



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 № 5

Дисциплина: Функциональное и логическое программирование

Студент: Платонова Ольга

Группа: ИУ7-65Б

Преподаватели: Толпинская Н. Б.
Строганов Ю. В.

Москва, 2021 г.

Задание

Создать базу знаний «Собственники», дополнив базу знаний, хранящую знания:

- «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:

- Строение, стоимость и другие его характеристики;
- Участок, стоимость и другие его характеристики;
- Водный_транспорт, стоимость и другие его характеристики.

Обеспечить возможность поиска:

1. Названий всех объектов собственности заданного субъекта,
2. Названий и стоимости всех объектов собственности заданного субъекта.

Листинг

```
domains
    name, phone, city, street = string.
    model, color = string.
    bank = string.

    house, flat, nfloors, nha = integer.
    cost, year = integer.
    account, amount, cell = integer.

    sea = string.

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

    property = car(model, color, cost, year);
                building(cost, year, nfloors);
                ground(cost, nha);
                boat(cost, year, sea).

predicates
    phoneRecord(name, phone, address).
    bankRecord(name, bank, account, amount, cell).
    ownRecord(name, city, property).

    getNameOwn(name, string).
    getNameCostOwn(name, string, integer).

clauses
    phoneRecord("Ivanov", "+7910284", address("Moscow", "Berezovaya", 4, 23)).
    phoneRecord("Petrov", "+7902873", address("Riga", "Grushnevaya", 36, 109)).
    phoneRecord("Sidorov", "+7902873", address("Kazan", "Dryzby", 84, 91)).
    phoneRecord("Fedotov", "+7918276", address("Piter", "Smolenskaya", 62, 41)).

    bankRecord("Petrov", "Sber", 8019, 190276, 346).
    bankRecord("Sidorov", "Tinkoff", 0923, 19872, 901).
    bankRecord("Fedotov", "Raif", 8801, 80238, 19).
    bankRecord("Fedotov", "Russ", 1874, 901238, 1984).

    ownRecord("Ivanov", "Moscow", building(8000000, 1990, 1)).
    ownRecord("Ivanov", "Moscow", car("Volvo", "blue", 891000, 2009)).
    ownRecord("Petrov", "Riga", ground(800000, 45)).
    ownRecord("Petrov", "Riga", car("BMW", "white", 1000000, 2010)).
    ownRecord("Sidorov", "Kazan", car("BMW", "white", 1200000, 2020)).
    ownRecord("Fedotov", "Piter", car("Lambargini", "red", 5901000, 2021)).
    ownRecord("Fedotov", "Piter", building(38000000, 2020, 5)).
    ownRecord("Fedotov", "Piter", boat(12000000, 2019, "Red")).


    getNameOwn(Name, NProp) :- ownRecord(Name, _, car(_, _, _), NProp = "Car".
    getNameOwn(Name, NProp) :- ownRecord(Name, _, building(_, _, _), NProp = "Building".
    getNameOwn(Name, NProp) :- ownRecord(Name, _, ground(_, _), NProp = "Ground".
    getNameOwn(Name, NProp) :- ownRecord(Name, _, boat(_, _, _), NProp = "Boat".

    getNameCostOwn(Name, NProp, CProp) :- ownRecord(Name, _, car(_, _, CProp, _), NProp = "Car".
    getNameCostOwn(Name, NProp, CProp) :- ownRecord(Name, _, building(CProp, _, _), NProp = "Building".
    getNameCostOwn(Name, NProp, CProp) :- ownRecord(Name, _, ground(CProp, _), NProp = "Ground".
    getNameCostOwn(Name, NProp, CProp) :- ownRecord(Name, _, boat(CProp, _, _), NProp = "Boat".
```

1. Названия всех объектов собственности заданного субъекта.

goal

```
getNameOwn("Fedotov", NameProperty).
```

 [Inactive C:\VIP52\DOC\EXAMPLES\TestGoal\Obj\goal\$000.exe]

NameProperty=Car
NameProperty=Building
NameProperty=Boat
3 Solutions

2. Названия и стоимости всех объектов собственности заданного субъекта.

```
goal
    %getNameOwn("Fedotov", NameProperty).
    getNameCostOwn("Fedotov", NameProperty, CostProperty).
```

[Inactive C:\VIP52\DOC\EXAMPLES\TestGoal\Obj\goal\$000.exe]

NameProperty=Car, CostProperty=5901000
NameProperty=Building, CostProperty=38000000
NameProperty=Boat, CostProperty=12000000
3 Solutions

`getNameCostOwn("Fedotov", NameProperty, CostProperty).`

| № шага | Сравниваемые термы; результат; подстановка | Дальнейшие действия |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | <code>getNameCostOwn("Fedotov", NameProperty, CostProperty).</code> <code>phoneRecord(...)</code> Неудача. Не унифицируемые. | Переход к следующему заголовку БЗ. |
| | ... | |
| 2 | <code>getNameCostOwn("Fedotov", NameProperty, CostProperty).</code> <code>getNameCostOwn(Name, NProp, CProp).</code> Успех. Унифицируемые. Подстановка: {Name = "Fedotov", NProp = NameProperty, CProp = CProperty}. | Удаляется из стека: <code>getNameCostOwn("Fedotov", NameProperty, CostProperty).</code> Новая резолювента: <code>ownRecord("Fedotov", _, car(_, _, CProp, _)),</code> <code>NProp = "Car".</code> |
| 3 | <code>ownRecord("Fedotov", _, car(_, _, CProp, _)).</code> <code>phoneRecord(...).</code> Неудача. Не унифицируемые. | Переход к следующему заголовку БЗ. |
| | ... | |

| | | |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4 | <p>ownRecord("Fedotov", _, car(_, _, CProp, _)).</p> <p>ownRecord("Fedotov", "Piter", car("Lambargini", "red", 5901000, 2021)).</p> <p>Успех. Унифицируемые.</p> <p>Подстановка: {"Fedotov" = "Fedotov", "Piter" = _, "Lambargini" = _, "red" = _, 5901000 = CProp, 2021 = _}.</p> | <p>Удаляется из стека:</p> <p>ownRecord("Fedotov", _, car(_, _, CProp, _)).</p> <p>Новая резолювента:</p> <p>NProp = "Car".</p> |
| 5 | | <p>Удаляется из стека:</p> <p>NProp = "Car".</p> <p>Выводится "Fedotov", "Car", 5901000</p> <p>Новая резолювента:</p> <p>ownRecord("Fedotov", _, car(_, _, CProp, _)),</p> <p>NProp = "Car".</p> |
| 6 | <p>ownRecord("Fedotov", _, car(_, _, CProp, _)).</p> <p>ownRecord("Fedotov", "Piter", building(38000000, 2020, 5)).</p> <p>Неудача. Не унифицируемые.</p> | <p>Переход к следующему заголовку БЗ.</p> |
| | ... | |
| 7 | <p>ownRecord("Fedotov", _, car(_, _, CProp, _)).</p> <p>getNameCostOwn().</p> <p>Неудача. Не унифицируемые.</p> | <p>Все факты пройдены.</p> <p>Удаляется из стека:</p> <p>ownRecord("Fedotov", _, car(_, _, CProp, _)),</p> <p>NProp = "Car".</p> <p>Новая резолювента:</p> <p>getNameCostOwn("Fedotov", NameProperty, CostProperty).</p> |
| 8 | <p>getNameCostOwn("Fedotov", NameProperty, CostProperty).</p> | <p>Удаляется из стека:</p> |

| | | |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | <p>getNameCostOwn(Name, NProp, CProp).</p> <p>Успех. Унифицируемые.</p> <p>Подстановка: {Name = "Fedotov", NProp = NameProperty, CProp = CProperty}.</p> | <p>getNameCostOwn("Fedotov", NameProperty, CostProperty).</p> <p>Новая резолювента:</p> <p>ownRecord("Fedotov", _, building(Cprop, _, _)), NProp = "Building".</p> |
| 9 | <p>ownRecord("Fedotov", _, building(CProp, _, _)). phoneRecord(...)</p> <p>Неудача. Не унифицируемые.</p> | <p>Переход к следующему заголовку БЗ.</p> |
| | ... | |
| 10 | <p>ownRecord("Fedotov", _, building(CProp, _, _)). ownRecord("Fedotov", "Piter", building(38000000, 2020, 5)).</p> <p>Успех. Унифицируемые.</p> <p>Подстановка: {"Fedotov" = "Fedotov", "Piter" = _, 38000000 = CProp, 2020 = _, 5 = _}.</p> | <p>Удаляется из стека:</p> <p>ownRecord("Fedotov", _, building(CProp, _, _)).</p> <p>Новая резолювента:</p> <p>NProp = "Building".</p> |
| 11 | | <p>Удаляется из стека:</p> <p>NProp = "Building".</p> <p>Выводится "Fedotov", "Building", 38000000</p> <p>Новая резолювента:</p> <p>ownRecord("Fedotov", _, building(CProp, _, _)), NProp = "Building".</p> |
| 12 | <p>ownRecord("Fedotov", _, building(CProp, _, _)). ownRecord("Fedotov", "Piter", boat(12000000, 2019, "Red")).</p> <p>Неудача. Не унифицируемые.</p> | <p>Переход к следующему заголовку БЗ.</p> |
| 13 | ... | |

| | | |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 14 | <p>ownRecord("Fedotov", _, building(CProp, _ _)). getNameCostOwn().</p> <p>Неудача. Не унифицируемые.</p> | <p>Все факты пройдены.</p> <p>Удаляется из стека: ownRecord("Fedotov", _, building(CProp, _ _)), NProp = "Building".</p> <p>Новая резолювента: getNameCostOwn("Fedotov", NameProperty, CostProperty).</p> |
| 15 | <p>getNameCostOwn("Fedotov", NameProperty, CostProperty). getNameCostOwn(Name, NProp, CProp).</p> <p>Успех. Унифицируемые.</p> <p>Подстановка: {Name = "Fedotov", NProp = NameProperty, CProp = CProperty}.</p> | <p>Удаляется из стека: getNameCostOwn("Fedotov", NameProperty, CostProperty).</p> <p>Новая резолювента: ownRecord("Fedotov", _, ground(CProp, _)), NProp = "Ground".</p> |
| 16 | <p>ownRecord(Name, _, ground(CProp, _)). phoneRecord(...)</p> <p>Неудача. Не унифицируемые.</p> | <p>Переход к следующему заголовку БЗ.</p> |
| | ... | |
| 17 | <p>ownRecord(Name, _, ground(CProp, _)). getNameCostOwn().</p> <p>Неудача. Не унифицируемые.</p> | <p>Все факты пройдены.</p> <p>Удаляется из стека: ownRecord("Fedotov", _, ground(CProp, _)), NProp = "Ground".</p> <p>Новая резолювента: getNameCostOwn("Fedotov", NameProperty, CostProperty).</p> |
| 18 | <p>getNameCostOwn("Fedotov", NameProperty, CostProperty). getNameCostOwn(Name, NProp, CProp).</p> | <p>Удаляется из стека: getNameCostOwn("Fedotov", NameProperty, CostProperty).</p> |

| | | |
|----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | <p>Успех. Унифицируемые.</p> <p>Подстановка: {Name = "Fedotov", NProp = NameProperty, CProp = CProperty}.</p> | <p>Новая резолювента:</p> <p>ownRecord("Fedotov", _, boat(CProp, _, _)), NProp = "Boat".</p> |
| 19 | <p>ownRecord("Fedotov", _, boat(CProp, _, _)). phoneRecord(...)</p> <p>Неудача. Не унифицируемые.</p> | <p>Переход к следующему заголовку БЗ.</p> |
| | ... | |
| 20 | <p>ownRecord("Fedotov", _, boat(CProp, _, _)). ownRecord("Fedotov", "Piter", boat(12000000, 2019, "Red")).</p> <p>Успех. Унифицируемые.</p> <p>Подстановка: {"Fedotov" = "Fedotov", "Piter" = _, 12000000= CProp, 2019 = _, "Red" = _}.</p> | <p>Удаляется из стека:</p> <p>ownRecord("Fedotov", _, boat(CProp, _, _)).</p> <p>Новая резолювента:</p> <p>NProp = "Boat".</p> |
| 21 | | <p>Удаляется из стека:</p> <p>NProp = "Boat".</p> <p>Выводится "Fedotov", "Boat", 12000000</p> <p>Новая резолювента:</p> <p>ownRecord("Fedotov", _, boat(CProp, _, _)), NProp = "Boat".</p> |
| 22 | <p>ownRecord("Fedotov", _, boat(CProp, _, _)). getNameOwn().</p> <p>Неудача. Не унифицируемые.</p> | <p>Переход к следующему заголовку БЗ.</p> |
| 23 | ... | |
| 24 | <p>ownRecord("Fedotov", _, boat(CProp, _, _)). getNameCostOwn().</p> | <p>Все факты пройдены.</p> <p>Удаляется из стека:</p> |

| | | |
|----|-----------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| | Неудача. Не унифицируемые. | ownRecord("Fedotov", _, boat(CProp, _, _)), NProp = "Boat". Новая резолювента: getNameCostOwn("Fedotov", NameProperty, CostProperty). |
| 25 | getNameCostOwn("Fedotov", NameProperty, CostProperty). | Все факты пройдены. Удаляется из стека: getNameCostOwn("Fedotov", NameProperty, CostProperty). Резолювента пуста. |

Вывод

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

Знание содержится в заголовке предложений базы знаний. А предложения - в разделе CLAUSES. Знание о том, что между аргументами терма (тела правила) существует отношение.

2. Что содержит тело правила?

Тело содержит условие истинности 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?

Переменная – способ обобщить. Предназначена для передачи значений «во времени и в пространстве». Чем больше переменных, тем более общий результат.

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

Переменные в факты и правила входят с квантором всеобщности («любой» элемент из множества). Именованные переменные уникальны в рамках предложения, а анонимная – любая уникальная.

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

DOMAINS – раздел описания доменов. В разделе объявляются любые нестандартные домены в формате: <имя домена> = <определение домена>. Используется с целью описания имен и семантики доменов, когда природа или структура объектов не может быть определена с помощью стандартных доменов.

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

PREDICATES – раздел описания предикатов. Раздел содержит описание структуры предиката. Используется с целью указания доменов, когда природа или структура объектов важна во время работы системы.

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

Вопроса и заголовка первого предложения базы знаний.

Алгоритм унификации необходим для того, чтобы подобрать знание, чтобы ответить на поставленный вопрос. Результатом работы алгоритма является значение переменной «неудача». Если неудача = 1, то унификация невозможна;

если неудача = 0, то побочным действием работы алгоритма является содержимое результирующей ячейки – результирующая подстановка.

8. *В каком случае запускается механизм отката?*

В случае, когда унификация на текущем шаге завершается тупиковой ситуацией, или был получен ответ «да».