

Е. М. БЕРЕЗОВСКАЯ, М. И. ЖАДАН

Математический факультет,

кафедра вычислительной математики и программирования

## **ЧАСТНОПРЕДМЕТНЫЕ ТЕХНОЛОГИИ В ИНФОРМАТИКЕ**

В настоящее время в педагогический лексикон прочно вошло понятие педагогической технологии. Попытки внести технологию в учебный процесс не прекращались все наше столетие. Приблизительно до середины 50-х гг. они были связаны с созданием некоей технической среды, комплекса автоматизированных средств для традиционного

обучения. С середины 50-х гг. появился новый технологический подход к построению учебного процесса.

Понятие «педагогическая технология» в образовательной практике употребляется на трех иерархически соподчиненных уровнях:

1) *общепедагогическая технология* характеризует целостный образовательный процесс на определенной ступени обучения.

2) *частнопредметная педагогическая технология* употребляется в значении «частная методика», т. е. как совокупность методов и средств для реализации определенного содержания обучения и воспитания в рамках одного предмета, класса (методика преподавания предметов, методика работы учителя).

3) *локальная технология* представляет собой технологию отдельных частей учебно-воспитательного процесса, решение частных дидактических и воспитательных задач.

Понятие педагогической технологии частнопредметного и локального уровней почти полностью перекрывается понятием методик обучения, разница между ними заключается лишь в расстановке акцентов. В технологиях более представлена процессуальная, количественная и расчетная компоненты, в методиках – целевая, содержательная, качественная и вариативно-ориентировочная стороны. Технология отличается от методик своей воспроизводимостью, устойчивостью результатов, отсутствием многих «если» (если талантливый учитель, если способные дети, хорошие родители...). Смешение технологий и методик приводит к тому, что иногда методики входят в состав технологий, а иногда, наоборот, те или иные технологии – в состав методик обучения. Примером частнопредметной педагогической технологии может служить «Технология обучения математике на основе решения задач» (Р. Г. Хазанкин).

По типу организации и управления познавательной деятельностью В. П. Беспалько предложены следующие виды технологий:

- 1) классическое лекционное обучение;
- 2) обучение с помощью аудиовизуальных технических средств;
- 3) система «консультант»;
- 4) обучение с помощью учебной книги;
- 5) система «малых групп»;
- 6) компьютерное обучение;
- 7) система «репетитор»;
- 8) «программное обучение» по заранее составленной программе.

Технологии формирования и развития навыков алгоритмического мышления являются одной из важнейших задач курса «Информатика». Различные вузовские и школьные предметы используют и развивают

разные аспекты мышления. Математика развивает математический стиль мышления, литература – эмоционально-чувственное, образное мышление, информатика – алгоритмический стиль мышления.

Привить учащимся навыки алгоритмического мышления и развивать их – трудная задача. Трудность ее заключается в специфичности образа мыслительной деятельности, необходимой для составления алгоритма. Это творческая работа, как научная деятельность, и не каждому под силу ее быстро освоить. Поэтому важно начинать изучение информатики с младших классов, приучая детей к алгоритмическому мышлению не только на уроках информатики. Наблюдения показывают, что почти все ученики и студенты легко решают задачу непосредственного поиска самого длинного слова в предложении: они полностью представляют себе, что и как надо сделать, но не в состоянии записать правильно соответствующий алгоритм, пусть даже и неформально.

В преподавательской практике стараемся донести специфичность алгоритмического стиля мышления: ведь учителю следует с пониманием относиться к тем учащимся, которые не могут самостоятельно написать, казалось бы, совсем простые алгоритмы. Если на начальной стадии изучения основ алгоритмизации после разбора, например, задачи на составление алгоритма вычисления суммы конечного ряда чисел, ученик не справляется с алгоритмом вычисления их произведения, то это не удивительно: переход к нему не прост. С таким заданием самостоятельно могут справиться ученики, имеющие алгоритмическое мышление.

Технология формирования навыков алгоритмического мышления не проста. Один из ее вариантов – постоянная умственная работа. Ученики должны решить большое количество упражнений, причем эти упражнения необходимо располагать по возрастающей сложности. При этом средства описания алгоритмов должны появляться по мере необходимости. Занятия необходимо строить таким образом, чтобы решение одной задачи указывало путь решения другой, более сложной задачи, чтобы учащиеся, пусть медленно, шаг за шагом, но самостоятельно накапливали опыт алгоритмического мышления. Такой метод предъявления теоретического материала, представляется полезным при изучении алгоритмических и информационных структур, средств их описания.

Существует еще один путь рационального обучения учащихся технике алгоритмизации: систематическое и целенаправленное применение идей структурного подхода. Он полезен и в плане воспитания учащихся, так как развивает умение планировать свои действия при решении сложных задач, способности к общению и коллективной деятельности. Следовало бы его применять с самого начала, уже при знакомстве с линейными алгоритмами, тем более что первый шаг

проектирования алгоритма – разбивка задачи на подзадачи – обычно порождает линейную структуру. К сожалению, предложить такой путь учителям пока невозможно, так как почти отсутствует соответствующая система упражнений. Разработать такую систему – интересная и важная задача для исследователя-методиста.

Хорошим приемом (не любимым учениками и студентами) является метод составления модели, реализующей условие поставленной задачи. Приведем пример поясняющий качество и методы составления алгоритмов, ярко иллюстрирующий методику первоначальной работы за столом, где строится хорошая математическая модель решения задачи.

**Пример.** Сколько существует  $k$ -значных десятичных чисел, удовлетворяющих следующим условиям:

- число содержит только значащие цифры;
- каждая цифра кратна заданному  $n$  ( $n = 1, 2, \dots, 9, 0$ );
- число не является полиндромом.

Время прохождения теста не более 1 секунды.

Последняя фраза ограничивает возможности программиста в выборе алгоритма. Здесь не проходит алгоритм, в котором присутствует необходимое количество вложенных циклов и ветвлений – это слишком расточительно по времени. Если же исследовать задачу, расписав некоторые варианты ручного решения, например: пусть  $n = 1$ , тогда при  $k = 1$  имеем цифры 1, 2, ..., 9, 0, которые являются полиндромом, а значит искомое количество однозначных чисел  $m=0$ ; при  $k = 2$  имеем следующие двузначные числа: 00, 01, ..., 09, 10, 11, 12, ..., 98, 99. Подчеркнутые числа условию задачи не удовлетворяют. Поэтому  $m = (10 - 1) \times (10 - 1) = 9 \times 9$ , то есть 81. Прodelывая аналогичные действия еще с некоторыми  $k$  и  $n$ , приходим к следующей зависимости дающей ответ в поставленной задаче:

$$m = (p - 1)p^{\lfloor \frac{k-1}{2} \rfloor} (p^{\lfloor \frac{k}{2} \rfloor} - 1), \quad \text{где } p = \begin{cases} 10, & n = 1 \\ 5, & n = 2 \\ 4, & n = 3 \\ 3, & n = 4 \\ 2, & n = 5 \dots 9 \end{cases}.$$

Программа, реализующая приведенную формулу, дает оптимальное решение поставленной задачи. В качестве тестов могут быть взяты, например, следующие значения:  $n = 1, k = 3$ , тогда  $m = 810$  и  $n = 4, k = 15 - m = 9\,561\,564$ .

Эффективная форма работы с алгоритмами – их исполнение (тестирование). Наибольший методический эффект, с точки зрения пояснения смысла алгоритма, проверки его правильности и обучения

навыкам алгоритмизации, может быть достигнут тогда, когда в процессе исполнения алгоритма, во-первых, прослеживаются все действия исполнителя и, во-вторых, наглядно и точно «протоколируются» результаты этих действий. Тестирование должно быть полным. Исполнение должно быть не только темой в учебнике, но и инструментом в руках учителя.

Естественным исполнителем компьютерных программ является сам компьютер. Но если после выполнения программы на экране дисплея появляется только числовой результат, то польза от такого исполнения, с точки зрения понимания смысла работы отдельных операторов и динамики выполнения программы, в целом близка к нулю. Большой эффект может быть достигнут при выполнении программ в режиме трассировки. Если такой режим в системе программирования не предусмотрен, необходимо организовывать, хотя бы выводы промежуточных результатов с подробными комментариями. Мы считаем, что при работе с компьютером необходимо практиковать ручное исполнение программ. Опыт показывает, что при сочетании ручного и компьютерного исполнений обучаемые быстрее осваивают язык программирования и овладевают программистскими навыками. В этом плане хорош язык программирования «Pascal ABC». В системе «Pascal ABC» возможно выполнение одной или одновременно нескольких программ и (или) модулей в различных скоростных режимах. Так программу можно: выполнить по командам, выполнить все, выполнить все по командам, выполнить до курсора, выполнить команду, выполнить шаг, выполнить все до курсора. Эта система программирования в настоящее время широко применяется при изучении курсов, связанных и информатикой.