|  |  |
| --- | --- |
| 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»\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Лабораторная работа № 12**

|  |  |
| --- | --- |
| **Студент: Зейналов З. Г.**  **Группа: ИУ7-61Б**  **Оценка (баллы) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Преподаватель:** Толпинская Н. Б. |  |

Москва.

2020 г.

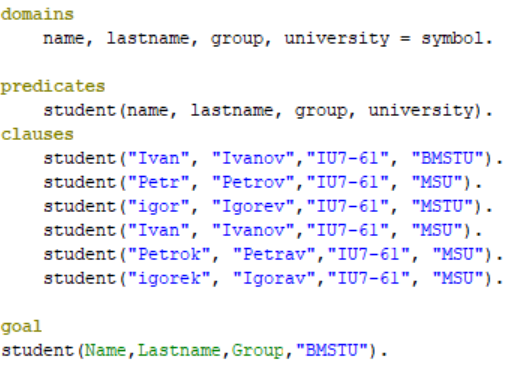
**Введение**

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

**Задачи:** 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.

1. **Исходную базу знаний сформировать с помощью только фактов.**

Листинг 1 - Код программы базы знаний студентов



В базе знаний в виде фактов хранится информация о студентах. Информация представляет собой Имя, Фамилию, группу и Университет.

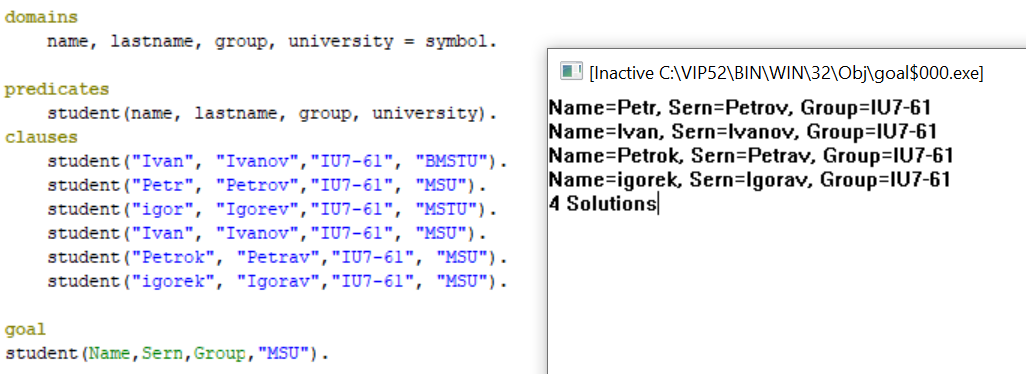


Рисунок 1 - Вывод студентов МГУ без использования правил.

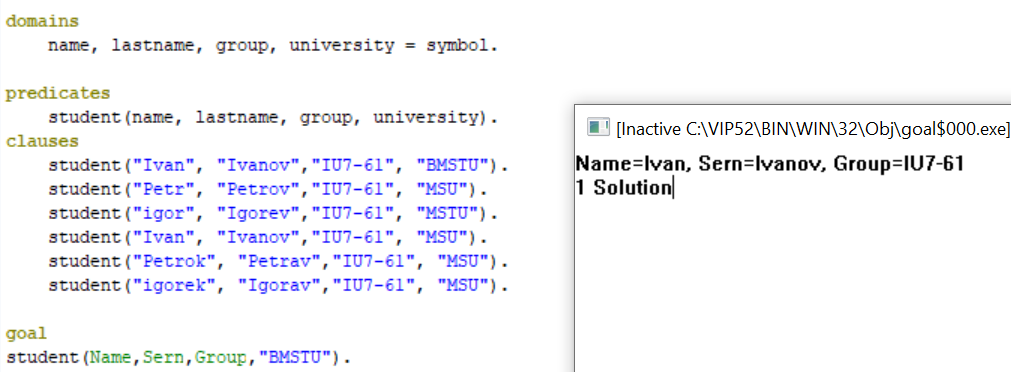


Рисунок 2 - Вывод студентов МГТУ без использования правил.

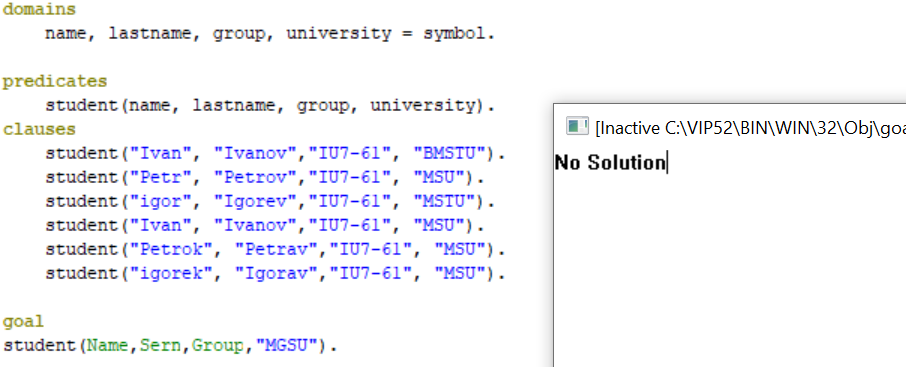


Рисунок 3 - Вывод студентов МГГУ без использования правил.

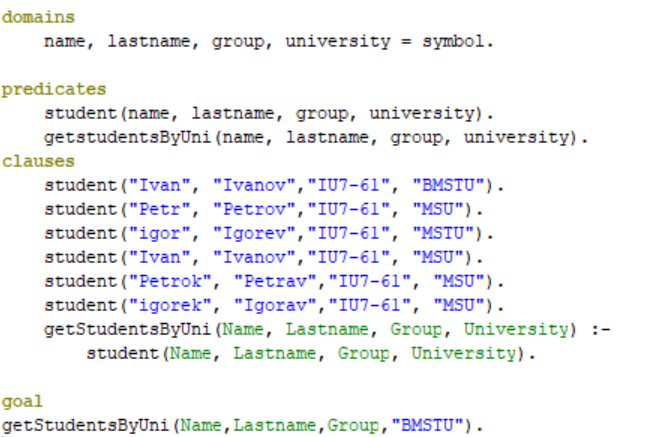
На рисунке 1 приведен пример вывода множества студентов без использования правил, для которых university является “MSU”. Поиск осуществляется с помощью переменных Name, Lastname, Group.

На рисунке 2 представлен Вывод студентов МГТУ.

На рисунке 3 представлен вывод студентов вуза, который отсутствует в базе знаний.

1. **Исходную базу знаний сформировать, используя правила.**

Листинг 2 – Программа с использованием правил.



В базе знаний в виде фактов хранится информация о студентах. Информация представляет собой Имя, Фамилию, группу и Университет. С помощью правила getStudentByUni можно получить различные данные, в том числе всех студентов данного университета.

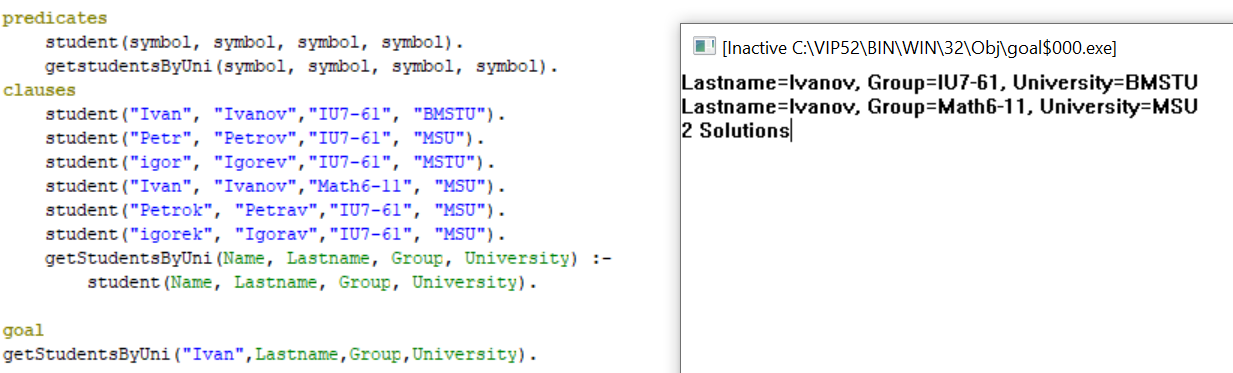


Рисунок 4 - Вывод студентов с именем Ivan c использованием правила.

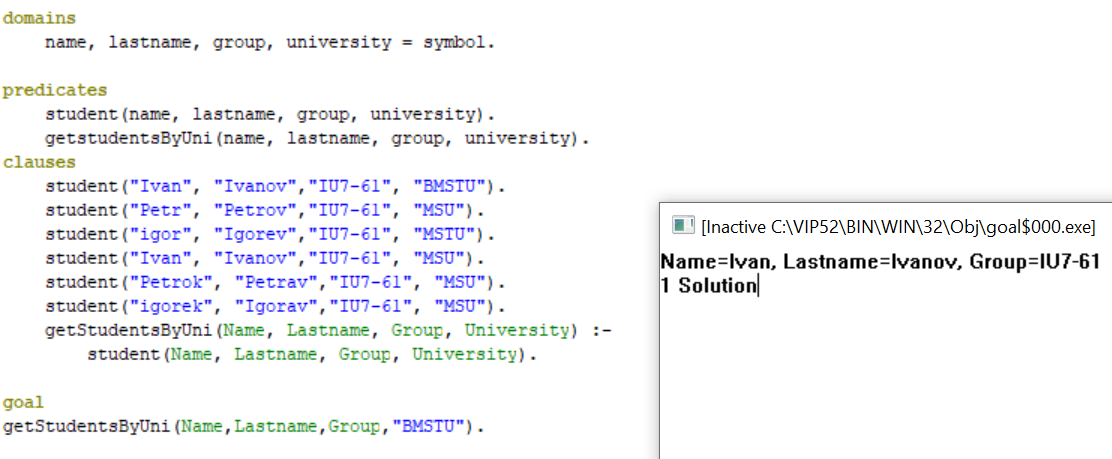


Рисунок 5 - вывод студентов МГТУ с использованием правил.

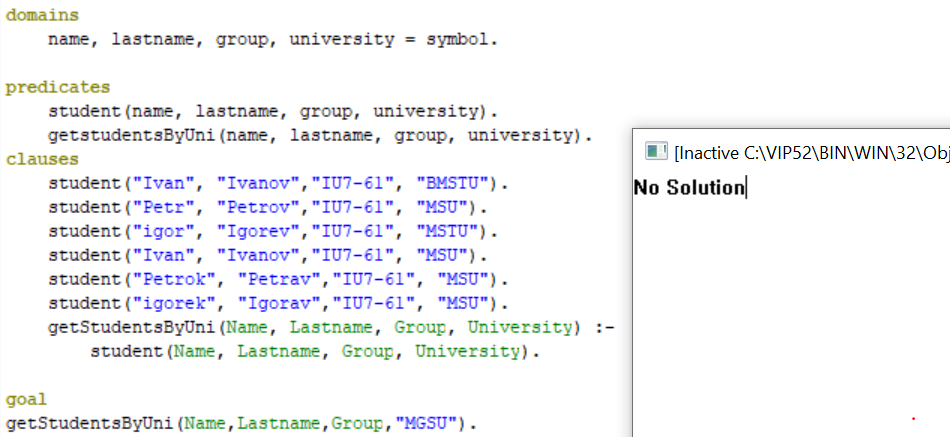


Рисунок 6 - Вывод студентов МГГУ с использованием правил.

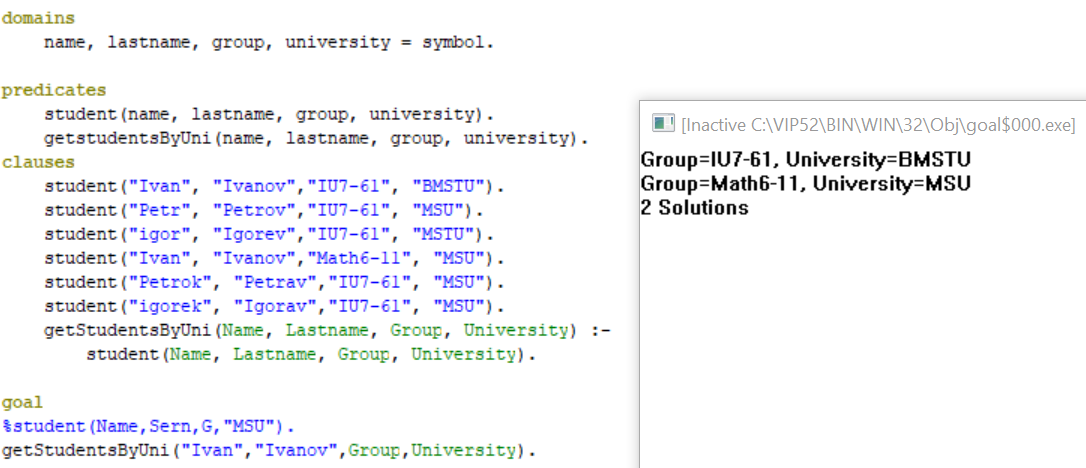


Рисунок 7 - Вывод студента, обучающегося в двух вузах.

На рисунке 4 приведен пример вывод студента с именем “Ivan”

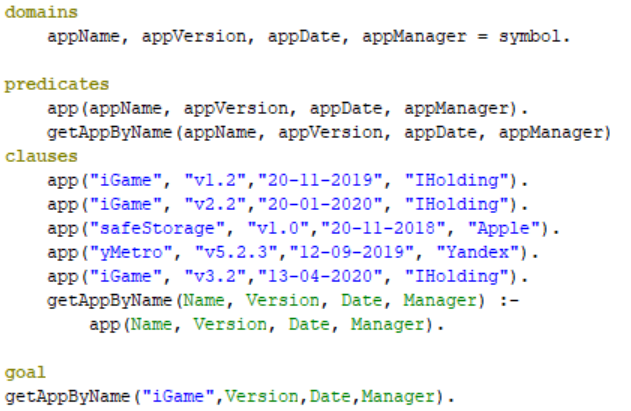
На рисунке 5 представлена реализация с использованием правил. Вывод производится всех студентов вуза МГТУ.

На рисунке 6 представлен вывод студентов, отсутствующего вуза.

На рисунке 7 представлен вывод студента, который учится в 2 вузах одновременно. В результате получаем 2 решения.

1. **Разработать свою базу знаний (содержание произвольно).**

Листинг 3 - Программа с базой знаний о приложениях



База знаний представляет собой информацию о названии приложения, его версии, даты выпуска и владельца приложения. С помощью правила getAppByName можно получить всю необходимую информацию о приложении.

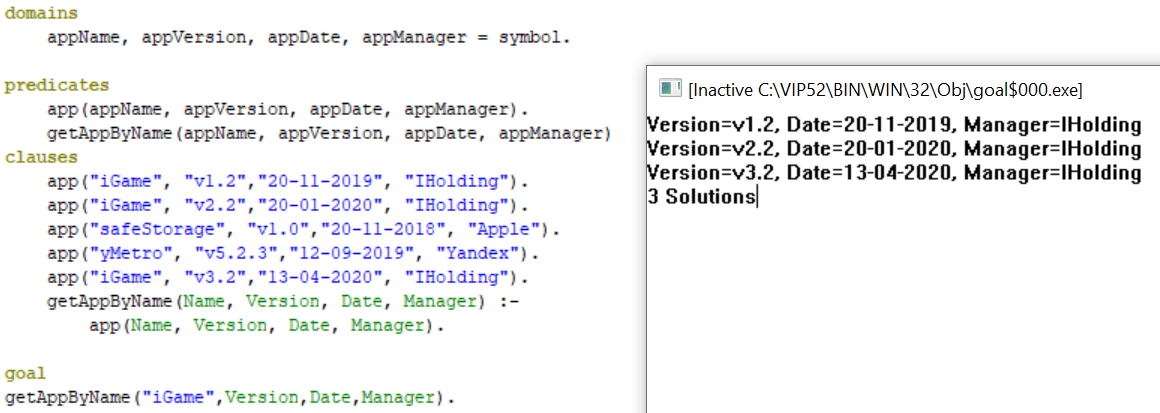


Рисунок 7 - вывод программы по входному названию приложения.

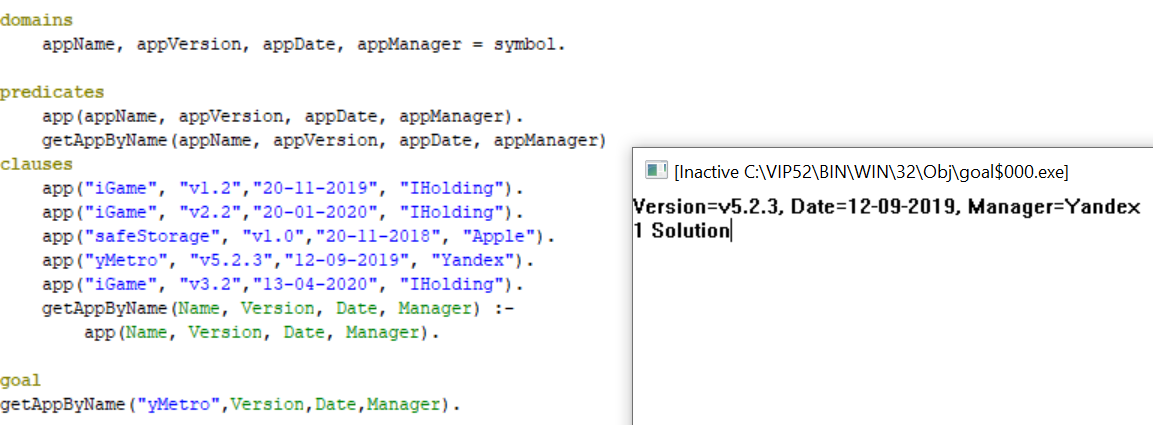


Рисунок 8 - Вывод программы по входному названию приложения.

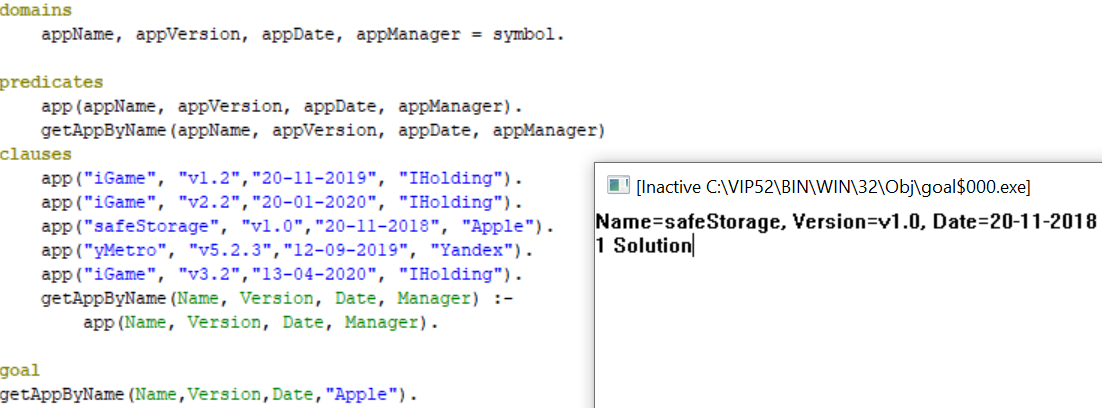


Рисунок 9 - Вывод программы по входному названию владельца.

**Теория:**

Основным элементом языка является Терм.

**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,

1. **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 ( \_ ),

1. **Составной терм:**

* Это средство организации группы отдельных элементов знаний в единый объект, синтаксически представляется: **f(t1, t2, …,tm)**, где f - функтор (функциональный символ) , t1, t2, …,tm – термы, в том числе и составные (их называют аргументами), (например: likes(judy, tennis) – знание о том, что judy любит tennis или еще, например: book( author(tolstoy, liev ), war\_and\_peace) и т.д. ). 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.

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**. **База знаний состоит из** предложений - CLAUSES (отдельных знаний или утверждений**): фактов и правил**. Каждое предложение заканчивается точкой. Предложения бывают двух видов: факты и правила. Предложение более общего вида – **правило** имеет вид:

A :- B1,... , Bn.

**A** называется **заголовком правила**, а B1,..., Bn – **телом правила**. **База знаний состоит из** предложений - CLAUSES (отдельных знаний или утверждений**): фактов и правил**. Каждое предложение заканчивается точкой.

Третьим специфическим видом предложений Prolog можно считать вопросы. Вопрос состоит только из тела – составного терма (или нескольких составных термов). Вопросы используются для выяснения выполнимости некоторого отношения между описанными в программе объектами. Система рассматривает вопрос как цель, к которой (к истинности которой) надо стремиться. Ответ на вопрос может оказаться логически положительным или отрицательным, в зависимости от того, может ли быть достигнута соответствующая цель. Третьим специфическим видом предложений Prolog можно считать вопросы. Вопрос состоит только из тела – составного терма (или нескольких составных термов). Вопросы используются для выяснения выполнимости некоторого отношения между описанными в программе объектами. Система рассматривает вопрос как цель, к которой (к истинности которой) надо стремиться. Ответ на вопрос может оказаться логически положительным или отрицательным, в зависимости от того, может ли быть достигнута соответствующая цель.

Факты, правила, и вопросы могут содержать переменные. В процессе выполнения программы переменные могут связываться с различными объектами – **конкретизироваться**. Это относится только к именованным переменным. Поиск содержательного ответа на поставленный вопрос, с помощью имеющейся базы знаний, фактически заключается в поиске нужного знания, но какое знание понадобится – заранее неизвестно. Этот поиск осуществляется формально с помощью механизма **унификации**, встроенного в систему и не доступного программисту. Упрощенно, процесс унификации можно представить как формальный процесс сравнивания (сопоставления) терма вопроса с очередным термом знания. При этом, знания по умолчанию просматриваются сверху вниз, хотя такой порядок и не очевиден.

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.