

А. Г. Леонов, Ю. А. Первин

### Элементы программирования в непрерывном курсе школьной информатики

Новейшая история отечественной школьной информатики прошла через разные этапы своего становления: от первого государственного курса с несколькими языками программирования до полного исключения элементов программирования из учебного плана. Здесь описываются стереотипы, встретившиеся на этом пути, и рассказывается о той роли, которую сыграл процедурный язык КуМир в борьбе за достойное место программирования в школьном учебном курсе информатики на всех его этапах.

**Ключевые слова:** непрерывный школьный курс информатики, программирование, пропедевтический курс информатики, базовый курс, предпрофессиональное обучение, процедурные учебно-ориентированные языки.

A. G. Leonov, Yu. A. Pervin

### Arguments for Programming Elements in the Non-Stop Course of School Informatics

The contemporary history of domestic school of Informatics passed through the different stages of its development, from the first public course with several programming languages to the complete exclusion of the programming elements from the curriculum. It describes the stereotypes encountered along the way and talks about the role played by the procedural language KuMir in the fight for its rightful place in the school programming tutorial course in all its phases.

**Keywords:** non-stop Informatics school course, programming, an introductory course of Informatics, a basic course, pre-training, school-oriented procedural languages.

Информатику как фундаментальную науку принято связывать с появлением известной книги Н. Винера «Кибернетика» [1], написанной в 1948 году (и впервые переведенной на русский язык только тридцать пять лет спустя). Однако потребовалось достаточно много времени, чтобы сформировалось научное и практическое направление – *программирование* для вычислительных машин, которое непосредственно породило свое теоретическое обобщение – *информатику*. И лишь в результате школьной реформы в нашей стране чуть менее 30 лет тому назад, в 1985 году, информатика появилась в отечественной общеобразовательной школе как обязательная образовательная дисциплина.

Неудивительно, что в информационном обществе, о становлении которого активно говорили в то время в нашем государстве, появление школьного предмета «Информатика» воспринималось как обучение школьников программированию. Это привело к рождению одного из стереотипов школьной информатики, которыми так насыщена новейшая история отечественной информатизации образования. В первом государственном школьном учебнике «Основы информатики и вычислительной техники» [11] две главы были посвящены описанию конкретных приемов программирования на двух языках – простом и очень распространенном, но методически порочном беспроцедурном языке Бэйсик и разработанном в сибирской школе учебно-ориентированном языке программирования Рапира (Расширенный Адаптированный Поплан-Интерпретатор, Редактор, Архив) [5].

К горячему лету 1985 года, которое стало насыщенным этапом переподготовки большой армии преподавателей к государственному решению проблемы внедрения отечественной школьной информатики, относится начало фундаментальных разработок на механико-математическом факультете МГУ им. М. В. Ломоносова по учебно-ориентированной системе программирования, которая поначалу называлась Е-практикумом (в память об А. П. Ершове), потом еще более откровенно – Ершолом и, наконец, КуМиром (Комплект Учебных Миров), ставшим в нашей стране одной из самых распространенных учебных языковых систем программирования, предназначенной для поддержки начальных курсов информатики и программирования в средней и высшей школе [8], и последовавшей за ней серьезной методической поддержки [7]. Сейчас КуМир продолжает свою активную жизнь в ву-

зовских аудиториях и школьных компьютерных классах, обрстая год от года новыми возможностями, основанными на самых современных достижениях программистского языкотворчества [9] и тенденциях понижения минимального возрастного порога обучаемых [10].

Характерной особенностью творческого коллектива разработчиков КуМира является постоянная поддержка своей оригинальной разработки на всех отечественных платформах компьютеров. Эта принятая в авторитетнейших мировых творческих программистских фирмах черта особенно актуальна в наших российских условиях и, наряду с обновлениями сопровождающей методической системы, является одним из главных следствий долгой и успешной жизни КуМира.

К началу XXI века актуальность непрерывного школьного курса информатики как ответа школы на социальное требование информационного общества – формирование операционного стиля мышления молодого поколения, выходящего из школы в большую жизнь в эпоху информационного общества, – была осознана до уровня постановки проблемы стандартизации и структурирования информатического школьного образования. Характерно, что понимание созревшей до уровня федерального стандарта образования концепции непрерывного школьного курса информатики стало основным дидактическим инструментом борьбы против специалистов, которые воспользовались словесными конструкциями о потребности стандартизации образования, для того чтобы выхолостить программирование из содержания школьного образования, сводя его к технологическим приемам в информационных технологиях. Потребовались настойчивые усилия наших ведущих математиков, информатиков и программистов – А. Л. Семенова, Е. К. Хеннера, И. Г. Семакина, Н. В. Макаровой, А. Н. Терехова, А. Г. Кушниренко и многих других, – опиравшихся на ключевые установки дидактических обоснований А. П. Ершова в области информатизации общества и, в частности, образования [9], чтобы устоять против набирающего силу нового стереотипа, отрицающего место программирования в школьном курсе информатики. Все они видели в программировании концентрированную сущность информатики, призванной описывать разнообразные процессы интеллектуальной деятельности ограниченным набором формальных лексических средств.

Дидактическая важность основных фундаментальных понятий программирования и информатики, составляющих мировоззренческий, алгоритмический и информационно-технологический потенциал человека, требует, чтобы они (эти понятия) были сформированы на том этапе человеческой жизни, который покрывается конституционно обязательным для всех людей периодом. Этот единый для всех период – школьная жизнь. В [4] приводится схема содержания (табл. 1) такого непрерывного курса информатики (вообще говоря, динамическая):

Таблица 1

Совокупность фундаментальных навыков, знаний, умений, понятий и представлений, необходимых для формирования операционного стиля мышления	Совокупность прикладных навыков, необходимых для применения идей и методов информатики в других отраслях человеческой деятельности	Система основных положений информатики как науки в соответствии с ее местом в современной системе научных знаний	Комплекс знаний, необходимых для общей ориентации в возможностях современной и перспективной техники и прикладных систем информатики
модуль 1	модуль 2	модуль 3	модуль 4
1–5 классы	3–8 классы	9 классы	10–11 классы

Поскольку роль программирования в школе непосредственно опирается на непрерывность школьного курса информатики, становится принципиально важным разделить содержание обучения элементам программирования по тем разделам, которые определены государственным образовательным стандартом как пропедевтический курс (модули 1 и 2), базовый курс (модули 2 и 3) и предпрофессиональное обучение (модуль 4).

Наполнение этой четырехмодульной схемы содержанию обучению информатике может, вообще говоря, быть различным в зависимости от принятой в учебном заведении языковой системы программирования. Здесь оно в значительной степени привязано к хорошо себя зарекомендовавшей в отечественной школе программно-методической системе КуМир [7, 9].

В пропедевтическом курсе – в начальной школе – непрерывный курс информатики составляется учебной средой ПиктоМир и открытой серией встроенных в КуМир программных исполнителей (Робот, Чертежник и др.). Начало пропедевтического курса представляет собой освоение элементарных линейных программ с помощью пиктографического языка. Управление программными исполнителями осуществляется сначала при помощи кнопочных (обобщенных) пультов, а затем – посредством пиктограмм. В учебной среде ПиктоМир (<http://www.piktomir.ru/>) дети получили возможность обучаться программированию: создавать программы, не опираясь на навыки работы с текстами. Набор пиктограмм здесь заменил текстовые команды. С их помощью дети могут «собирать» на экране компьютера несложную программу, управляющую виртуальными роботами. ПиктоМир позволяет педагогу постепенно вводить такие важнейшие понятия программирования, как циклы, подпрограммы (процедуры) и условные операторы. Однако возникает вопрос о том, насколько эти базовые понятия программирования доступны детям дошкольного и младшего школьного возраста, не умеющим или не слишком любящим читать и писать. Этот вопрос был снят годами результативных педагогических экспериментов по обучению старших дошкольников (5- и 6-летних детей) азам программирования в среде ПиктоМир [10].

Центральной проблемой для них оказалось освоение понятия *программного управления*. Этап составления программы детям был понятен. А вот процесс пошагового выполнения программы вызывал у дошкольников затруднения. Дело в том, что процесс выполнения каждой команды линейной программы имеет две стадии, две грани:

- 1) работа с исполнителем – выдача роботу очередной команды и
- 2) работа с программой – мысленный перевод этой команды из разряда еще не выполненных в уже исполненные.

В то время как первая стадия наглядна и при компьютерном представлении, и в игровом режиме (когда роль робота играет один из детей), вторая стадия – работа с программой – визуализируется недостаточно четко. Здесь программному дидактическому инструменту на помощь приходит методика, которая, кстати, практически убеждает, что этот важнейший этап освоения воображаемого управления роботом в пропедевтическом курсе наиболее эффективно выполняется не столько профессионалом-информатиком или программистом, сколько педагогом, получившим знания по возрастной психологии и методике обучения в начальной школе. Здесь возникает серьезная, до сих пор до конца не решенная ни в политических, ни в экономических аспектах проблема – коренная конструктивная модернизация начального и дошкольного образования, построенная на их информатизации (в данном случае эти важные аспекты осознанно «вынесены за скобки»). Поэтому, например, приведенная выше общая схема содержания информатического образования сегодня вынужденно ограничена потребностями школы с 1-го класса; эта схема предусматривает информатизацию образования на этапе обязательного школьного образования.

Пиктографическое (бестекстовое) программирование появилось не с КуМиром и его представителем в пропедевтическом курсе – ПиктоМиром – и не только им продолжается. Конструирование бестекстовых программ было известно и в ПервоЛого [13], и в Скретче [3], и в «Азбуке Роботландии» [2], и в работе [6] греческих исследователей, количественно измеривших затруднения детей в детском саду при воображении реальных действий в ходе записи еще не выполненных действий, и в других программных учебных системах. Однако у ПиктоМира есть одно важное отличие, выделяющее его в сравнении с [13] и [3]: он не является автономной системой программирования, а (подобно исполнителям «Роботландии» и роботам «Школьницы» [5]) представляет собой часть принципиально более широкой системы программ, реализующих методический последовательный переход от ручного (или кнопочного) управления исполнителями через язык управления ими к учебному, а затем и учебно-производственному языку программирования.

В системе виртуальных роботов существуют, как правило, несколько постепенно усложняющихся роботов (в КуМире – это ПиктоМир, Робот, Чертежник, РобоТор и др.). Каждого из них младший школьник осваивает отдельно, автономно. Однако достаточно обычной наблюдательности ученика (впрочем, методически компетентному учителю нетрудно стимулировать такую наблюдательность), чтобы обнаружить общие, единые элементы управления, составляющие инвариантную программную базу всех виртуальных роботов одного семейства. Элементами такой базы являются описания алгоритмов со структурными отступами, операторы простых циклов ( $n$  раз), начальные и конечные условия в работе алгоритма. Из таких инвариантов виртуальных роботов формируется язык управления

программными роботами, освоив который, ученик всегда одинаково записывает операторами языка одинаковые для всех исполнителей действия, отвлекаясь на специфику каждого виртуального робота только тогда, когда при программировании приходится обращаться к уникальным особенностям того или иного робота.

Переход от языка управления роботами к учебно-ориентированному языку программирования высокого уровня стимулирован расширением множества задач, предлагаемых для решения. Такой новый для школьника этап позволяет поднять учащегося над уровнем специфики игровых, модельных и вычислительных задач с помощью абстрактных средств языка программирования. Этот момент методика школьного курса информатики считает наиболее подходящим для математизации информатики: ученики знакомятся с понятием переменной, оператором присваивания, вычислением выражений. Заметим, что в это время школьник еще не перешел от пропедевтической части непрерывного курса к базовой, но уже подготовлен к пониманию функциональной сущности языка программирования и многообразию таких языков.

В этом отношении оказался чрезвычайно интересным методический опыт организации курса (и организации производственной программистской практики в нем) в известном новосибирском центре – Районной школе юных программистов, организованной Г. А. Звенигородским с одобрения и при активной поддержке академика А. П. Ершова. В это внешкольное учебное заведение дополнительного образования школьники приходили, в большинстве своем еще не перейдя в среднюю школу из начальной. Первые два года обучения были наполнены знакомством с Робиком – простым языком управления исполнителями – и вслед за ним с Рапирой – учебно-ориентированным высокоуровневым языком программирования. После окончания двухлетнего обучения дети направлялись на летнюю производственную практику в многочисленные научно-исследовательские институты новосибирского Академгородка для самостоятельного освоения разнообразных «производственных» языков. Для тех, кто приходил в конструкторское бюро, освоение незнакомого еще Бейсика становилось упражнением на пару часов в первое утро практики. Школьники, практиковавшиеся в Институте ядерной физики, получали самостоятельное упражнение на пару вечеров – освоение почитаемого физиками-ядерщиками Фортрана. А молодые люди, чья практика оказалась связанной с Институтом экономики Сибирского отделения АН СССР, на пару недель были увлечены весьма трудоемкими, но распространенными среди экономистов языками PL/I и Кобол.

У московских школьников, чье информатическое образование проходило на базовом уровне под руководством механико-математического факультета Московского университета им. М. В. Ломоносова, мировоззрение складывалось под непосредственным влиянием КуМира и системы программирования на школьном алгоритмическом языке. Именно на уровне базового курса молодые люди усваивают многообразие языков программирования как естественную и объективную картину современного мира.

В этой части курса школьник сегодня с большим интересом создает программы для управления широким набором аппаратных и программных (роботы LEGO NXT) исполнителей.

Оценивая новый уровень знаний своих учеников, смело бросающихся в широкое море языков программирования, компетентный учитель уже может (и должен!) показать философскую границу в программировании, которая разделяет беспроцедурные языки (Бейсик) и процедурные (Паскаль, КуМир и др.): процедурные языки моделируют естественную склонность человека к обобщениям и универсальности, тогда как беспроцедурные языки выглядят, скорее, как инструменты кустарного производства [12]. Осознание опасности меточной структуры бейсиковых программ, породившей его порочные блок-схемы, становится рубежным и важным фактом для школьника, который планирует в будущем посвятить себя педагогическому компьютерному языкотворчеству. При понимании такой угрозы процедурный и структурированный КуМир становится школой будущего программиста.

Подготовленные для перехода на уровень предпрофессионального информатического образования подростки понимают, что на любом языке программирования, каждый из которых по-своему универсален и, следовательно, может стать инструментом реализации любой написанной на нем программы, можно описать необходимую для решения на компьютере задачу (если компьютер оснащен транслятором с такого языка). Поэтому такими молодыми людьми программист теперь уже воспринимается не как человек, сумевший написать компьютерную программу на том или ином языке программирования, а, прежде всего, как специалист, сумевший обоснованно выбрать в качестве инструмента ре-

шения именно такой язык, который наиболее адекватен требованиям соответствующей предметной области.

В последнем из уровней школьного образования – предпрофессиональном обучении – существенно заметен уклон учебного плана в сторону дисциплин, по которым происходит профессиональная специализация на разных профилях. Это отчетливо видно по количеству часов в схеме организации образовательных курсов на базовом и профильном уровнях (табл. 2).

Как пропедевтический курс готовит учащихся к следующему за ним курсу базовому, обязательному для завершения основной средней школы, так и предпрофессиональное обучение становится основой специализации в выбранной профессии, в которой ученик будет совершенствовать свои компетенции после получения полного среднего образования в высшем учебном заведении.

Таблица 2

		Элективные курсы				
		Образовательный курс				Другие профили
10–11 кл.	Базовый уровень (70 ч.)			Профильный уровень (280 ч)		
	Социально-экономический	Индустриально-технологический	Универсальный	Физико-математический	Информационно-технологический	
8–9 кл.	Базовый курс (35+70 ч.)					

С точки зрения информатики именно здесь на профильном уровне выпускного класса полной средней школы и начинается формирование программиста современного уровня – происходит знакомство школьников с новыми языковыми парадигмами, с основными понятиями и конструкциями объектно-ориентированных языков программирования (Питон, Си++, Си# и др.).

Не рассматривая здесь и сейчас этот отдельный пласт сегодняшнего программирования, следует, однако, констатировать, что долгий и непрерывный путь от полуигрушечного ПиктоМира до нынешних титанов – языков ООП – оказался возможным благодаря плодотворной концепции учебных языков программирования.

#### Библиографический список

1. Винер, Н. Кибернетика, или Управление и связь в животном и машине [Текст] / Н. Винер ; пер. с англ. И. В. Соловьева и Г. Н. Поварова ; под ред. Г. Н. Поварова. – 2-е издание. – М. : Наука ; Главная редакция изданий для зарубежных стран, 1983. – 344 с.
2. Дуванов, А. А. Азбука Роботландии. Информация [Текст] / А. А. Дуванов [и др.] // Информатика, ИД Первое сентября. – 2012. – № 7. – С. 36–49, электронное приложение на CD.
3. Еремин, Е. А. Среда Scratch – первое знакомство. «Информатика» [Текст] / Е. А. Еремин // ИМД 1 сентября. – 2008. – № 18 – С. 17–24, № 20 – С. 16–28.
4. Ершов, А. П. Школьная информатика (концепции, состояние, перспективы) [Текст] / А. П. Ершов, Г. А. Звенигородский, Ю. А. Первин. – Препринт ВЦ СО АН СССР, №152, Новосибирск, 1979. – 26 с.
5. Звенигородский, Г. А. Основные операторы учебно-производственно-го языка Рапира [Текст] / Г. А. Звенигородский // Квант. – 1980. – № 1. – С. 52–55.
6. Комис, В., Мисирли, А. Педагогическая робототехника и предварительные концепции программирования в детском саду : исследование на примере программируемой игрушки Пчелка [Текст] / В. Комис, А. Мисирли // Труды Большого Московского семинара по методике раннего обучения информатике. – Т. 3. – М. : Изд-во РГСУ – 2012.

7. Кушниренко, А. Г., Лебедев, Г. В. 12 лекций о том, для чего нужен школьный курс информатики и как его нужно преподавать [Текст] / А. Г. Кушниренко, Г. В. Лебедев. – М. : Лаборатория Базовых Знаний, 2000.
8. Кушниренко, А. Г. Информатика 7–9 классы [Текст] : учебник для общеобразовательных учебных заведений / А. Г. Кушниренко, Г. В. Лебедев, Я. Н. Зайдельман. – М. : Дрофа, 2002 – 336 с.
9. Кушниренко, А. Г., Леонов, А. Г. Кумир вернулся! [Текст] / А. Г. Кушниренко, А. Г. Лебедев // ИД 1 сентября. Информатика. – № 6. – 2009.
10. Кушниренко, А. Г. ПиктоМир : пропедевтика алгоритмического языка (опыт обучения программированию старших дошкольников) [Текст] / А. Г. Кушниренко, И. Б. Рогожкина, А. Г. Леонов // Труды Большого Московского семинара по методике раннего обучения информатике. – Т. 3. – М. : РГСУ, 2012. – 322 с.
11. Основы информатики и вычислительной техники. [Текст] : учебное пособие. В 2-х ч. Ч. 2 / А. П. Ершов [и др.]. – М. : Просвещение, 1986. – 143 с.
12. Первин, Ю. А. Методика раннего обучения информатике [Текст] / Ю. А. Первин. – М. : БИНОМ, 2-е издание, 2008. – 282 с.
13. Яковлева, Е. И. ЛогоМозаика [Текст] : сборник проектов / Е. И. Яковлева. – М. : Институт новых технологий. – 75 с.

### Bibliograficheskiy spisok

1. Viner, N. Kibernetika, ili Upravleniye i svyaz' v zhivotnom i mashine [Tekst] / N. Viner ; per. s angl. I. V. Solov'yeva i G. N. Povarova ; pod red. G. N. Povarova. – 2-ye izdaniye. – M. : Nauka ; Glavnaya redaktsiya izdaniy dlya zarubezhny'h stran, 1983. – 344 s.
2. Duvanov, A. A. Azbuka Robotlandii. Informatsiya [Tekst] / A. A. Duvanov [i dr.] // Informatika, ID Pervoye sentyabrya. – 2012. – № 7. – S. 36–49, elektronnoye prilozheniye na CD.
3. Yeremin, Ye. A. Sreda Scratch – pervoye znakomstvo. «Informatika» [Tekst] / Ye. A. Yeremin // IMD 1 sentyabrya. – 2008. – № 18 – S. 17–24, № 20 – S. 16–28.
4. Yershov, A. P. Shkol'naya informatika (kontseptsii, sostoyaniye, perspektivy) [Tekst] / A. P. Yershov, G. A. Zvenigorodskiy, Yu. A. Pervin. – Preprint VTS SO AN SSSR, №152, Novosibirsk, 1979. – 26 s.
5. Zvenigorodskiy, G. A. Osnovny'ye operatory' uchebno-proizvodstvennogo yazy'ka Rapira [Tekst] / G. A. Zvenigorodskiy // Kvant. – 1980. – № 1. – S. 52–55.
6. Komis, V., Misirli, A. Pedagogicheskaya robototekhnika i predvaritel'ny'ye kontseptsii programmirovaniya v det'skom sade : issledovaniye na primere programmirovaniya igrushki Pchelka [Tekst] / V. Komis, A. Misirli // Trudy Bol'shogo Moskovskogo seminaru po metodike rannego obucheniya informatike. – T. 3. – M. : Izd-vo RGSU – 2012.
7. Kushnirenko, A. G., Lebedev, G. V. 12 lektsey o tom, dlya chego nuzhen shkol'ny'y kurs informatiki i kak yego nuzhno prepodavat' [Tekst] / A. G. Kushnirenko, G. V. Lebedev. – M. : Laboratoriya Bazovykh Znaniy, 2000.
8. Kushnirenko, A. G. Informatika 7–9 klassy' [Tekst] : uchebnyy dlya obshcheobrazovatel'ny'h uchebny'h zavedeniy / A. G. Kushnirenko, G. V. Lebedev, YA. N. Zaydel'man. – M. : Drofa, 2002 – 336 s.
9. Kushnirenko, A. G., Leonov, A. G. Kumor vernul'sya! [Tekst] / A. G. Kushnirenko, A. G. Lebedev // ID 1 sentyabrya. Informatika. – № 6. – 2009.
10. Kushnirenko, A. G. PictoMir : propedevtika algoritmicheskogo yazyka (opyt obucheniya programmirovaniyu starshih doshkol'nikov) [Tekst] / A. G. Kushnirenko, I. B. Rogozhkina, A. G. Leonov // Trudy Bol'shogo Moskovskogo seminaru po metodike rannego obucheniya informatike. – T. 3. – M. : RGSU, 2012. – 322 s.
11. Osnovy' informatiki i vychislitel'noy tekhniki. [Tekst] : uchebnoye posobiye. V 2-h ch. CH. 2 / A. P. Yershov [i dr.]. – M. : Prosveshcheniye, 1986. – 143 s.
12. Pervin, Yu. A. Metodika rannego obucheniya informatike [Tekst] / Yu. A. Pervin. – M. : BINOM, 2-ye izdaniye, 2008. – 282 s.
13. Yakovleva, Ye. I. LogoMozaika [Tekst] : sbornik proyektov / Ye. I. Yakovleva. – M. : Institut novy'h tekhnologiy. – 75 s.