

## ИЗУЧЕНИЕ ЭЛЕМЕНТОВ ЛОГИЧЕСКОГО ПРОГРАММИРОВАНИЯ В РАМКАХ ФАКУЛЬТАТИВА ПО ИНФОРМАТИКЕ

Акимова И.В.<sup>1</sup>, Губанова О.М.<sup>1</sup>, Титова Н.В.<sup>1</sup>

<sup>1</sup>ФГБОУ ВО «Пензенский государственный университет», Пенза, e-mail: [ulrih@list.ru](mailto:ulrih@list.ru)

В данной статье авторы поднимают проблему изучения элементов логического программирования в средней школе. Авторы анонсируют, что одной из задач школьного информационно-технологического образования является развитие мышления учащегося, как наглядно-образного, наглядно-действенного, так и логического. Данная задача может быть достигнута с помощью изучения логической парадигмы программирования. Актуальность темы исследования обуславливается необходимостью знакомства учащихся, интересующихся информатикой, с логической парадигмой программирования с одной стороны, и отсутствием в действующих школьных учебниках даже частичного рассмотрения данной темы. Попытки методической разработки данной темы делались со стороны ведущих отечественных методистов, но не нашли широкого применения в школьной практике. В качестве методического решения данной проблемы авторы выбирают факультатив «Основы логического программирования», который затрагивает изучение основных тем языка Prolog. В статье рассматриваются цели и задачи факультатива, приводится тематическое планирование. Подробно рассматривается с методических позиций изучение одной из тем – «Встроенные предикаты языка программирования Prolog». Данный факультатив был внедрен в процесс преподавания информатики в МБОУ «СОШ № 68 с углубленным изучением информатики» г. Пензы и имел положительный методический эффект.

Ключевые слова: парадигма программирования, факультатив, Prolog.

## STUDYING OF ELEMENTS OF LOGICAL PROGRAMMING WITHIN THE OPEN CLASSROOM ON INFORMATICS

Akimova I.V.<sup>1</sup>, Gubanova O.M.<sup>1</sup>, Titova N.V.<sup>1</sup>

<sup>1</sup>Penza State University, Penza, e-mail: [ulrih@list.ru](mailto:ulrih@list.ru)

In the article authors lift a problem of studying of elements of logical programming at high school. Authors announce that one of problems of school information and technological education is the development of thinking of the pupil as evident and figurative, evident and effective, and logical. This task can be reached by means of studying of a logical paradigm of programming. The relevance of a subject of a research is caused by need of acquaintance of the pupils who are interested in informatics with a logical paradigm of programming on the one hand, and absence in the existing school textbooks even of partial consideration of this subject. Attempts of methodical development of this subject became from the leading domestic methodologists, but haven't found broad application in school practice. As the methodical solution of this problem authors choose an open classroom of "A basis of logical programming" which are affected by studying of the main subjects of the Prolog language. In article the purposes and problems of an open classroom are considered, thematic planning is given. In detail studying of one of subjects – "The built-in predicates of the Prolog programming language" is considered from methodical positions. This open classroom has been introduced in teaching process to informatics in school №. 68 with profound studying of informatics of Penza and had positive methodical effect.

Keywords: programming paradigm, open classroom, Prolog.

Одной из задач школьного информационно-технологического образования является развитие мышления учащегося, как наглядно-образного, наглядно-действенного, так и логического. Формирование логического мышления – важная составная часть педагогического процесса, которая невозможна без изучения основ математической логики. Тема «Логика, логические основы ЭВМ», на наш взгляд, достаточно представлена в школьном образовании. Но в основном упор идет на рассмотрение теоретического материала, решение логических задач. А основной задачей должно являться развитие

логического аппарата учащихся.

На протяжении всей истории школьной информатики изучение элементов программирования является основой курса, хотя может быть представлено в различном процентном соотношении. Такая популярность объясняется тем, что алгоритмизация и программирование являются не только целью, но и средством формирования мышления учащегося, его развитием, а также закладывают основу для продолжения обучения в старших профильных классах [1-3]. Таким образом, темой нашего исследования становится методика работы с логическим языком программирования Prolog.

Данный язык программирования относится к классу так называемых декларативных языков. Работа с ним предполагает учет его особенностей, с одной стороны, а с другой - является мощным средством для развития логического аппарата учащихся, предполагает знакомство с несколькими парадигмами программирования, расширяет мировоззренческие основы курса информатики в средней школе.

Анализ действующих школьных учебников по информатике различных авторов 9-11 классов, показал, что авторы отдают предпочтение императивной процедурной и объектно-ориентированной парадигме программирования. Авторы выбирают либо язык Pascal (Босова Л.Л., Босова А.Ю. [4-5], Семакин И.Г., Залогова Л.А. [6], Поляков К.Ю. [7]), либо язык Visual Basic (Угринович Н.Д. [8]). В то время как знакомство с логической парадигмой имеет важное значение как для формирования информационно-технологической компетенции учащихся, так и для их дальнейшего профориентационного развития, формирования мышления в целом.

Идеи обучения элементам логического программирования не являются новыми в методике обучения информатике. В середине 80-90 гг. в работах С.Г. Григорьева, Е.А. Ерохиной, С.А. Бешенкова, В.А. Каймина, Н.Д. Угриновича, А.Г. Щеголева [3; 9; 10] предпринимались попытки внедрения логического программирования в школы, были разработаны методические материалы по логическому программированию [11]. Но широкого применения данные методические идеи не находят. Поэтому задачей нашего исследования становится внедрение изучения элементов логического программирования в рамках факультативного курса в среднюю школу, в частности в МБОУ «СОШ № 68 с углубленным изучением информатики» г. Пензы [12].

Далее рассмотрим программу факультатива «Основы логического программирования».

*Цель факультатива* – освоить основы программирования на логическом языке программирования Prolog, что должно способствовать интеллектуальному и творческому развитию школьников, совершенствованию их логического и алгоритмического мышления.

Освоение программы предполагает повышение мотивации к обучению на основе формирования личностных и предметных компетенций, универсальных учебных действий.

#### *Задачи факультатива*

##### на личностном уровне:

- формирование образного, логического и алгоритмического мышления школьников;
- развитие навыков анализа и самоанализа;
- воспитание целеустремленности и усидчивости в процессе творческой работы и учебной деятельности;
- формирование готовности к выбору профильного образования, связанного с профессией программиста или частично связанного с ней;

##### на предметном уровне:

- формирование основных понятий математической логики;
- формирование понятий об основных конструкциях языка Prolog;
- формирование знаний об основных предикатах языка Prolog;
- формирование знаний об основных типах и структурах данных;

##### на метапредметном уровне:

- создание и преобразование моделей и схем для решения задач, построение логического рассуждения, включающее установление причинно-следственных связей;
- осуществление выбора наиболее эффективных способов решения логических задач на компьютере в зависимости от конкретных условий;
- овладение составляющими проектной деятельности, включая умения видеть проблему, формулировать тему и цель проекта, составлять план своей деятельности, осуществлять действия по реализации плана, результат своей деятельности соотносить с целью, классифицировать, наблюдать, проводить эксперименты, делать выводы и заключения, доказывать, защищать свои идеи, оценивать результаты своей работы;
- проектирование и организация своей индивидуальной и групповой деятельности, организация своего времени с использованием ИКТ.

В таблице представлено тематическое планирование факультатива.

#### Тематическое планирование учебного материала

№	Название темы	Кол-во часов
<b>1 полугодие</b>		
1	Основные понятия математической логики	2
2	Основные понятия языка программирования Prolog:	2

№	Название темы	Кол-во часов
	предикаты, операции над предикатами	
3	Основы работы в среде программирования Visual Prolog	2
4	Встроенные предикаты языка программирования Prolog	4
5	Понятие рекурсивного алгоритма, виды рекурсии. Реализация рекурсивных алгоритмов в языке программирования Prolog	6
6	Обработка строковых данных в языке программирования Prolog	4
7	Выполнение индивидуального задания	2
8	Понятие списка. Работа со списками в языке программирования Prolog	8
9	Выполнение индивидуального задания	4
10	Резерв	2
	Итого	36

Приведем пример содержания некоторых тем.

#### Тема 4. Встроенные предикаты языка программирования Prolog

Встроенные предикаты для ввода данных, встроенные предикаты для вывода данных, встроенные математические предикаты, встроенные арифметические предикаты.

##### *Практическое задание*

1. Запишите правила, являющиеся решениями следующих заданий:

а) даны 2 числа  $a$  и  $b$ , получить их сумму, разность, произведение;

б) дана длина ребра куба, найти его объем и площадь боковой поверхности;

в) даны радиус основания и высота цилиндра, найти его объем и площадь поверхности;

г) даны стороны  $a$ ,  $b$  параллелограмма и угол между ними, найти площадь параллелограмма.

2. Вычислить:

1)  $2x + \frac{1}{3}y + 5.6;$

2)  $(2x + 8y + 4)/2;$

3)  $y - x^2;$

- 4)  $x^2+xy+y^2$ ;
- 5)  $\sin(x)+\cos(3y)$ ;
- 6)  $5x+\sqrt{34z}+\frac{y}{3}$ .

3. Написать программу решения квадратного уравнения.

#### *Методические рекомендации*

Как было сказано ранее, в языке Prolog нет понятия «оператор». Все основные действия реализуются с помощью встроенных предикатов. Поэтому задачей учителя становится донести до ученика важность использования встроенных предикатов, особенности их работы. Например, что предикат `readint` всегда является только одноместным, то есть нет возможности ввода с его помощью одновременно более одного целого значения с клавиатуры.

При решении задач также важным становится момент работы с механизмом отката, что является вторым важным моментом в данной теме. При решении задачи поиска корней квадратного уравнения учитель может начать с более простой задачи – решения линейного уравнения. Именно на ней можно продемонстрировать приемы программирования на Prolog, реализацию такой конструкции, как ветвления, которая как таковая в структуре языка отсутствует.

Приведем пример решения задачи «поиск решения линейного уравнения».

Учитель предлагает учащимся вспомнить из курса алгебры алгоритм решения линейного уравнения  $a \cdot x + b = 0$ .

Вместе учащимися рассматриваются следующие случаи:

- 1)  $a=0, b=0$ . В этом случае уравнение принимает вид  $0 \cdot x + 0 = 0$ . Его решением является любое число;
- 2)  $a=0, b \neq 0$ . В этом случае уравнение принимает вид  $0 \cdot x + b = 0$ . В этом случае у уравнения нет корней;
- 3)  $a \neq 0$ . В этом случае корень уравнения находится по формуле:  $x = -\frac{b}{a}$ .

Решение уравнения необходимо оформить в виде правил для предиката. Учащиеся предлагают назвать предикат `urav`. Каков будет его ранг? Ранг равен двум. Далее записываем правила.

Здесь учитель может предложить использовать такой прием написания программы на языке Prolog, как использование специальных нуль-местных предикатов, в правилах для которых будут вводиться данные. Назовем такой предикат `do`. Его правило имеет вид:

`do:- readreal(A), readreal(B), urav(A,B).`

Этот же предикат и будет вынесен в цель согласования.

В целом программа примет следующий вид (рис. 1).

```
include "proj111.inc"

predicates

    urav(real,real)
    do

clauses
|
urav(A,B):-A=0,B=0,write("любое число"),!.
urav(A,B):-A=0,B<>0,write("нет корней"),!.
urav(A,B):-X=-B/A, write("X=",X).
do:-readreal(A), readreal(B), urav(A,B).

goal
do.
```

Рис. 1. Листинг программы

В этом же примере учитель может раскрыть еще несколько приемов программирования на Prolog.

Учитель обращает внимание учащихся на первое правило:

`urav(A,B):-A=0, B=0, write("любое число"),!.`

Часто для сокращения текста программы данное условие  $A=0, B=0$  можно непосредственно сформулировать в заголовке правила в следующем виде:

`urav(0,0):- write("любое число"),!.`

Тогда программа запишется в следующем виде (рис. 2).

```
Comments:
*****

include "proj111.inc"

predicates

    urav(real,real)
    do

clauses

urav(0,0):-write("любое число"),!.
urav(A,B):-A=0,B<>0,write("нет корней"),!.
urav(A,B):-X=-B/A, write("X=",X).
do:-readreal(A), readreal(B), urav(A,B).

goal
do.
```

Рис. 2. Листинг программы

Учитель предлагает учащимся самостоятельно исправить второе правило, которое отражает случай  $a=0, b \neq 0$ .

```
urav(0,B):- B<>0, write("нет корней"),!.
```

Учитель задает вопрос: а важно ли нам значение коэффициента B? Надо ли сравнивать его с 0? Ведь случай равенства 0 был рассмотрен в первом правиле. Нет, значение B нам уже не важно. Тогда предлагаем еще раз переписать правило, используя анонимную переменную, в следующем виде:

```
urav(0,_):- write("не корней"),!.
```

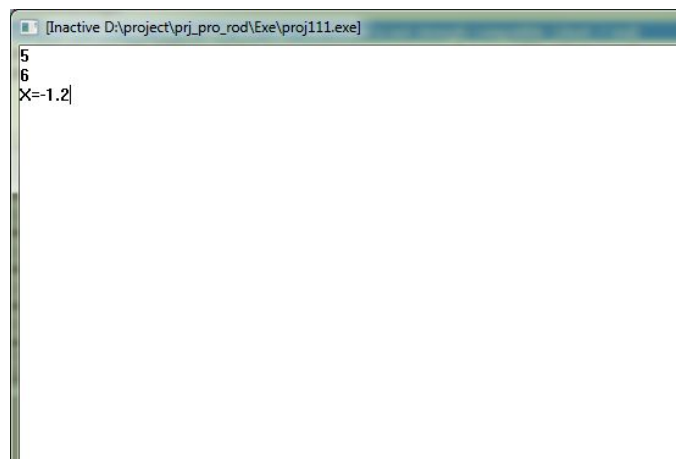
Окончательно программа примет следующий вид (рис. 3).

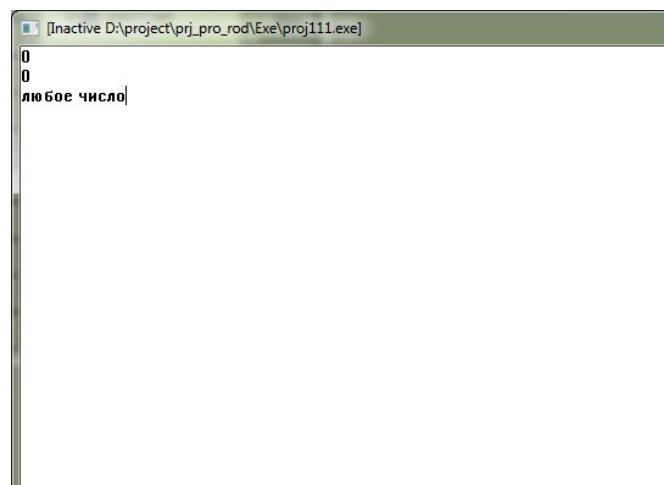
```
*****/
include "proj111.inc"
predicates
    urav(real,real)
do
clauses
|
urav(0,0):-write("любое число"),!.
urav(0,_):-write("нет корней"),!.
urav(A,B):-X=-B/A, write("X=",X).
do:-readreal(A), readreal(B), urav(A,B).

goal
do.
```

Рис. 3. Листинг программы

Результаты работы представлены на рисунке 4.





*Рис. 4. Результат работы программы*

Здесь же учитель может подчеркнуть важность использования предиката `cut (!)`, предложив ученикам протестировать работу программы без использования данного предиката.

После решения данного примера учитель предлагает ученикам самостоятельно разработать и протестировать программу решения квадратного уравнения.

Данный факультатив был внедрен в МБОУ «СОШ № 68 с углубленным изучением информатики» г. Пензы. В течение учебного года учащиеся 9-х классов посещали данный факультатив на добровольной основе. В конце учебного года среди учащихся была проведена контрольная работа и опрос с целью выявить эффективность предложенного методического решения. Проведенная контрольная работа показала высокий уровень освоения учащимися основных тем факультатива. В состав контрольной работы входили задачи, для решения которых учащиеся могли использовать как язык Prolog, так и сделать выбор в пользу той или иной парадигмы.

В результате проведенного опроса большинство учеников (95%) высказались в необходимости данного факультатива в школьном курсе. Большая часть опрошиваемых (80%) первоначально испытывали трудности при посещении занятий и выполнении заданий. На наш взгляд, это объясняется попыткой решения задач с использованием ранее им знакомой императивной парадигмы. В ходе работы на факультативных занятиях и подробного освоения основных элементов логического языка программирования данные трудности были преодолены. В качестве пожеланий дальнейшего совершенствования курса учащиеся предложили расширить задачный материал, учителю рассматривать больше практических примеров.

Таким образом, можно сделать вывод, что разработанный нами факультатив «Основы логического программирования» является эффективным средством изучения элементов



логического программирования в средней школе.

### Список литературы

1. Акимова И.В., Родионов М.А. Методика изучения темы «массивы» при обучении программированию бакалавров педагогических специальностей профиля «Информатика» // Информатика и образование. – 2004. – № 3. – С. 20-24.
2. Губанова О.М. Проектирование курсов методической направленности // Известия Пензенского государственного педагогического университета им. В.Г. Белинского. – 2011. – № 26. – С. 446-451.
3. Дудина И.П. Методическая система обучения основам логического программирования в профессиональном образовании учителя информатики: дис. ... канд. пед. наук: 13.00.08. – Тольятти, 1997. – 182 с.
4. Босова Л.Л., Босова А.Ю. Информатика и ИКТ: учебник для 8 класса. – М., 2013. – 220 с.
5. Босова Л.Л., Босова А.Ю. Информатика и ИКТ: учебник для 9 класса. – М., 2013. – 182 с.
6. Семакин И.Г., Залогова Л.А. и др. Информатика и ИКТ. Учебник для 9 класса. - 5-е изд. - М., 2012. – 341 с.
7. Поляков К.Ю. Информатика. 9 класс: учебник. – М.: БИНОМ. Лаборатория знаний, 2016. – 288 с.
8. Угринович Н.Д. Информатика и ИКТ: учебник для 9 класса. – М., 2012. – 295 с.
9. Жужжалов В.Е. Совершенствование содержания обучения программированию на основе интеграции парадигм программирования: автореф. дис. ... докт. пед. наук: 13.00.02. – М., 2004. – 48 с.
10. Каймин В.А. и др. Основы информатики и вычислительной техники: проб. учеб. пособ. для 10-11 кл. сред. шк. / В.А. Каймин, А.Г. Щеголев, Е.А. Ерохина, Д.П. Федюшин. – М.: Просвещение, 1989. – 272 с.
11. Жемчужников Д.Г. Методика обучения программированию, основанная на создании школьниками динамических компьютерных игр: дис. ... докт. пед. наук: 13.00.02. – М., 2013. – 226 с.
12. Акимова И.В. Использование интерактивных программных средств при обучении программированию // Информатика в школе. – 2012. – № 9. – С. 49-50.