



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 №11*

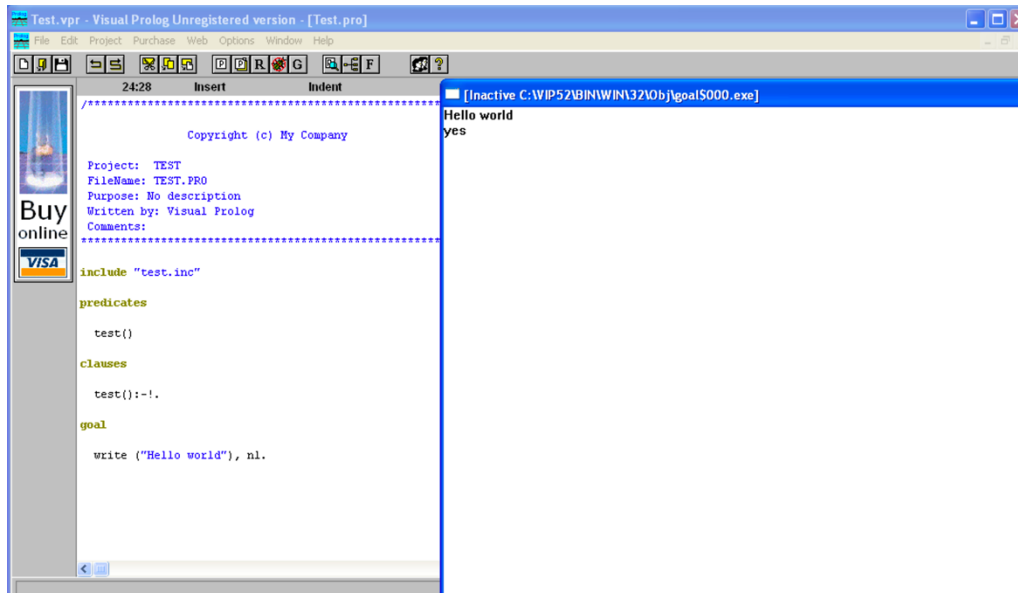
*По предмету: «Функциональное и логическое  
программирование»*

Студент: Лаврова А. А.,  
Группа: ИУ7-65Б  
Преподаватель: Толпинская Н. Б.

Москва, 2020 г.

## Практическая часть

Пример работы тестовой программы №1:



The screenshot shows the Visual Prolog IDE with the file 'Test.pro' open. The editor displays the following code:

```
*****
Copyright (c) My Company

Project: TEST
FileName: TEST.PRO
Purpose: No description
Written by: Visual Prolog
Comments:
*****
include "test.inc"

predicates
    test()

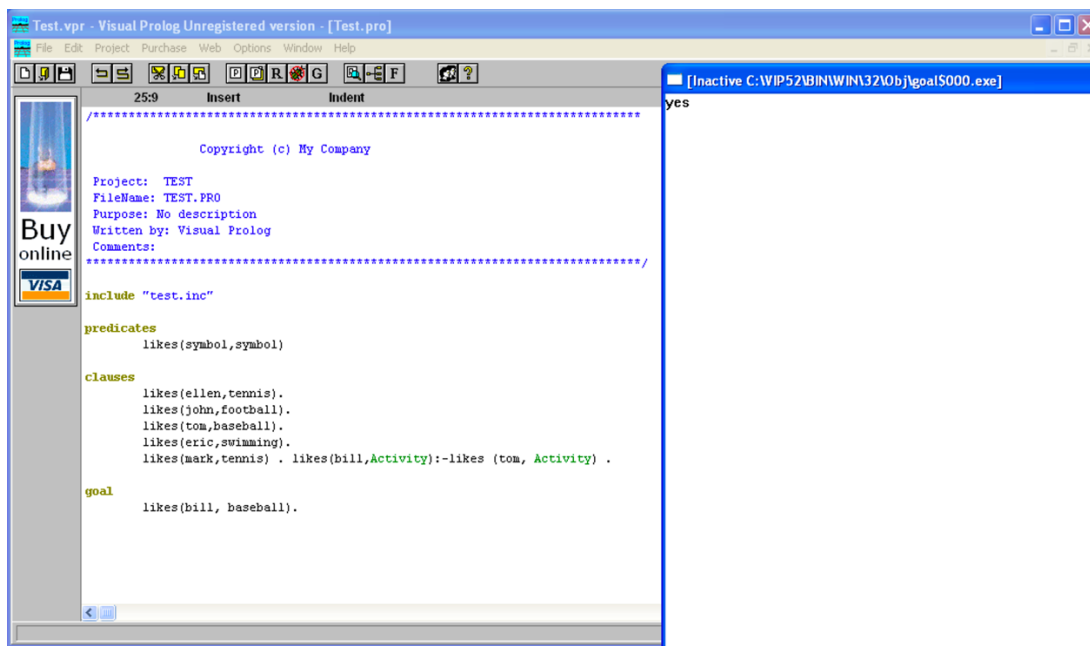
clauses
    test():-!.

goal
    write ("Hello world"), nl.
```

The output window on the right shows the execution results:

```
[Inactive C:\WP52\BIN\WIN32\Obj\goal$000.exe]
Hello world
yes
```

Пример тестовой программы №2:



The screenshot shows the Visual Prolog IDE with the file 'Test.pro' open. The editor displays the following code:

```
*****
Copyright (c) My Company

Project: TEST
FileName: TEST.PRO
Purpose: No description
Written by: Visual Prolog
Comments:
*****
include "test.inc"

predicates
    likes(symbol,symbol)

clauses
    likes(ellen,tennis).
    likes(john,football).
    likes(tom,baseball).
    likes(eric,swimming).
    likes(mark,tennis) . likes(bill,Activity):-likes (tom, Activity) .

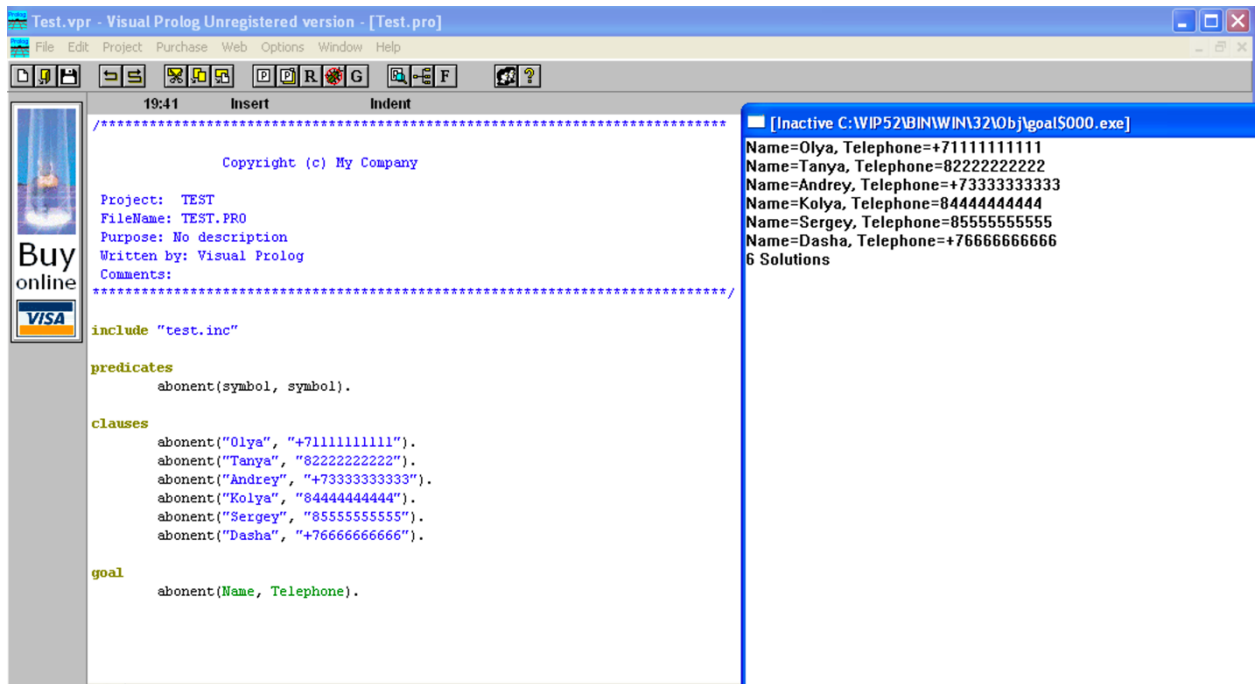
goal
    likes(bill, baseball).
```

The output window on the right shows the execution results:

```
[Inactive C:\WP52\BIN\WIN32\Obj\goal$000.exe]
yes
```

Т. к. прописано правило «все, что любит Том, любит Билл» и есть факт «Том любит бейсбол», то из этого следует, что Билл любит бейсбол. Программа на запрос `likes(bill, baseball)` вывела «yes».

## Реализация программы «Телефонный справочник»:



The screenshot shows the Visual Prolog IDE with a file named 'Test.vpr'. The editor contains a Prolog program for a phone directory. The program includes a copyright notice, project information, and a list of facts (abonent) for six people. A goal is defined to find all solutions for the 'abonent' predicate. The output window on the right shows the results of the query, listing the name and telephone number for each person, followed by the number of solutions found (6).

```
Test.vpr - Visual Prolog Unregistered version - [Test.pro]
File Edit Project Purchase Web Options Window Help

19:41 Insert Insert

/*****
Copyright (c) My Company

Project: TEST
FileName: TEST.PRO
Purpose: No description
Written by: Visual Prolog
Comments:
*****/

include "test.inc"

predicates
    abonent(symbol, symbol).

clauses
    abonent("Olya", "+71111111111").
    abonent("Tanya", "82222222222").
    abonent("Andrey", "+73333333333").
    abonent("Kolya", "84444444444").
    abonent("Sergey", "85555555555").
    abonent("Dasha", "+76666666666").

goal
    abonent(Name, Telephone).
```

[Inactive C:\WIP52\BIN\WIN32\Obj\goal5000.exe]

Name=Olya, Telephone=+71111111111  
Name=Tanya, Telephone=82222222222  
Name=Andrey, Telephone=+73333333333  
Name=Kolya, Telephone=84444444444  
Name=Sergey, Telephone=85555555555  
Name=Dasha, Telephone=+76666666666  
6 Solutions

### Теоретическая часть

Prolog — это декларативный язык программирования, при использовании которого решение задачи получается путем логического вывода из ранее известных положений. Программа на Prolog не является определением последовательности действий, она представляет собой набор фактов и правил, обеспечивающих получение заключений на основе этих утверждений. Prolog базируется на предложениях Хорна, являющихся подмножеством формальной системы, которая называется логикой предикатов. Prolog использует упрощенную версию синтаксиса логики предикатов – предикаты первого порядка.

Основным элементом языка является терм. Терм – это константа, переменная или составной терм.

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:

$$A :- B_1, \dots, B_n.$$

A называется заголовком правила, а  $B_1, \dots, B_n$  – телом правила.

Факт – это частный случай правила. Факт – это предложение, в котором отсутствует тело (т.е. тело пустое).

Причем, A,  $B_1, \dots, B_n$  – это термы; символ ":-" это специальный символ-разделитель.

Заголовок содержит отдельное знание о предметной области (составной терм), а тело содержит условия истинности этого знания. Правило называют условной истиной, а факт, не содержащий тела – безусловной истиной.

Еще одним 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.

Для выполнения логического вывода используется механизм (алгоритм) унификации, встроенный в систему.

Унификация – операция, которая позволяет формализовать процесс логического вывода (наряду с правилом резолюции). С практической точки зрения - это основной вычислительный шаг, с помощью которого происходит:

- Двухнаправленная передача параметров процедурам,
- Неразрушающее присваивание,
- Проверка условий (доказательство).