

Тезисы к выступлению по теме: "Эффективные методики подготовки к
итоговой аттестации по информатике"

учителя информатики МОБУГ №2 г. Новокубанска

Сабецкой Ирины Викторовны

С 2004 года в системе ЕГЭ присутствует экзамен по информатике в качестве необязательного предмета по выбору. Перед учителем и учеником стоит цель не просто сдать ЕГЭ по информатике, а сделать это как можно более качественно, получив при этом как можно более высокие баллы. Выполнение этой задачи невозможно без решения заданий из 2 части КИМ.

При этом, следует отметить, что по данным сайта Официального информационного портала Единого Государственного Экзамена доля участников ЕГЭ по информатике, не приступавших к выполнению заданий с развёрнутым ответом (часть С) в 2011 году - 22,1%, в 2012 году - 19,9 %, в 2013 году - 26,1 % от всех учащихся выбравших предмет. Это обусловлено тем, что программа информатике и ИКТ для базового уровня изучения не может обеспечить подготовку выпускников школы к сдаче ЕГЭ. Некоторые темы, которые присутствуют в кодификаторе ЕГЭ, либо не включены в него, либо недостаточно представлены. К числу таких тем относятся в частности такие темы, как: системы счисления, логика, алгоритмизация, программирование на языках высокого уровня.

Выше приведенная ситуация говорит о недостаточной подготовке выпускников и требует разработки эффективных методик обучения решению задач части С, и в частности заданий типа С1(24).

Задание С1(24) Части 3 ЕГЭ по информатике относится к заданиям повышенного уровня сложности. К проверяемым элементам содержания относятся умение читать фрагменты программ на языках программирования, находить и исправлять допущенные в нём ошибки.

И хотя все виды ошибок предсказать невозможно, анализ заданий типа С1(24) за 2004-2015 г. г. позволяет выделить несколько типичных ошибок, которые необходимо обнаружить и исправить при выполнении этих заданий:

- 1) ошибки, связанные с вводом неверных исходных данных (вводятся не те данные, которые указаны в условии); необходимо чётко следить за тем, что дано в условии задачи;
 - а) ввод промежуточной переменной в задаче, где требуется поменять местами две переменные, используя промежуточную переменную;
 - б) отсутствует ввод переменной N (количества элементов массива) перед вводом массива из N элементов и т. д;
- 2) ошибки, связанные с выводом неверных итоговых данных (выводится не то, что требовалось найти по заданию);
- 3) ошибки в логике программы:
 - а) неверно вложенные условные операторы;
 - б) неверно сформулированные условия в условных операторах, не соответствующие поставленной задаче;
 - с) неверное условие продолжения цикла;
 - д) использование неверно выбранных стандартных функций для достижения необходимого результата;
 - е) не учитываются частные случаи решения задачи, краевые точки областей и т. д;
 - ф) частично решённые задачи;
 - г) программы, решающие немного другую задачу и т.д.

Работа по обучению решению задач типа С1(24) должна начинаться с пропедевтического этапа ещё в 9 классе. Именно в 9 классе, согласно УМК «Информатика и ИКТ» для основной школы (8–9 классы) Л.Л. Босовой и А.Ю. Босовой рассматриваются соответствующие разделы «Алгоритмизация» и «Программирование». Важно добиваться от учащихся не формального усвоения программного материала, но его глубокого осознанного понимания.

В плане практически каждого урока может быть предусмотрено время (от 5 до 15 минут) на тестирование, объем таких мини-тестов – 5–10 вопросов. Желательно при закреплении материала на уроке давать контрольные вопросы и задания в стандартном формате, соответствующем ЕГЭ. Важно систематически проводить тренировочные тесты по подготовки к ЕГЭ, выбирая материал уже пройденный учащимися.

Например, задания на поиск ошибок в записи операторов:

В каком из условных операторов допущена синтаксическая ошибка?

- a) `if B = 0 then Writeln ('Деление на нуль невозможно.');`
- b) `if a > b then max := a else max := b;`
- c) `if (a>b) and (b>0) then c:=a+b;`
- d) `If a < b then min: = a; else min: = b.`

Особое внимание при изучении раздела «Программирование» в 9 классе предполагается уделить формированию и развитию навыку построения трассировочных таблиц выполнения программ.

Предложенные методические рекомендации позволяют развить у учащихся умения анализировать программы, находить в них синтаксические и семантические ошибки и способы их исправления, то есть, обучить учащихся решению задач типа C1(24).