



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 № 12**

Тема Структура программы на Prolog

Студент Сушина А.Д.

Группа ИУ7-616

Оценка (баллы) \_\_\_\_\_

Преподаватель Толпинская Н.Б.

Москва.  
2020 г

**Цель работы** – познакомиться со структурой, принципами оформления и логикой выполнения программы на Prolog

**Задачи работы:** приобрести навыки декларативного описания предметной области с использованием фактов и правил. Изучить способы использования фактов и правил в программе на Prolog, принципы и правила сопоставления и отождествления, принцип унификации.

### Ход работы

#### Задание:

Составить программу – базу знаний, с помощью которой можно определить, например, множество студентов, обучающихся в одном ВУЗе. Студент может одновременно обучаться в нескольких ВУЗах. Привести примеры возможных вариантов вопросов и варианты ответов (не менее 3-х). Описать порядок формирования вариантов ответа.

Исходную базу знаний сформировать с помощью только фактов.

\*Исходную базу знаний сформировать, используя правила.

\*Разработать свою базу знаний (содержание произвольно).

#### Листинг 1. База знаний студентов

```
1. domains
2.   Name,Surname, Group, University = symbol.
3. predicates
4.   student(Name, Surname, Group, University).
5.   getStudentsByUni(Name, Surname,University)
6. clauses
7.   student("Vanya","Pupkin" ,"iu7-61b", "BMSTU").
8.   student("Sergey","Romanov" ,"iu7-61b", "BMSTU").
9.   student("Kostya","Naumov" ,"iu7-61b", "BMSTU").
10.  student("Kolya", "Lovkin","iu7-63b", "BMSTU").
11.  student("Petya", "Arimov","iu7-63b", "BMSTU").
12.  student("Vanya","Gornov" ,"iu7-62b", "MSU").
13.  student("Olga","Buzova" ,"iu7-23b", "MSU").
14.  student("Vanya","Pupkin" ,"iu7-33b", "MSU").
15.  getStudentsByUni(N, L, U) :- student(N, L,_, U).
16.
17. goal
18.  %student(Name,Surname,Group, "BMSTU").
19.  %student(Name,Surname,"iu7-61b", "BMSTU").
20.  student("Vanya","Pupkin", Group, University).
21.  %getStudentsByUni(Name,Surname,"MSU").
```

В базе знаний содержится информация о студентах: их имя, фамилия, группа и университет.

Name используется для объявления имени

Surname используется для объявления фамилии

Group используется для объявления группы

University используется для объявления университета

```
domains
    Name, Surname, Group, University = symbol.

predicates
    student(Name, Surname, Group, University).
    getStudentsByUni(Name, Surname, University)

clauses
    student("Vanya", "Pupkin", "iu7-61b", "BMSTU").
    student("Sergey", "Romanov", "iu7-61b", "BMSTU").
    student("Kostya", "Naumov", "iu7-61b", "BMSTU").
    student("Kolya", "Lovkin", "iu7-63b", "BMSTU").
    student("Petya", "Arimov", "iu7-63b", "BMSTU").
    student("Vanya", "Gornov", "iu7-62b", "MSU").
    student("Olga", "Buzova", "iu7-23b", "MSU").
    student("Vanya", "Pupkin", "iu7-33b", "MSU").
    getStudentsByUni(N, L, U) :- student(N, L, _, U).

goal
    student(Name, Surname, Group, "BMSTU").
    %student(Name, Surname, "iu7-61b", "BMSTU").
    %student("Vanya", "Pupkin", Group, University).
    %getStudentsByUni(Name, Surname, "MSU").
```

```
[Inactive C:\VIP52\BIN\WIN\32\Obj\goal$00...
Name=Vanya, Surname=Pupkin, Group=iu7-61b
Name=Sergey, Surname=Romanov, Group=iu7-61b
Name=Kostya, Surname=Naumov, Group=iu7-61b
Name=Kolya, Surname=Lovkin, Group=iu7-63b
Name=Petya, Surname=Arimov, Group=iu7-63b
5 Solutions
```

Рис 1. Все студенты общающиеся в BMSTU

```
domains
    Name, Surname, Group, University = symbol.

predicates
    student(Name, Surname, Group, University).
    getStudentsByUni(Name, Surname, University)

clauses
    student("Vanya", "Pupkin", "iu7-61b", "BMSTU").
    student("Sergey", "Romanov", "iu7-61b", "BMSTU").
    student("Kostya", "Naumov", "iu7-61b", "BMSTU").
    student("Kolya", "Lovkin", "iu7-63b", "BMSTU").
    student("Petya", "Arimov", "iu7-63b", "BMSTU").
    student("Vanya", "Gornov", "iu7-62b", "MSU").
    student("Olga", "Buzova", "iu7-23b", "MSU").
    student("Vanya", "Pupkin", "iu7-33b", "MSU").
    getStudentsByUni(N, L, U) :- student(N, L, _, U).

goal
    %student(Name, Surname, Group, "BMSTU").
    student(Name, Surname, "iu7-61b", "BMSTU").
    %student("Vanya", "Pupkin", Group, University).
    %getStudentsByUni(Name, Surname, "MSU").
```

```
[Inactive C:\VIP52\BIN\WIN\32\...
Name=Vanya, Surname=Pupkin
Name=Sergey, Surname=Romanov
Name=Kostya, Surname=Naumov
3 Solutions
```

Рис 2. Все студенты одной группы

```
domains
    Name, Surname, Group, University = symbol.

predicates
    student(Name, Surname, Group, University).
    getStudentsByUni(Name, Surname, University)

clauses
    student("Vanya", "Pupkin", "iu7-61b", "BMSTU").
    student("Sergey", "Romanov", "iu7-61b", "BMSTU").
    student("Kostya", "Naumov", "iu7-61b", "BMSTU").
    student("Kolya", "Lovkin", "iu7-63b", "BMSTU").
    student("Petya", "Arimov", "iu7-63b", "BMSTU").
    student("Vanya", "Gornov", "iu7-62b", "MSU").
    student("Olga", "Buzova", "iu7-23b", "MSU").
    student("Vanya", "Pupkin", "iu7-33b", "MSU").
    getStudentsByUni(N, L, U) :- student(N, L, _, U).

goal
    %student(Name, Surname, Group, "BMSTU").
    %student(Name, Surname, "iu7-61b", "BMSTU").
    student("Vanya", "Pupkin", Group, University).
    %getStudentsByUni(Name, Surname, "MSU").
```

```
[Inactive C:\VIP52\BIN\...
Group=iu7-61b, University=BMSTU
Group=iu7-33b, University=MSU
2 Solutions
```

Рис 3. Студент, обучающийся сразу в двух вузах

```
domains
    Name,Surname, Group, University = symbol.

predicates
    student(Name, Surname, Group, University).
    getStudentsByUni(Name, Surname,University)

clauses
    student("Vanya","Pupkin" ,"iu7-61b", "BMSTU").
    student("Sergey","Romanov" ,"iu7-61b", "BMSTU").
    student("Kostya","Naumov" ,"iu7-61b", "BMSTU").
    student("Kolya", "Lovkin", "iu7-63b", "BMSTU").
    student("Petya", "Arimov", "iu7-63b", "BMSTU").
    student("Vanya","Gornov" ,"iu7-62b", "MSU").
    student("Olga","Buzova" ,"iu7-23b", "MSU").
    student("Vanya","Pupkin" ,"iu7-33b", "MSU").
    getStudentsByUni(N, L, U) :- student(N, L,_, U).

goal
    %student(Name,Surname,Group, "BMSTU").
    %student(Name,Surname,"iu7-61b", "BMSTU").
    %student("Vanya","Pupkin", Group, University).
    getStudentsByUni(Name,Surname,"MSU").
```

[Inactive C:\VIP52\BIN\WIN\32\Obj\goal\$000.exe]

**Name=Vanya, Surname=Gornov**  
**Name=Olga, Surname=Buzova**  
**Name=Vanya, Surname=Pupkin**  
**3 Solutions**

Рис 4. Использование правила

## Теория

Программа на Prolog представляет собой базу знаний и вопрос. База знаний содержит истинностные знания, используя которые программа выдает ответ на запрос. При поиске ответов на вопрос, он рассматривает альтернативные варианты и находит все возможные решения (методом проб и ошибок) — множества значений переменных, при которых на поставленный вопрос можно ответить — «да».

Основным элементом языка является терм. Терм – это:

### 1. Константа:

- Число (целое, вещественное),
- Символьный атом (комбинация символов латинского алфавита, цифр и символа подчеркивания, начинающаяся со строчной буквы: aA, ab\_2), используется для обозначения конкретного объекта предметной области или для обозначения конкретного отношения,
- Строка: последовательность символов, заключенных в кавычки,

### 2. Переменная:

- Именованная – обозначается комбинацией символов латинского алфавита, цифр и символа подчеркивания, начинающейся с прописной буквы или символа подчеркивания ( X, A21, \_X),
- Анонимная - обозначается символом подчеркивания ( \_ ),

### 3. Составной терм

- Составной терм - это средство организации группы отдельных элементов знаний в единый объект, синтаксически представляется:  $f(t_1, t_2, \dots, t_m)$ , где  $f$  - функтор

(функциональный символ) ,  $t_1, t_2, \dots, t_m$  – термы, в том числе и составные (их называют аргументами). Аргументом или параметром составного терма может быть константа, переменная или составной объект. Число аргументов предиката называется его арностью или местностью. Составные термы с одинаковыми функторами, но разной арности, обозначают разные отношения.

Особенность использования переменных:

Именованные переменные уникальны в предикатах одного предложения, анонимные уникальны везде. Анонимные переменные не возвращают значение. Переменной можно обозначить любой объект. При описании переменная может потерять свое значение, но потом его можно вернуть.

С помощью термов и более сложных конструкций языка Prolog – фактов и правил «описываются» знания о предметной области, т.е. база знаний. Используя базу знаний, система Prolog будет делать логические выводы, отвечая на наши вопросы. Таким образом, программа на Prolog представляет собой базу знаний и вопрос.

Программа представляет набор фактов и правил. Факты представляют собой составные термы, с помощью которых фиксируется наличие истинностных отношений между объектами предметной области — аргументами терма.

Правила являются обобщенной формулировкой условия истинности знания – отношения между объектами предметной области (аргументами терма), которое записано в заголовке правила. Условие истинности этого отношения является телом правила.

Утверждения программы — это предикаты. Предикаты могут не содержать переменных (основные) или содержать переменные (не основные). В процессе выполнения программы — система пытается найти, используя базу знаний, такие значения переменных, при которых на поставленный вопрос можно дать ответ «Да».

Программа на Prolog состоит из разделов. Каждый раздел начинается со своего заголовка. Структура программы:

- директивы компилятора — зарезервированные символьные константы
- `CONSTANTS` — раздел описания констант
- `DOMAINS` — раздел описания доменов
- `DATABASE` — раздел описания предикатов внутренней базы данных
- `PREDICATES` — раздел описания предикатов
- `CLAUSES` — раздел описания предложений базы знаний
- `GOAL` — раздел описания внутренней цели (вопроса).

В программе не обязательно должны быть все разделы.