Вопрос: names\_and\_cost(sidorov, Name, Price).

Результат (рис. 6):

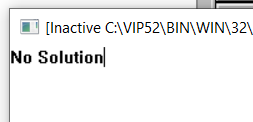


Рисунок 6. Задание 2 (часть 3).

|  |  |  |
| --- | --- | --- |
| № шага | Сравниваемые термы; результат; подстановка, если есть | Дальнейшие действия: прямой ход или откат (к чему приводит?) |
| 1 | Попытка унификации:  T1 = names\_and\_cost(petrov, Name, Price).  T2 = owns(ivanov, car("Land Rover", "Dark blue", 1000000)).  Результат: неудача. Термы не унифицируемы: разные функторы. | Откат и переход к следующему предложению. |
| 2 | Попытка унификации:  T1 = names\_and\_cost(petrov, Name, Price).  T2 = owns(ivanov, water\_tr("Yacht", 2000000)).  Результат: неудача. Термы не унифицируемы: разные функторы. | Откат и переход к следующему предложению. |
| 3 | Попытка унификации:  T1 = names\_and\_cost(petrov, Name, Price).  T2 = owns(ivanov, house("Flat", address("Moscow", "Lenina", 1, 3), 300000)).  Результат: неудача. Термы не унифицируемы: разные функторы. | Откат и переход к следующему предложению. |
| 4 | Попытка унификации:  T1 = names\_and\_cost(petrov, Name, Price).  T2 = owns(ivanov, area("Dacha area", 300, "12:34:567890:123", 50000)).  Результат: неудача. Термы не унифицируемы: разные функторы. | Откат и переход к следующему предложению. |
| 5 | Попытка унификации:  T1 = names\_and\_cost(petrov, Name, Price).  T2 = owns(petrov, car("Ford", "Black", 400000)).  Результат: неудача. Термы не унифицируемы: разные функторы. | Откат и переход к следующему предложению. |
| 6 | Попытка унификации:  T1 = names\_and\_cost(petrov, Name, Price).  T2 = names\_and\_cost(SURNAME, NAME, PRICE)  Результат: успех.  Подстановка: {SURNAME=petrov, NAME=Name, PRICE=price} | Прямой ход |
| 7 | Попытка унификации:  T1 = owns(petrov, car(Name, \_, Price)).  T2 = owns(ivanov, car("Land Rover", "Dark blue", 1000000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 8 | Попытка унификации:  T1 = owns(petrov, car(Name, \_, Price)).  T2 = owns(ivanov, water\_tr("Yacht", 2000000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 9 | Попытка унификации:  T1 = owns(petrov, car(Name, \_, Price)).  T2 = owns(ivanov, house("Flat", address("Moscow", "Lenina", 1, 3), 300000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 10 | Попытка унификации:  T1 = owns(petrov, car(Name, \_, Price)).  T2 = owns(ivanov, area("Dacha area", 300, "12:34:567890:123", 50000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 11 | Попытка унификации:  T1 = owns(petrov, car(Name, \_, Price)).  T2 = owns(petrov, car("Ford", "Black", 400000)).  Результат: успех.  Подстановка: {petrov=petrov, Name=”Ford”, \_\_ =”Black”, Price=400000} | Найдена часть ответа, переход к следующему предложению |
| 12 | Попытка унификации:  T1 = owns(petrov, car(Name, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, car(NAME, \_, PRICE)).  Результат: неудача, термы не унифицируемы: разные функторы. | Откат, переход к следующему предложению |
| 13 | Попытка унификации:  T1 = owns(petrov, car(Name, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, water\_tr(NAME, PRICE)).  Результат: неудача, термы не унифицируемы: разные функторы. | Откат, переход к следующему предложению |
| 14 | Попытка унификации:  T1 = owns(petrov, car(Name, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, house(NAME, \_, PRICE)).  Результат: неудача, термы не унифицируемы: разные функторы. | Откат, переход к следующему предложению |
| 15 | Попытка унификации:  T1 = owns(petrov, car(Name, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, area(NAME, \_, \_, PRICE)).  Результат: неудача, термы не унифицируемы: разные функторы. | Откат, переход к следующему предложению. Предложений больше нет. Возврат к шагу 10. |
| 16 | Попытка унификации:  T1 = names\_and\_cost(petrov, Name, Price).  T2 = names\_and\_cost(SURNAME, NAME, PRICE)  Результат: успех.  Подстановка: {SURNAME=petrov, NAME=Name, PRICE=price} | Прямой ход |
| 17 | Попытка унификации:  T1 = owns(petrov, water\_tr(NAME, PRICE)).  T2 = owns(ivanov, car("Land Rover", "Dark blue", 1000000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 18 | Попытка унификации:  T1 = owns(petrov, water\_tr(Name, Price)).  T2 = owns(ivanov, water\_tr("Yacht", 2000000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 19 | Попытка унификации:  T1 = owns(petrov, water\_tr(Name, Price)).  T2 = owns(ivanov, house("Flat", address("Moscow", "Lenina", 1, 3), 300000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 20 | Попытка унификации:  T1 = owns(petrov, water\_tr(Name, Price)).  T2 = owns(ivanov, area("Dacha area", 300, "12:34:567890:123", 50000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 21 | Попытка унификации:  T1 = owns(petrov, water\_tr(Name, Price)).  T2 = owns(petrov, car("Ford", "Black", 400000)).  Результат: неудача. Функторы water\_tr и car не совпадают. | Откат, переход к следующему предложению |
| 22 | Попытка унификации:  T1 = owns(petrov, water\_tr(Name, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, car(NAME, \_, PRICE)).  Результат: неудача. Термы не унифицируемы. Разные функторы. | Откат, переход к следующему предложению |
| 23 | Попытка унификации:  T1 = owns(petrov, water\_tr(Name, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, water\_tr(NAME, PRICE)).  Результат: неудача. Термы не унифицируемы. Разные функторы. | Откат, переход к следующему предложению |
| 24 | Попытка унификации:  T1 = owns(petrov, water\_tr(Name, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, house(NAME, \_, PRICE)).  Результат: неудача. Термы не унифицируемы. Разные функторы. | Откат, переход к следующему предложению |
| 25 | Попытка унификации:  T1 = owns(petrov, water\_tr(Name, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, area(NAME, \_, \_, PRICE)).  Результат: неудача. Термы не унифицируемы. Разные функторы. | Откат, переход к следующему предложению. Предложений больше нет. Возврат к шагу 24 |
| 26 | Попытка унификации:  T1 = names\_and\_cost(petrov, Name, Price).  T2 = names\_and\_cost(SURNAME, NAME, PRICE)  Результат: успех.  Подстановка: {SURNAME=petrov, NAME=Name, PRICE=price} | Прямой ход |
| 27 | Попытка унификации:  T1 = owns(petrov, house(Name, \_, Price)).  T2 = owns(ivanov, car("Land Rover", "Dark blue", 1000000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 28 | Попытка унификации:  T1 = owns(petrov, house(Name, \_, Price)).  T2 = owns(ivanov, water\_tr("Yacht", 2000000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 29 | Попытка унификации:  T1 = owns(petrov, house(Name, \_, Price)).  T2 = owns(ivanov, house("Flat", address("Moscow", "Lenina", 1, 3), 300000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 30 | Попытка унификации:  T1 = owns(petrov, house(Name, \_, Price)).  T2 = owns(ivanov, area("Dacha area", 300, "12:34:567890:123", 50000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 31 | Попытка унификации:  T1 = owns(petrov, house(Name, \_, Price)).  T2 = owns(petrov, car("Ford", "Black", 400000)).  Результат: неудача. Функторы house и car не совпадают. | Откат, переход к следующему предложению |
| 32 | Попытка унификации:  T1 = owns(petrov, house(Name, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, car(NAME, \_, PRICE)).  Результат: неудача. Разные функторы. | Откат, переход к следующему предложению |
| 33 | Попытка унификации:  T1 = owns(petrov, house(Name, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, water\_tr(NAME, PRICE)).  Результат: неудача. Разные функторы. | Откат, переход к следующему предложению |
| 34 | Попытка унификации:  T1 = owns(petrov, house(Name, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, house(NAME, \_, PRICE)).  Результат: неудача. Разные функторы. | Откат, переход к следующему предложению |
| 35 | Попытка унификации:  T1 = owns(petrov, house(Name, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, area(NAME, \_, \_, PRICE)).  Результат: неудача. Разные функторы. | Откат, переход к следующему предложению. Предложений больше нет. Возврат к шагу 38 |
| 36 | Попытка унификации:  T1 = names\_and\_cost(petrov, Name, Price).  T2 = names\_and\_cost(SURNAME, NAME, PRICE)  Результат: успех.  Подстановка: {SURNAME=petrov, NAME=Name, PRICE=price} | Прямой ход |
| 37 | Попытка унификации:  T1 = owns(petrov, area(Name, \_, \_, Price)).  T2 = owns(ivanov, car("Land Rover", "Dark blue", 1000000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 38 | Попытка унификации:  T1 = owns(petrov, area(Name, \_, \_, Price)).  T2 = owns(ivanov, water\_tr("Yacht", 2000000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 39 | Попытка унификации:  T1 = owns(petrov, area(Name, \_, \_, Price)).  T2 = owns(ivanov, house("Flat", address("Moscow", "Lenina", 1, 3), 300000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 40 | Попытка унификации:  T1 = owns(petrov, area(Name, \_, \_, Price)).  T2 = owns(ivanov, area("Dacha area", 300, "12:34:567890:123", 50000)).  Результат: неудача, petrov != ivanov | Откат, переход к следующему предложению |
| 41 | Попытка унификации:  T1 = owns(petrov, area(Name, \_, \_, Price)).  T2 = owns(petrov, car("Ford", "Black", 400000)).  Результат: неудача. Функторы area и car не совпадают. | Откат, переход к следующему предложению |
| 42 | Попытка унификации:  T1 = owns(petrov, area(Name, \_, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, car(NAME, \_, PRICE)).  Результат: неудача. Разные функторы. | Откат, переход к следующему предложению |
| 43 | Попытка унификации:  T1 = owns(petrov, area(Name, \_, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, water\_tr(NAME, PRICE)).  Результат: неудача. Разные функторы. | Откат, переход к следующему предложению |
| 44 | Попытка унификации:  T1 = owns(petrov, area(Name, \_, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, house(NAME, \_, PRICE)).  Результат: неудача. Разные функторы. | Откат, переход к следующему предложению |
| 45 | Попытка унификации:  T1 = owns(petrov, area(Name, \_, \_, Price)).  T2 = names\_and\_cost(SURNAME, NAME, PRICE) :- owns(SURNAME, area(NAME, \_, \_, PRICE)).  Результат: неудача. Разные функторы. | Откат, переход к следующему предложению. Предложений больше нет. Возврат к шагу 52 |
| 46 | Предложений больше нет | Ответ найден. Завершение работы. Вывод ответа на экран. |