

Алгоритмизация и программирование

Представленный здесь текст был опубликован в 2004 г. в газете «Информатика» в рамках дискуссии под общим названием «Вопросы на понимание». Дискуссия была посвящена содержанию проекта Примерной программы по информатике. Надо сказать, что обсуждение оказалось не напрасным и в какой-то степени повлияло на окончательное содержание Примерной программы

В данном выпуске проекта «Вопросы на понимание» продолжу «петь свою песню» в контрапункте нашей дискуссии. Тема: становление информатики как полноценного общеобразовательного предмета в школе. На этот раз на примере содержательной линии «Алгоритмизация и программирование». Тематика очень благодатная, т.к. она позволяет наиболее наглядно проследить эволюцию школьной информатики в концептуальном, содержательном и методическом плане.

Место и содержание алгоритмизации в школьном курсе информатики – тема насколько традиционная, настолько и спорная. Утверждение авторов программы о том, что алгоритм – центральное понятие современной информатики, вызывает, мягко говоря, недоумение. От него веет архаикой 80-х годов. Вспоминаются тезисы: «Программирование – вторая грамотность», «Развитие алгоритмического мышления – главная задача школьной информатики».

Безусловно, за прошедшие 20 лет акценты сместились как в предметной, так и в образовательной области информатики. Безусловно и то, что место для алгоритмизации в общеобразовательном курсе информатики есть, но это место уже не председательское.

Изучение в школе любого общеобразовательного предмета преследует три основные цели:

- вносить вклад в научное мировоззрение учащихся (теоретико-мировоззренческая цель);
- развивать определенные формы мышления (развивающая цель);
- передавать учащимся определенные практические умения и навыки, полезные в жизни, в учебе и в будущей профессиональной деятельности (прагматическая цель).

Это применимо и к информатике, если ее трактовать как общеобразовательную дисциплину.¹ Содержательная линия «Алгоритмизация и программирование» имеет отношение ко всем трем целям. В связи с этим следует рассматривать несколько аспектов изучения данной линии: теоретический, развивающий и прагматический.

Чтобы разобраться в этих вопросах, разделим содержательную линию на два предмета обучения: обучение алгоритмизации и обучение программированию на ЭВМ (языкам программирования).

Начнем с *развивающего аспекта* обучения алгоритмизации. Хорошо известно, что развитие алгоритмического (процедурного) мышления учащихся происходит тем эффективнее, чем раньше оно начинается. Методика и средства пропедевтического обучения

¹ Основные компоненты содержания информатики в общеобразовательных учреждениях. Приложение 2 к решению Коллегии Минобразования РФ от 22.02.95 N 4/1. // Информатика и образование. 1995, №4, С.17 – 36

Проект федерального компонента государственного образовательного стандарта начального общего, основного общего и среднего (полного) образования. Образовательная область «Информатика». Авторский коллектив под рук. А.А.Кузнецова. // Информатика и образование. 1997. № 1. С.3-11.

алгоритмизации широко известны и хорошо отработаны. Если же, согласно базисному учебному плану, изучение информатики должно начинаться с 8 класса, то освоение алгоритмизации учениками будет происходить несколько более затруднительно. Но это не означает, что от него надо отказываться, а его развивающая функция уже не актуальна.

Цель обучения алгоритмизации заключается в овладении учащимися структурной методикой построения алгоритмов. Это значит, ученики должны научиться использовать в практике построения алгоритмов основные управляющие структуры: следование, ветвление, цикл; уметь разбивать задачу на подзадачи, применять метод последовательной детализации алгоритма. Дидактические средства для этого хорошо отработаны - это разнообразные учебные исполнители алгоритмов: черепахи, роботы, чертежники, кенгуренки и пр. Известна методическая идея, идущая еще от А.П.Ершова: исполнители алгоритмов делятся на исполнителей, работающих «в обстановке» и исполнителей, работающих с величинами. Перечисленные выше исполнители относятся к первой группе.

Использование таких исполнителей с методической точки зрения очень эффективно. Основные достоинства – понятность решаемых задач, наглядность работы исполнителя, поддержка структурной методики алгоритмизации. Если кто-то думает, что это лишь игрушки для младших школьников, то он ошибается. Из личного преподавательского опыта знаю, с каким интересом ученики 8-х и даже 10-х классов решают алгоритмические задачи для исполнителей. Уровни сложности задач можно варьировать от тривиальных до головоломных, адаптируя к способностям учеников. *Задача развития структурного алгоритмического мышления учащихся решается в полной мере на учебных исполнителях, работающих «в обстановке».*

При изучении алгоритмизации в пропедевтическом курсе развивающий аспект является основным. Однако в базовом курсе информатики к нему добавляются еще новые аспекты, которые следует отнести к теоретическим целям школьной информатики. Таких аспектов два. Первый – *кибернетический аспект*. Речь идет о знакомстве с *информационными основами процессов управления*. Место алгоритмов в этой теме определяется следующим тезисом: *алгоритм управления – это информационная составляющая всякой системы управления*. В процессе управления происходит передача данных о состоянии управляемого объекта по линии обратной связи, а по линии прямой связи – управляющая информация, т.е. команды управления. Последовательность команд управления и составляет алгоритм управления. Его должен «знать» управляющий объект.

Учебные исполнители алгоритмов являются прекрасными моделями процессов управления. На них, в частности, хорошо иллюстрируется тот факт, что без обратной связи алгоритм управления может быть только линейным, а при наличии обратной связи может содержать ветвления и циклы. Например, в исполнителе «Кенгуренок» управляющий изображен мальчиком Кристоффером, который управляет кенгуренком Ру. При проверке условий Кристоффер задает Ру вопрос и получает от него ответ. В зависимости от ответа выдается последующая команда. После этого любой ученик поймет, что такое обратная связь.

Второй аспект заключается в связи линии алгоритмизации и программирования с *линией компьютера*, с более глубоким раскрытием понятия программного управления ЭВМ. Ученики должны получить ответы на вопрос: что такое программа для ЭВМ? Как ЭВМ управляет «сама собой»? Почему ЭВМ можно назвать самоуправляемой системой?²

Изучение алгоритмизации *в программистском аспекте* связано с введением новой группы понятий (в дополнение к алгоритмическим структурам): понятия величины, типа и структуры величины, константы и переменной, присваивания значения переменной, действия (операции) над величинами, выражения (арифметические, логические, строковые). Если до изучения этой темы ученики работали с базами данных и электронными таблицами

² Не имея возможности подробно здесь обсуждать эти вопросы, сошлюсь на книгу «Лапчик М.П., Семакин И.Г., Хеннер Е.К. Методика преподавания информатики. Учеб пособие для студ. пед. вузов. Под общей редакцией М.П.Лапчика. – М.: Издательский центр «Академия», 2001. – 624 с»

ми, то представление о величинах и их свойствах у них уже имеется. От этих представлений можно оттолкнуться, вводя понятие величины в языках программирования.

При наличии небольшого объема учебного времени, *программирование в базовом курсе может изучаться лишь на уровне введения*. Основная задача ограничивается рамками все той же линии компьютера: раскрывается понятие программного управления работой компьютера. Изучение происходит на примерах простых программ на Паскале (предпочтительный вариант) или на структурированной версии Бейсика. Показывается, как организуется простейший диалог компьютера с человеком: компьютер спрашивает, ученик отвечает, компьютер реагирует на ответ в соответствии с его содержанием. Показывается, как организуются простейшие вычисления, например, вводится числовая последовательность, выводится ее среднее арифметическое значение; или вводятся два числа, выводится их наибольший общий делитель (алгоритм Евклида) и т.п. И все! Этого вполне достаточно с точки зрения поставленной цели.

Изучение программирования – как прагматическая цель заключается в освоении основ профессионального программирования. Сегодня программирование на любительском уровне с практической точки зрения не представляет интереса. Используя прикладные программы можно сделать гораздо больше, чем с помощью языков программирования на ученическом уровне. Такую цель можно ставить только перед профильным или элективным курсом информатики.

Массовая практика обучения информатике в основной школе показала реальность следующего распределения учебного времени по темам содержательной линии «Алгоритмизация и программирование»:

№ урока	Содержание	Форма занятия
1	Кибернетика. Кибернетическая модель управления. Место алгоритма в системе управления. Системы управления с использованием компьютеров. Что такое программное управление.	теория
2 – 3	Определение и свойства алгоритма. Линейные алгоритмы. Использование алгоритмического языка и блок-схем для записи алгоритмов. Знакомство с учебным исполнителем алгоритмов. Разработка линейных алгоритмов для учебного исполнителя	Теория + практика
4 – 5	Вспомогательные алгоритмы. Последовательная детализация (на примерах линейных задач). Программирование для учебного исполнителя с использованием вспомогательных алгоритмов, последовательной детализации (линейные задачи)	Теория + практика
6 – 7	Команда цикла. Циклические алгоритмы. Программирование циклических алгоритмов для учебного исполнителя.	Теория + практика
8 – 10	Ветвления. Циклы в сочетании с ветвлениями. Программирование для учебного исполнителя ветвящихся алгоритмов и алгоритмов, сочетающих циклы, ветвления, вспомогательные алгоритмы.	Теория + практика
11	Зачетное практическое занятие по алгоритмизации	Контроль
12	Тестирование по алгоритмизации (при использовании модульной технологии обучения)	Контроль
13	Понятие языка программирования, системы программирования. Знакомство с простейшим примером программы на Паскале (или Бейсике)	Теория + практическая демонстрация программы
14 – 15	Алгоритмы работы с величинами. Понятия: величина, константа, переменная, присваивание, ввод, вывод. Линейные вычисли-	Теория + практика

	тельные алгоритмы (на блок-схемах и на учебном алгоритмическом языке)	
16 – 17	Язык программирования Паскаль: запись числовых констант, описание переменных, операторы присваивания, ввода, вывода. Составление линейных программ на Паскале и их исполнение на компьютере	Теория + практика
18	Ветвления. Представление ветвящихся вычислительных алгоритмов на блок-схемах и учебном алгоритмическом языке (задачи типа определения наибольшего значения, простого диалога и пр.). Программирование ветвлений на Паскале.	Теория + практика
19	Цикл. Представление циклических вычислительных алгоритмов на блок-схемах и учебном алгоритмическом языке (задачи типа вычисления суммы или произведения числовой последовательности). Программирование циклов на Паскале	Теория + практика
20	Сочетание циклов и ветвлений (задачи типа алгоритма Евклида). Программирование на Паскале.	Теория + практика
21	Зачетное практическое занятие по программированию	Контроль
22	Тестирование по программированию (при использовании модульной технологии обучения)	контроль

Даже в таком объеме не удастся уложиться в 19 учебных часов. Любой практикующий учитель поймет, что и этот план очень напряженный и не прост в реализации (особенно в режиме 1 урок в неделю). А уж где там до массивов, списков, деревьев, графов! Неужели авторы всерьез предлагают освоение программирования динамических структур в базовом курсе?

Теперь об олимпиадах по информатике. Эта тема достаточно подробно освещена в цикле лекций профессора С.В.Русакова «Олимпиады по базовому курсу информатики», опубликованном в 2003 г. в газете «Информатика» (нечетные номера с №33 по №47). В городе Перми олимпиада по базовому курсу информатики проводится ежегодно, начиная с 1995 г. Соревнуются команды школ (по 3 человека), тренерами которых выступают школьные учителя информатики. Соревнования проводятся отдельно для учеников 8 и 9 классов. По итогам олимпиады награждаются участники команды и тренеры-учителя. Такая форма реально демонстрирует уровень обучения информатике в конкретной школе, стимулирует его повышение. Содержание олимпиады полностью согласуется с содержанием базового курса, изучаемого в школах.

Начиная с 2001 года, олимпиада по базовому курсу проводится на областном уровне. Она идет параллельно с федеральной олимпиадой по программированию, в которой участвуют ученики старших классов. Это соревнование, как и по программированию, проходит в личном зачете. В отличие от программистов, для которых есть перспектива выхода на федеральный и международный туры, для участников олимпиады по базовому курсу соревнования заканчиваются на областном туре.

Организаторы пермской олимпиады готовы выступить инициаторами федерального тура олимпиады по базовому курсу. Разумно проводить этот этап соревнования в дистанционном режиме. У нас есть программное обеспечение для поддержки такого режима олимпиады. Можем предоставить свой сервер. Данное ПО позволяет проводить соревнование и его судейство как в режиме off-line, так и в on-line. Хотелось бы надеяться, что эту идею когда-нибудь поддержит Минобразования РФ. Организаторами в разных регионах РФ могли бы выступить учителя, получившие сертификаты по итогам заочного обучения, проведенного проф. С.В.Русаковым через газету Информатика (около 60 человек).