

Заверен
Директор



/Р.М. Тасанова/

ДОКЛАД

на тему:

«Задания по алгоритмике и программированию в ГИА

Подготовила: Загирбекова П.К.

Информатика в школьном курсе дает несколько особых знаний и умений, без которых невозможно ни быть успешным на рынке труда сегодня, ни получить образование, которое позволит остаться успешным завтра. Одно из самых важных человеческих умений – это умение составить, а затем и претворить в жизнь план некой будущей деятельности. Заглянув в Энциклопедический словарь, можно обнаружить: такой план называется программой. Решение задач по физике и информатике в реализации задач на программирование можно связать, да и нужно связать. Привычка тратить время и силы на обдумывание, запись и отработку планов будущей деятельности себя самого, других людей или больших коллективов называется алгоритмическим стилем мышления. Овладеть алгоритмическим стилем мышления непросто. Для этого нужно научиться заранее предсказывать ситуации, которые могут случиться в будущем, и предусматривать в планах правильное поведение в этих ситуациях. С другой стороны, как и другие человеческие навыки, алгоритмический стиль мышления можно развивать и тренировать путем целенаправленно подобранной системы упражнений.

Поэтому большое внимание в преподавании я обращаю на раздел «Алгоритмы. Программирование». За время моей работы учителем информатики и физики актуальность темы «Понятие алгоритма. Программирование» претерпела значительные изменения. Большая часть времени отводится на преподавание тем цикла «Информационные и коммуникационные технологии». Наряду с этим несколько не изменились требования к уровню усвоения знаний и умений этого раздела программы по информатике, так как он остается основой фундаментальных знаний по предмету. Вместе с тем ФИПИ требует знаний учащихся в старом объеме курса информатики, особенно в части знания программирования. Учащиеся для успешной сдачи экзамена должны не только знать основные алгоритмические конструкции и операторы изучаемого языка программирования, но и иметь опыт самостоятельной записи алгоритмов и программ, решения практических задач методом разработки и отладки компьютерной программы. Следует уделять больше внимания формализации записи и исполнения алгоритмов, так как результаты экзамена показывают, что у части учащихся так и не формируется умение формального исполнения алгоритмов. Поэтому для себя я решил, что необходимо изыскать все возможности, чтобы преподавание темы «Понятие алгоритма. Программирование» оставалось на достаточном высоком теоретическом и практическом уровне.

Мной была включена в календарно-тематическое планирование 8 класса тема «Исполнители алгоритмов», а в 9 классе «Программирование». Хорошо известно, что развитие алгоритмического (процедурного) мышления учащихся происходит тем эффективнее, чем раньше оно начинается. Дидактические средства для этого хорошо отработаны - это разнообразные учебные исполнители алгоритмов: черепахи, роботы, чертежники. За основу я взяла учебник Угриновича «Информатика 8-9 класс, а так же Босова Л.Л. Дети разрабатывают алгоритмы и пишут программы для исполнителя стрелки, знакомятся с понятием алгоритма и основными алгоритмическими конструкциями, а также с процедурами и подпрограммами. В ходе уроков используются игровые моменты, при написании программы все действия исполнителя отображаются на экране, причем при выполнении программы используется анимация. 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».

Обучение школьников алгоритмизации и программированию с методической точки зрения является одной из самых трудных задач. Составление программ – весьма сложный процесс, включающий в себя значительное число качественно разнообразных этапов. Наиболее сложные из них – постановка задачи и ее алгоритмизация. Именно этим этапам уделяю основное внимание при разработке методик по преподаванию разделов алгоритмизации и программирования. Как правило, в учебных пособиях при объяснении данных разделов приводятся конкретные задачи различных типов и соответствующие им алгоритмы с некоторыми частными пояснениями. В основном рассматриваются примеры линейного, разветвляющегося и циклического типов, а также алгоритмов с «вложенными циклами». Если дается решение более сложных задач, то приводятся готовые алгоритмы решения без их описания процесса построения алгоритмов. Считается, что, изучив приведенные примеры,

учащийся сможет составить алгоритм любой задачи. Однако это далеко не так.

При обучении алгоритмизации необходимо использовать унифицированную и формализованную процедуру перехода от словесно-формульного описания метода решения задачи к схеме алгоритма этой задачи, причем такой схеме, которая может быть формально перекодирована в программу на алгоритмическом языке. Эта процедура позволяет строго логически выводить формулы и условия, составляющие «начинку» алгоритма (программы). Особенность такого подхода еще и в том, что он требует от учащегося подробного описания процесса вывода алгоритма, поэтому процесс обучения алгоритмизации становится хорошо наблюдаемым для учителя, а значит, и хорошо управляемым. В качестве основного средства описания алгоритмов я выбираю блок-схемы алгоритмов – наиболее наглядный и понятный, а кроме того, и наиболее естественный для человека способ, т.к. человек мыслит образами и ему легче воспринимать образы, нежели текст.

Данная методика вполне доступна любой категории учащихся на начальной стадии обучения информатике и программированию. Она базируется на известных подходах к построению программ – структурном программировании и проектировании «сверху вниз» и является их развитием. Методика достаточно универсальна.

И в дополнение, как обязательный элемент обучения алгоритмизации должна входить отладка алгоритма каждого вида. Она позволяет учащимся «прочувствовать» суть каждого вида алгоритмов, предлагает им правила самоконтроля – правила нахождения ответов к задачам алгоритмизации, позволяет понять и закрепить принципы отладки программ. Как показывает практика, использование в обучении информатике методики формализованного перехода от схемы алгоритма к программе на алгоритмическом языке позволяет научить основную массу учащихся составлять алгоритмы и программы достаточно сложных задач в отличие от традиционного подхода, при котором можно говорить лишь о знакомстве с понятиями «алгоритм», «программа» и пр. в процессе изучения этого предмета.

Второй аспект заключается в связи линии алгоритмизации и программирования с линией компьютера, с более глубоким раскрытием понятия программного управления.

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. 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 прагматической цели 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 элективными курсами информатики.

Конечно, очень тяжело уложиться в имеющиеся учебные часы. 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, не говоря уж об освоении программирования динамических структур в базовом курсе.

Работа по развитию логического и алгоритмического мышления проходит не только на уроках, а также и на внеклассных занятиях.

Также, в нашей школе имеется возможность изучать программирование в 10-11 классах на базе Точка Роста. В этом случае можно последовательно продвигаться от одного практического занятия к другому.

Ведь для успешного участия в олимпиаде по программированию школьник должен не только владеть языком программирования, но и уметь придумывать и реализовывать алгоритмы решения задач, оценивать время их работы, тестировать и отлаживать свои программы.