



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»

ЛАБОРАТОРНАЯ РАБОТА №15

Дисциплина

Функциональное и

логическое программирование

Тема

Структура программы на Prolog

и ее реализация

Студент

Набиев Ф.М.

Группа

ИУ7-63Б

Преподаватель

Толпинская Н.Б., Строганов Ю.В.

Москва, 2020 г.

ВВЕДЕНИЕ

Цель работы — изучить структуру, особенности и принципы оформления программы, и способ выполнения программы на Prolog.

Задачи работы: приобрести навыки декларативного описания предметной области с использованием фактов, правил и некоторых специальных разделов программы. Изучить порядок использования фактов и правил в программе на Prolog, принципы и особенности сопоставления и отождествления термов, на основе механизма унификации.

1 Теоретическая часть

В этом разделе приведены ответы на контрольные вопросы.

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

Знание формируется в заголовке правила (правила описываются в разделе **clauses**), может быть описано набором фактов и правил, формирующих процедуру.

Знание о предметной области.

1.2 Что содержит тело правила?

Тело правила содержит условия истинности знания, содержащегося в заголовке этого правила. Тело состоит из термов.

1.3 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?

Использование переменных дает возможность передачи знаний о предметной области. Например, связанная с каким-то значением переменная, в рамках одного предложения, может быть использована в последующих подцелях тела правила

Чем больше переменных используется в определении терма, тем более общим будет являться терм. Например, `car(petrov, ford, pink, Price)` является примером терма `car(Lastname, ford, pink, Price)`, т.е. `car(Lastname, ford, pink, Price)` является более общим.

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

На все переменные в имени предиката наложен квантор всеобщности; на переменные в теле предиката, которые отсутствуют в имени, наложен квантор существования.

Именованная переменная уникальна в рамках предложения, в котором она используется. Любая анонимная переменная является уникальной.

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

Раздел описания доменов является аналогом раздела описания типов в императивных языках программирования. Домены должны быть определены до их использования. Например: lastname=symbol.

Из доменов можно конструировать составные или структурные домены (структуры). Структура описывается следующим образом:

<имя структуры>=<имя функтора>(<имя домена первой компоненты>, . . . , <имя домена последней компоненты>).

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

В разделе, озаглавленном зарезервированным словом **predicates**, содержатся описания определяемых пользователем предикатов. В императивных языках программирования подобными разделами являются разделы описания заголовков процедур и функций. Предикаты должны быть определены до их использования. Описание n-местного предиката имеет следующий вид:

<имя предиката>(<имя домена первого аргумента>, . . . , <имя

домена n -го аргумента $>$).

Домены аргументов должны быть либо стандартными, либо объявленными в разделе описания доменов. Один предикат может иметь несколько описаний

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

На первом шаге работы системы унифицируется вопрос с первым термом в разделе **clauses**. Унификация — основной вычислительный шаг работы программы, предназначенный для формализации процесса логического вывода. Унификация – попытка сопоставить два терма. Результатом алгоритма унификации является «успех» или «неудача». В случае удачи в результате также является наиболее общий унификатор двух унифицируемых термов.

1.8 В каком случае запускается механизм отката?

Откат дает возможность получить много решений в одном вопросе к программе. Во всех точках программы, где существуют альтернативы, в стек заносятся точки возврата.

Если впоследствии окажется, что выбранный вариант не приводит к успеху, то осуществляется откат к последней из имеющихся в стеке точек программы, где был выбран один из альтернативных вариантов.

Выбирается очередной вариант, программа продолжает свою работу. Если все варианты в точке уже были использованы, то регистрируется неудачное завершение и осуществляется переход на предыдущую точку возврата, если такая есть. При откате все связанные переменные, которые были означены после этой точки, опять освобождаются.

2 Практическая часть

В этом разделе приведено описание задания и его решения.

2.1 Условие

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 пункта и одной фамилии составить таблицу, отражающую конкретный порядок работы системы, с объяснениями порядка работы и особенностей использования доменов.

2.2 База знаний

Листинг 2.1 – База знаний

```
1 domains
2   lastname , city , street , carbrand , color , bankname , shipname = symbol .
3   phonenum , housenum , apartnum , price , account , deposit = integer .
4   area=real .
5
6   address = address(city , street , housenum , apartnum) .
7   own = car(price , carbrand , color) ;
8       building(price , address) ;
9       landlot(price , area) ;
10      ship(price , shipname) .
11 predicates
12   phonebook(lastname , phonenum , address) .
13   depositor(lastname , bankname , account , deposit) .
14   owner(lastname , own) .
15
16   owns(lastname , symbol , price) .
17 clauses
18   phonebook( petrov , 74400297 ,
19             address( moscow , lenina , 4 , 2 ) ) .
20   phonebook( igorev , 77270935 ,
21             address( moscow , marksa , 3 , 5 ) ) .
22   phonebook( ivanov , 79345669 ,
23             address( moscow , pushkinskaya , 11 , 1 ) ) .
24   phonebook( stasov , 74024456 ,
25             address( spb , marksa , 4 , 4 ) ) .
26   phonebook( igorev , 73243243 ,
27             address( volgograd , lenina , 9 , 9 ) ) .
28
29   depositor( petrov , agricole , 5 , 52150322 ) .
```

```

30 depositor(igorev , sberbank , 8 , 421342352).
31 depositor(ivanov , sberbank , 6 , 442423123).
32 depositor(igorev , paribas , 3 , 41424214).
33
34 owner(petrov , car      (24000 ,      ford ,      pink)).
35 owner(stasov , car      (55000 ,      tesla ,      purple)).
36 owner(ivanov , building(429999 ,      address(ekb , lenina , 2 , 5))).
37 owner(petrov , landlot (23449999 , 43000)).
38 owner(igorev , ship     (86000 ,      chehov)).
39
40 owns(Lastname , Category , Price) :-
41     owner(Lastname , car(Price , \_ , \_)) , Category = car .
42 owns(Lastname , Category , Price) :-
43     owner(Lastname , building(Price , \_)) , Category = building .
44 owns(Lastname , Category , Price) :-
45     owner(Lastname , landlot(Price , \_)) , Category = landlot .
46 owns(Lastname , Category , Price) :-
47     owner(Lastname , ship(Price , \_)) , Category = ship .

```

2.3 Цели

Рассмотрим примеры трёх целей, текст и решение которых приведены в листингах 2.2, 2.3.

Листинг 2.2 – Пример №1

```

1 goal
2   owns(petrov , Category , _).
3
4 % Category=car
5 % Category=landlot
6 % 2 Solutions

```

Листинг 2.3 – Пример №2

```

1 goal
2   owns(stasov , Category , Price).
3
4 % Category=car , Price=55000
5 % 1 Solution

```


Таблица 2.1 – Поиск решения в примере №1

№ шага	Сравниваемые термы; результат; подстановка (если есть)	Дальнейшие действия: прямой ход или откат
1	<p>Сравнение: owns(stasov, Category, Price) и owns(Lastname, Category, Price)</p> <p>Результат: Унификация выполнена. Lastname=stasov, Category=Category, Price=Price</p>	<p>В стек откладывается терм owns(stasov, Category, Price).</p> <p>Прямой ход.</p> <p>На резольвенту попадают: owner(Lastname, car(Price, _, _)), Category = car.</p>
2	<p>Сравнение: owner(stasov, car(Price, _, _)) и owner(petrov, car (24000, ford, pink))</p> <p>Результат: Унификация не выполнена.</p>	<p>Прямой ход.</p>
3	<p>Сравнение: owner(stasov, car(Price, _, _)) и owner(stasov, car (55000, tesla, purple))</p> <p>Результат: Унификация выполнена. Price=5500</p>	<p>В стек откладывается терм owner(stasov, car(Price, _, _))</p> <p>Прямой ход.</p>
4	<p>Category = car — терм, в котором используется встроенный предикат =, проводящий попытку унификации передаваемых аргументов.</p> <p>Результат: Унификация выполнена. Category=car</p>	<p>Получено первое решение: Category=car, Price=55000.</p> <p>В стек откладывается терм Category = car</p> <p>Прямой ход.</p>
5	<p>Резольвента пуста.</p> <p>Результат: Унификация провалена.</p>	<p>Из стека восстанавливается терм Category = car</p> <p>Откат.</p>

6	Процедура исчерпана. Результат: Унификация провалена.	Из стека восстанавливается терм owner(stasov, car(Price, _, _)) Откат.
7	Сравнение: owner(stasov, car(Price, _, _)) и owner(ivanov, building(429999, address(ekb, lenina, 2, 5))) Результат: Унификация не выполнена.	Прямой ход.
8	Сравнение: owner(stasov, car(Price, _, _)) и owner(petrov, landlot (23449999, 43000)) Результат: Унификация не выполнена.	Прямой ход.
9	Сравнение: owner(stasov, car(Price, _, _)) и owner(igorev, ship (86000, chehov)). Результат: Унификация не выполнена.	Прямой ход.
10	Процедура исчерпана. Результат: Унификация провалена.	Из стека восстанавливается терм owner(stasov, car(Price, _, _)) Откат. Стек покидают термы: owner(Lastname, car(Price, _, _)), Category = car.

11	<p>Сравнение: owns(stasov, Category, Price) и owns(Lastname, Category, Price)</p> <p>Результат: Унификация выполнена. Lastname=stasov, Category=Category, Price=Price</p>	<p>В стек откладывается терм owns(stasov, Category, Price).</p> <p>Прямой ход.</p> <p>На резольвенту попадают: owner(Lastname, building(Price, _)), Category = building.</p>
12	<p>Сравнение: owner(stasov, building(Price, _)) и owner(petrov, car (24000, ford, pink)).</p> <p>Результат: Унификация не выполнена.</p>	<p>Прямой ход.</p>
13	<p>Сравнение: owner(stasov, building(Price, _)) и owner(stasov, car (55000, tesla, purple)).</p> <p>Результат: Унификация не выполнена.</p>	<p>Прямой ход.</p>
14	<p>Сравнение: owner(stasov, building(Price, _)) и owner(ivanov, building(429999, address(ekb, lenina, 2, 5))).</p> <p>Результат: Унификация не выполнена.</p>	<p>Прямой ход.</p>
15	<p>Сравнение: owner(stasov, building(Price, _)) и owner(petrov, landlot (23449999, 43000)).</p> <p>Результат: Унификация не выполнена.</p>	<p>Прямой ход.</p>

16	<p>Сравнение: owner(stasov, building(Price, _)) и owner(igorev, ship (86000, chehov)).</p> <p>Результат: Унификация не выполнена.</p>	<p>Прямой ход.</p>
17	<p>Процедура исчерпана.</p> <p>Результат: Унификация провалена.</p>	<p>Из стека восстанавливается терм owner(stasov, building(Price, _))</p> <p>Откат.</p> <p>Стек покидают термы: owner(stasov, building(Price, _)), Category = building.</p>
18	<p>Сравнение: owns(stasov, Category, Price) и owns(Lastname, Category, Price)</p> <p>Результат: Унификация выполнена. Lastname=stasov, Category=Category, Price=Price</p>	<p>В стек откладывается терм owns(stasov, Category, Price).</p> <p>Прямой ход.</p> <p>На резольвенту попадают: owner(Lastname, landlot(Price, _)), Category = landlot.</p>
19	<p>Сравнение: owner(stasov, landlot(Price, _)) и owner(petrov, car (24000, ford, pink)).</p> <p>Результат: Унификация не выполнена.</p>	<p>Прямой ход.</p>
20	<p>Сравнение: owner(stasov, landlot(Price, _)) и owner(stasov, car (55000, tesla, purple)).</p> <p>Результат: Унификация не выполнена.</p>	<p>Прямой ход.</p>

21	<p>Сравнение: owner(stasov, landlot(Price, _)) и owner(ivanov, building(429999, address(ekb, lenina, 2, 5))).</p> <p>Результат: Унификация не выполнена.</p>	Прямой ход.
22	<p>Сравнение: owner(stasov, landlot(Price, _)) и owner(petrov, landlot (23449999, 43000)).</p> <p>Результат: Унификация не выполнена.</p>	Прямой ход.
23	<p>Сравнение: owner(stasov, landlot(Price, _)) и owner(igorev, ship (86000, chehov)).</p> <p>Результат: Унификация не выполнена.</p>	Прямой ход.
24	<p>Процедура исчерпана.</p> <p>Результат: Унификация провалена.</p>	<p>Из стека восстанавливается терм owns(stasov, Category, Price)</p> <p>Откат.</p> <p>Стек покидают термы: owner(stasov, landlot(Price, _)), Category = landlot.</p>
25	<p>Сравнение: owns(stasov, Category, Price) и owns(Lastname, Category, Price)</p> <p>Результат: Унификация выполнена. Lastname=stasov, Category=Category, Price=Price</p>	<p>В стек откладывается терм owns(stasov, Category, Price).</p> <p>Прямой ход.</p> <p>На резолювенту попадают: owner(Lastname, ship(Price, _)), Category = ship.</p>

26	<p>Сравнение: owner(stasov, ship(Price, _)) и owner(petrov, car (24000, ford, pink)).</p> <p>Результат: Унификация не выполнена.</p>	Прямой ход.
27	<p>Сравнение: owner(stasov, ship(Price, _)) и owner(stasov, car (55000, tesla, purple)).</p> <p>Результат: Унификация не выполнена.</p>	Прямой ход.
28	<p>Сравнение: owner(stasov, ship(Price, _)) и owner(ivanov, building(429999, address(ekb, lenina, 2, 5))).</p> <p>Результат: Унификация не выполнена.</p>	Прямой ход.
29	<p>Сравнение: owner(stasov, ship(Price, _)) и owner(petrov, landlot (23449999, 43000)).</p> <p>Результат: Унификация не выполнена.</p>	Прямой ход.
30	<p>Сравнение: owner(stasov, ship(Price, _)) и owner(igorev, ship (86000, chehov)).</p> <p>Результат: Унификация не выполнена.</p>	Прямой ход.
31	<p>Процедура исчерпана.</p> <p>Результат: Унификация провалена.</p>	<p>Из стека восстанавливается терм owns(stasov, Category, Price)</p> <p>Откат.</p> <p>Стек покидают термы: owner(stasov, ship(Price, _)), Category = ship.</p>

32	Процедура исчерпана. Результат: Унификация провалена.	Стек покидает терм owns(stasov, Category, Price) Поиск решений завершен. Найдено одно решение.
----	--	---