

Попов Константин Алексеевич

канд. физ.-мат. наук, доцент

ФГБОУ ВО «Волгоградский государственный
социально-педагогический университет»

г. Волгоград, Волгоградская область

DOI 10.31483/r-112333

ТЕМАТИЧЕСКИЕ ОСОБЕННОСТИ ОБУЧЕНИЯ ШКОЛЬНИКОВ ПРОГРАММИРОВАНИЮ

***Аннотация:** в статье рассматриваются элементы содержания основного курса информатики и курсов дополнительного обучения школьников в рамках линии «Алгоритмизация и программирование», которые представляются ключевыми для успешной подготовки учащихся к решению задач Единого государственного экзамена.*

***Ключевые слова:** программирование, методика обучения информатике, язык программирования Python, внутрипредметные связи, Единый государственный экзамен.*

Проблема обучения школьников программированию остается актуальной уже несколько десятков лет, пожалуй, с момента введения в школьную программу предмета «Основы информатики и вычислительной техники» еще в советской школе. С тех пор методическое сообщество активно обсуждает подходы к выстраиванию наиболее оптимального процесса обучения. И данные подходы обсуждаются с самых разных сторон. Так, в частности, в работе [1] приводится общий обзор к системам обучения программированию школьников в разных странах, включая и отечественный опыт. Статья [2] посвящена персонализированному (индивидуальному) подходу к обучению с использованием возможностей интеграции основного курса информатики и элективных курсов. Вопросам работы с учащимися разного возраста посвящена статья [3]. И подобных «сторон» в научно-методической литературе мы можем найти довольно много.

В данной статье мы остановимся на особенностях обучения программированию школьников, связанных с последующим прохождением итоговой аттестации в форме Единого государственного экзамена (ЕГЭ).

Система контроля знаний учащихся в форме ЕГЭ постоянно меняется. Уже несколько лет практикуется проведение экзамена по информатике в компьютерной форме без проверки экспертами решений задач на бумаге, что существенно отразилось на содержании задач, и, соответственно на доминирующих методах их решения, которые должны быть максимально доступно изложены учащимся.

Если 5–7 лет назад большую часть задач, входящих в КИМ ЕГЭ по информатике, можно было решить при помощи ручки и листа бумаги, то сейчас при попытке проделать то же самое, во-первых, просто не хватит времени на решение даже тех заданий, в которых жестким условием не прописано написание программного кода. Во-вторых, почти все задачи могут быть решены посредством программирования. Для этого зачастую достаточно заучить клише кода соответствующей задачи.

```
1- for x in range(2):
2-     for y in range(2):
3-         for z in range(2):
4-             for w in range(2):
5-                 f = not (not y or x) or (not z or w) or not z
6-                 if not f:
7-                     print(x, y, z, w, ' ', int(f))
8
```

Рис. 1. Пример листинга задачи №2 ЕГЭ по информатике

Ярким примером подобного клише может служить приведенный выше листинг второй задачи компьютерного ЕГЭ. Здесь вариации подлежат только строки 5 и 6. В строке 5 вводится выражение для функции, указанной в условии задачи, а строка 6 (не всегда необходимая) позволяет выборочно выводить строки значений соответствующие либо нулевым, либо единичным (False/True) значения указанной функции, что облегчает дальнейшую работу с данными

задачи, которые требуется привести в соответствие с заданной таблицей. Все остальное можно оставлять без изменения.

Попытка решения подобной задачи «вручную» приведет к большой потере времени и большой вероятности ошибок, поскольку, как показывает опыт, уже на трех переменных (восемь строк) при построении таблицы истинности ошибаются примерно семь учащихся из десяти. Во второй задаче строк должно быть в два раза больше, что приведет к еще большему количеству ошибок.

Ряд других задач КИМ, позволяющих опереться в решении на программирование, также могут быть представлены в виде нескольких клише (не одного, как вторая задача). Тем не менее, нельзя сводить обучение школьников программированию к отработке навыков заполнения готовых программ новыми данными и условиями. Поэтому приведем перечень основных тематических рубрик, которые должны быть включены в обязательный перечень освоения учащимися для успешного прохождения итоговой аттестации.

Итак, ключевые разделы могут быть следующими.

1. Переменные. Ввод и вывод.
2. Целые числа. Операции с ними.
2. Строки.
3. Типы данных. Преобразование типов.
4. Условный оператор.
5. Операторы циклов. Вложенные циклы.
6. Функции. Рекурсия.
7. Одномерный массив.
8. Двумерный массив.
9. Файлы. Чтение данных из файлов.
10. Более сложные структуры данных.

Заметим, что указанные разделы не зависят от языка программирования, на котором реализуется подготовка школьников, поскольку любой язык высокого уровня располагает средствами работы с указанными структурами и типами данных.

Остановимся на особенностях изучения некоторых из указанных разделов именно в свете подготовки к сдаче ЕГЭ.

Прежде всего, отметим взаимную связь целого типа данных и строк. Целый ряд задач требует умения переводить из одной системы счисления в другую. И здесь задачи можно условно разделить на требующие традиционных систем счисления (двоичной, восьмеричной, десятичной и шестнадцатеричной) или более экзотичных систем (троичных, пятеричных и т. д.). Традиционные системы счисления представлены в языках программирования функциями перевода. Например, в Python перевод числа 1234 в традиционные системы счисления может выглядеть следующим образом:

```
print(bin(1234)[2:], oct(1234)[2:], hex(1234)[2:])
```

А результат: «10011010010 2322 4d2».

Результат выполнения каждой из указанных функций является строкой, поэтому для удаления ненужных символов можно воспользоваться операцией среза для строк (срез указан в квадратных скобках). Соответственно, уже на этом примере становится ясно, что изучение методов преобразования целых чисел без понимания принципов работы со строковыми величинами приведет, как минимум, к сужению спектра используемых для решения задач инструментов.

То же самое хочется сказать и о ключевых конструкциях любого языка программирования – операторах условий и циклов. Если изучать их вне связи друг с другом, круг решаемых задач будет достаточно мал, тогда как совместное изучение этих тем даст учащимся полный набор инструментов для обработки информации любого типа. Также отметим, что тема вложенных циклов не является панацеей в решении любой задачи прямым перебором вариантов. Даже на школьном экзамене есть задачи, при переборе вариантов которой процесс решения выходит за рамки отведенного на экзамен времени (в основном это касается последней задачи). Поэтому не следует пренебрегать знаниями, полученными в курсе математики, чтобы оптимизировать алгоритмы решения.

Очень важными разделами курса обучения программированию представляются части, посвященные массивам данных: как одномерным, так и двумерным

(более сложные массивы, как правило, в школьном курсе информатики не встречаются).

Начиная с простых алгоритмов ввода и вывода массивов данных, потом обучение переходит к генерации массивов и обработке размещенных в них данных. И здесь мы вновь говорим о симбиозе тем массивов и циклов, поскольку только на базе циклического обращения к ячейкам массива можно что-либо с ними сделать. А для работы с двумерными массивами потребуются вложенные циклы.

Наконец, наиболее сложные задачи требуют обращения к файлам, как источникам данных, которые необходимо считать и записать опять же в виде массива данных для последующей обработки. Соответственно, умение обращаться с файлами, как записями (строковый тип данных), потребует максимального набора знаний и навыков, обсуждаемых выше.

Таким образом, можно констатировать, что для оптимальной подготовки школьников к сдаче итоговой аттестации по информатике в форме компьютерного ЕГЭ требуется весьма жесткий «костяк» знаний в области программирования, насыщенный большим количеством взаимных тематических зависимостей и связей, которые в других предметах принято называть внутрипредметными связями.

Список литературы

1. Босова Л.Л. Как учат программированию в XXI веке: отечественный и зарубежный опыт обучения программированию в школе / Л.Л. Босова // Информатика в школе. – 2018. – №6 (139). – С. 3–11. – ISSN 2221-1993. EDN XZOOJV
2. Минина И.В. О персонификации обучения школьников программированию / И.В. Минина, Т.П. Петухова // Современные информационные технологии и ИТ-образование. – 2018. – Т. 14. №4. – С. 986–993. – ISSN 2411-1473. – DOI 10.25559/SITITO.14.201804.986-993. EDN AUOINP
3. Сикора И.В. Обучение программированию. Опыт преподавания детям младшего школьного и среднего школьного возраста / И.В. Сикора // Педагогический журнал. – 2023. – Т. 13. №5А. – С. 324–333. – ISSN 2223-5434. – DOI 10.34670/AR.2023.77.21.034. EDN NKFNHH