



**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»

Дисциплина: «Функциональное и логическое  
программирование»

## Лабораторная работа №11

Студент: Левушкин И. К.

Группа: ИУ7-62Б

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

Москва, 2020 г.

# 1 Запуск среды Visual Prolog 5.2 и настройка утилиты TestGoal

Среда *Visual Prolog 5.2* установлена, запущена и настроена утилита *TestGoal* в соответствии с дополнительными материалами к 11 лабораторной работе.

Установка была произведена на операционную систему *Windows XP*, запущенную из-под виртуальной среды *Parallels Desktop*.

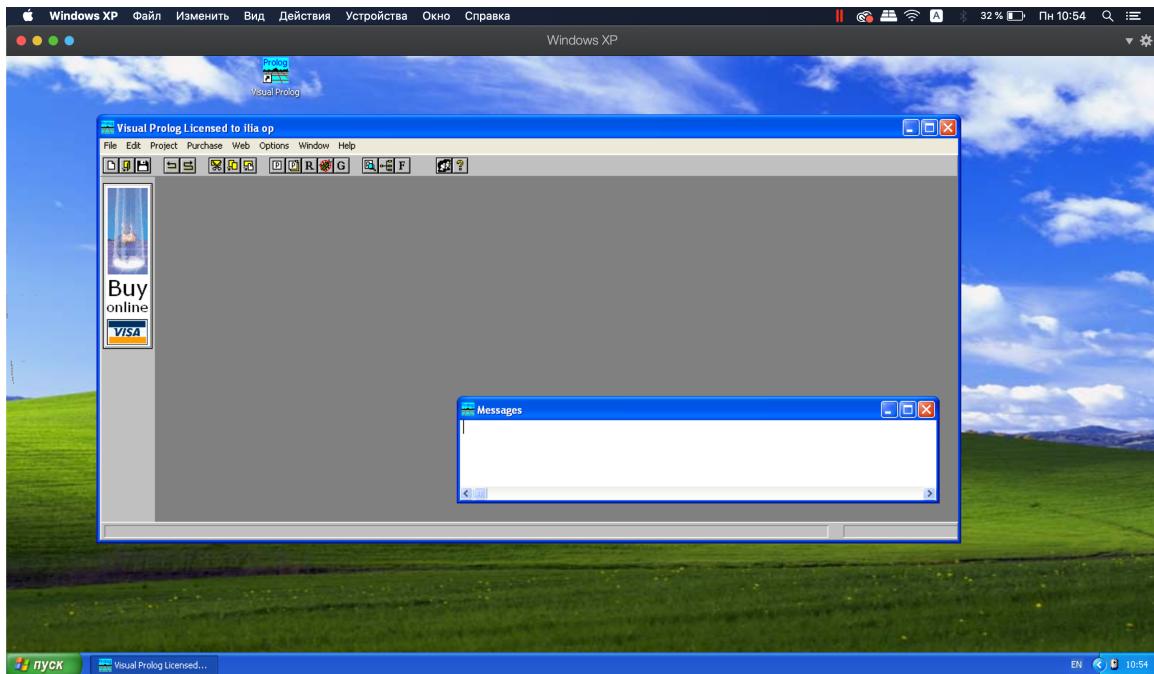


Рис. 1: Запуск среды *Visual Prolog 5.2* на *Windows XP* из-под *Parallels Desktop*.

## 2 Запуск тестовых программ

### 2.1 Пример с выводом множества ответов

Следующая программа демонстрирует возможности утилиты *TestGoal*, а именно: поиск всех решений для определенной в программе цели. Для каждого решения *TestGoal* отображает значения всех переменных из секции *GOAL* и число решений.

```
Goal
X = 2;
X = 1,
Y = X + 1.

X=2
X=1, Y=2
2 Solutions
```

Рис. 2: Вывод режима *TestGoal*.

### 2.2 Пример ch02e01.pro

Следующая программа создает предикат *likes(symbol, symbol)*. И в разделе *clauses* записывается 4 факта и 1 правило: Биллу нравится то же, что и том: *likes(bill, Activity):- likes(tom, Activity)*. В разделе *goal* на первом скриншоте задается вопрос: *Bill likes baseball?*

Поскольку Тому нравится baseball, ответ да:

```
predicates
    likes(symbol,symbol) - nondet (i,i)

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).
```

```
[Inactive C:\DOCUME...]
yes
```

Рис. 3: Программа ch02e01\_1.pro.

В разделе goal на втором скриншоте задается вопрос: *Bill likes tennis?*

Поскольку ни Билл, ни Том не увлекаются теннисом, ответ нет:

```
predicates
    likes(symbol,symbol) - nondetrm (i,i)

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,tennis).
```

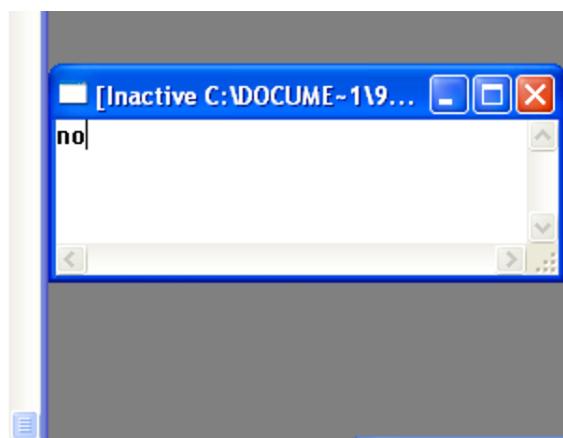


Рис. 4: Программа ch02e01\_2.pro.

### 3 Программа Телефонный справочник

Ниже приведена программа телефонный справочник, использующая только один предикат: *phone\_list(string First, string Last, string Phone, string Month, integer Day, integer Year)*:

```
predicates
    phone_list(string First, string Last, string Phone, string Month, integer Day, integer Year)

clauses
    phone_list("Ilya", "Levushkin", "8-985-977-14-92", "december", 11, 1999).
    phone_list("Pavel", "Pavlovich", "8-985-977-14-93", "may", 12, 1999).
    phone_list("Pavel", "Pavlov", "8-985-977-14-94", "may", 13, 2000).
    phone_list("Anton", "Antonich", "8-985-977-14-95", "may", 12, 2000).
    phone_list("Anton", "Anton", "8-985-977-14-96", "may", 13, 2000).
    phone_list("Michel", "Michelich", "433-9906", "may", 12, 1982).
    phone_list("Michel", "Michel", "433-9907", "may", 13, 1982).
    phone_list("Dima", "Dimaich", "8-985-977-14-97", "may", 12, 2001).
    phone_list("Dima", "Dima", "8-985-977-14-98", "may", 13, 2001).
    phone_list("Vova", "Vovaich", "8-985-977-14-99", "may", 12, 2002).
    phone_list("Vova", "Vova", "8-985-977-14-00", "may", 13, 2002).
    phone_list("Roma", "Romaich", "8-985-977-14-01", "may", 12, 2003).
    phone_list("Roma", "Roma", "8-985-977-14-02", "may", 13, 2003).
    phone_list("Olya", "Olyaich", "8-985-977-14-03", "may", 12, 2004).
    phone_list("Olya", "Olya", "8-985-977-14-04", "may", 13, 2004).
    phone_list("Sveta", "Svetaich", "8-985-977-14-05", "may", 12, 2005).
    phone_list("Sveta", "Sveta", "8-985-977-14-06", "may", 13, 2005).

goal
    phone_list("Ilya", "Levushkin", Phone, Month, Day, Year).
```

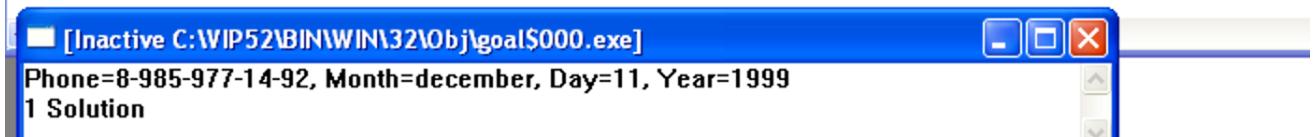


Рис. 5: Программа Test.pro.

В разделе goal задается вопрос: Есть ли *Ilya Levushkin* в телефонной книге?

Ответ будет содержать всех таких *Ilya Levushkin* (в данном случае одного).

```
predicates
    phone_list(string First, string Last, string Phone, string Month, integer Day, integer Year)

clauses
    phone_list("Ilya", "Levushkin", "8-985-977-14-92", "december", 11, 1999).
    phone_list("Pavel", "Pavlovich", "8-985-977-14-93", "may", 12, 1999).
    phone_list("Pavel", "Pavlov", "8-985-977-14-94", "may", 13, 2000).
    phone_list("Anton", "Antonich", "8-985-977-14-95", "may", 12, 2000).
    phone_list("Anton", "Anton", "8-985-977-14-96", "may", 13, 2000).
    phone_list("Michel", "Michelich", "433-9906", "may", 12, 1982).
    phone_list("Michel", "Michel", "433-9907", "may", 13, 1982).
    phone_list("Dima", "Dimaich", "8-985-977-14-97", "may", 12, 2001).
    phone_list("Dima", "Dima", "8-985-977-14-98", "may", 13, 2001).
    phone_list("Vova", "Vovaich", "8-985-977-14-99", "may", 12, 2002).
    phone_list("Vova", "Vova", "8-985-977-14-00", "may", 13, 2002).
    phone_list("Roma", "Romaich", "8-985-977-14-01", "may", 12, 2003).
    phone_list("Roma", "Roma", "8-985-977-14-02", "may", 13, 2003).
    phone_list("Olya", "Olyaich", "8-985-977-14-03", "may", 12, 2004).
    phone_list("Olya", "Olya", "8-985-977-14-04", "may", 13, 2004).
    phone_list("Sveta", "Svetaich", "8-985-977-14-05", "may", 12, 2005).
    phone_list("Sveta", "Sveta", "8-985-977-14-06", "may", 13, 2005).

goal
    phone_list("Pavel", Last, Phone, Month, Day, Year).
```

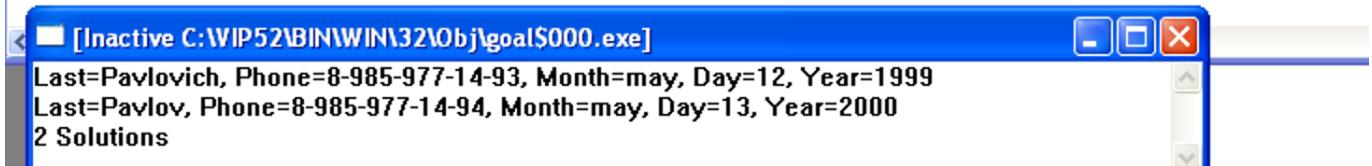


Рис. 6: Программа Test.pro.

В разделе goal задается вопрос: Есть ли *Pavel* в телефонной книге?

Ответ будет содержать всех таких *Pavel* (в данном случае двоих).

```

predicates
  phone_list(string First, string Last, string Phone, string Month, integer Day, integer Year)

clauses
  phone_list("Ilya", "Levushkin", "8-985-977-14-92", "december", 11, 1999).
  phone_list("Pavel", "Pavlovich", "8-985-977-14-93", "may", 12, 1999).
  phone_list("Pavel", "Pavlov", "8-985-977-14-94", "may", 13, 2000).
  phone_list("Anton", "Antonich", "8-985-977-14-95", "may", 12, 2000).
  phone_list("Anton", "Anton", "8-985-977-14-96", "may", 13, 2000).
  phone_list("Michel", "Michelich", "433-9906", "may", 12, 1982).
  phone_list("Michel", "Michel", "433-9907", "may", 13, 1982).
  phone_list("Dima", "Dimaich", "8-985-977-14-97", "may", 12, 2001).
  phone_list("Dima", "Dima", "8-985-977-14-98", "may", 13, 2001).
  phone_list("Vova", "Vovaich", "8-985-977-14-99", "may", 12, 2002).
  phone_list("Vova", "Vova", "8-985-977-14-00", "may", 13, 2002).
  phone_list("Roma", "Romaich", "8-985-977-14-01", "may", 12, 2003).
  phone_list("Roma", "Roma", "8-985-977-14-02", "may", 13, 2003).
  phone_list("Olya", "Olyach", "8-985-977-14-03", "may", 12, 2004).
  phone_list("Olya", "Olya", "8-985-977-14-04", "may", 13, 2004).
  phone_list("Sveta", "Svetaich", "8-985-977-14-05", "may", 12, 2005).
  phone_list("Sveta", "Sveta", "8-985-977-14-06", "may", 13, 2005).

goal
  phone_list(First, Last, Phone, Month, Day, 2006).

```



Рис. 7: Программа Test.pro.

В разделе goal задается вопрос: Есть ли кто-то 2006-ого года рождения в телефонной книге?

Ответ будет содержать всех таких людей, кто родился в 2006 году (в данном случае ни одного).

## 4 Ответы на вопросы

### 4.1 Что собой представляет программа на Prolog?

Программа на Prolog представляет собой базу знаний и вопрос.

- База знаний состоит из предложений - CLAUSES (отдельных знаний или утверждений): фактов и правил.
- 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.

## 4.2 Какова структура программы на Prolog?

- База знаний состоит из предложений - CLAUSES (отдельных знаний или утверждений): фактов и правил.
- 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.

## 4.3 Как она реализуется на Prolog?

- Предложения бывают двух видов: факты и правила;
- Каждое предложение заканчивается точкой;
- Предложение более общего вида – правило имеет вид:  $A : -B_1, \dots, B_n$   
А называется заголовком правила, а  $B_1, \dots, B_n$  – телом правила;
- Факт – это предложение, в котором отсутствует тело (частный случай правила);

## 4.4 Как формируются результаты работы программы?

Ответ на поставленный вопрос система дает в логической форме – «Да» или «нет». Цель системы состоит в том, чтобы на поставленный вопрос найти возможность, исходя из базы знаний, ответить «Да». Вариантов ответить «Да» на поставленный вопрос может быть несколько. Система может (в нашем случае обучения – должна) быть настроена в режим получения всех возможных вариантов ответа «Да» на поставленный вопрос.

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.