



ФЕДЕРАЛЬНЫЙ ИНСТИТУТ ПЕДАГОГИЧЕСКИХ ИЗМЕРЕНИЙ

**М.А. Ройтберг**

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ  
ПО НЕКОТОРЫМ АСПЕКТАМ  
СОВЕРШЕНСТВОВАНИЯ  
ПРЕПОДАВАНИЯ  
ИНФОРМАТИКИ И ИКТ**

*(на основе анализа типичных затруднений выпускников при  
выполнении заданий ЕГЭ)*

Москва, 2013

Содержание экзаменационной работы определялось на основе Федерального компонента государственного стандарта среднего (полного) общего образования, базовый и профильный уровень (Приказ Минобразования России № 1089 от 5.03.2004 г.).

Экзаменационная работа ЕГЭ 2013 года по информатике содержала 32 задания и состояла из трёх частей. В каждой из частей были сгруппированы задания одного типа. В первой части работы (А) содержалось 13 заданий с выбором ответа (выбор одного правильного ответа из четырех предложенных). Во второй части (В) были собраны 15 заданий, требующие самостоятельного формулирования краткого ответа в виде последовательности символов (например, ответом может быть целое число). И, наконец, третья часть (С) содержала 4 задания, требующие записи развернутого ответа на специальном бланке ответа в произвольной форме.

Общее время, отводимое на выполнение работы, составляло 235 минут, из которых полтора часа рекомендовалось потратить на решение заданий первой и второй части, а оставшиеся 145 минут – на задания с развернутым ответом. При этом разделение экзамена на два этапа не осуществляется, экзаменуемые получают в начале экзамена полный комплект КИМ и могут выполнять задания в любом порядке, самостоятельно определяя время, необходимое на выполнение каждого задания.

Экзамен проверяет знания и умения выпускников по десяти разделам курса информатики. При этом удельный вес разделов в экзамене различен и примерно соотносится со значением соответствующего раздела школьного курса. Наибольшее количество заданий приходится на разделы «Элементы теории алгоритмов» и «Программирование», что связано с ведущей ролью вопросов алгоритмизации и программирования в учебном предмете.

Экзамен проверял знания и умения выпускников с использованием заданий различного уровня сложности: базового (15 заданий), повышенного (13 заданий) и высокого (4 задания).

Задания базового уровня содержались только в первых двух частях работы, задания повышенного содержались во всех трех частях, одно задание высокого уровня содержалось во второй части, а остальные три задания высокого уровня требовали развернутого ответа. При этом задания базового уровня ориентированы на проверку знаний и умений инвариантной составляющей курса информатики, преподающегося в классах и учебных заведениях всех профилей. Таких заданий в работе было 15, то есть немного более половины, но их правильное решение позволяло получить только 37,5% первичных баллов (15 из 40), что давало недостаточно высокий для поступления в профильные вузы результат. Правильный ответ экзаменуемого на половину заданий базового уровня позволяет получить минимально необходимый результат для участия в конкурсном отборе для поступления в вуз. В 2013 г. Рособрнадзором был установлен минимальный уровень в 8 первичных баллов.

Из заданий базового уровня 5 заданий посвящены основам информатики (разделы «Системы счисления», «Основы логики», «Информация и её кодирование», «Моделирование»), 5 заданий – информационно-коммуникационным технологиям (разделы «Обработка числовой информации», «Технологии поиска и хранения информации», «Технология обработки графической и звуковой информации», «Архитектура компьютеров и компьютерных сетей») и 5 заданий посвящены основам теории алгоритмов и программирования. К темам, которые проверялись в заданиях базового уровня, относятся следующие: умение записывать числа в двоичной системе; элементарные сведения о графах (описание графа таблицей, длина пути и т.д.); свойства таблиц истинности для логических выражений; кодирование текстов (равномерное и неравномерное); кодирование звука; файловая система; электронные таблицы (адресация, соответствие между формулами и диаграммами); строение баз данных; умение выполнить алгоритм, записанный на естественном языке; умение создавать линейный алгоритм для формального исполнителя; знание основных алгоритмических конструкций, понятия

переменной и оператора присваивания, общие представления о рекурсивных алгоритмах.

Задания повышенного уровня (их в работе 13 из 32, и содержатся они во всех трех частях работы) проверяют освоение содержания профильного уровня стандарта 2004 г. по информатике и ориентированы на оценку подготовки выпускников, изучавших предмет по углубленной программе. Правильное решение этих заданий позволяет экзаменуемому получить 37,5% максимального первичного балла.

Из заданий повышенного уровня 5 заданий посвящены основаниям информатики (разделы «Системы счисления», «Основы логики», «Информация и её кодирование», «Моделирование»), 2 задания – информационно-коммуникационным технологиям (разделы «Архитектура компьютеров и компьютерных сетей», «Технологии поиска и хранения информации») и 6 заданий посвящены основам теории алгоритмов и программированию. К темам, которые проверялись в заданиях повышенного уровня, относятся следующие: позиционные системы счисления; определение истинности логических выражений; информационный объем сообщения; определение скорости передачи информации при заданной пропускной способности канала; адресация в Интернете; поиск при помощи сложных запросов; подсчет числа путей в графе; анализ алгоритмов, в том числе алгоритмов обработки массивов и программ, содержащих вызов пользовательской функции, поиск ошибок в программе).

И, наконец, 4 задания высокого уровня сложности были призваны выделить выпускников, в наибольшей степени овладевших содержанием учебного предмета, ориентированных на получение высшего профессионального образования в областях, связанных с информатикой и компьютерной техникой, то есть абитуриентов ведущих технических вузов. Выполнение этих заданий давало до 25% от максимального первичного балла. Среди этих заданий одно посвящено основам логики и три – теории алгоритмов и программированию.

В КИМ ЕГЭ по информатике в 2013 г. была соблюдена преемственность с КИМ 2012 г., изменения касаются только заданий В6, В13 и С3. Задание В6 КИМ ЕГЭ 2013 г. проверяет владение экзаменуемыми понятиями рекурсии и связанных с ним умений и навыков. Задание В13, сохраняя тематику 2012 г. (проверяется умение анализировать результат исполнения алгоритма), по содержанию соответствует заданию С3 из КИМ 2012 г. Результаты ЕГЭ 2012 г. показали, что получить правильный ответ в задании С3, не владея необходимым арсеналом знаний, умений и навыков, невозможно, и в данном случае нет необходимости требовать от экзаменуемого развернутого ответа. В свою очередь, задание С3 из КИМ 2013 г., как и аналогичное задание КИМ 2011 г., посвящено анализу выигрышных стратегий в детерминированных играх двух лиц с полной информацией.

Кроме того изменен порядок следования заданий в части В (см. таблицу 1), что связано с желанием более точно следовать принципу последовательного увеличения сложности заданий.

Таблица 1. Соответствие между заданиями группы В в КИМ по информатике 2013 и 2012 гг.

Задание КИМ ЕГЭ 2013 г.	В1	В2	В3	В4	В5	В6	В7	В8	В9	В10	В11	В12	В13	В14	В15
Соответствующее задание КИМ ЕГЭ 2012 г.	В2	В6	В5	В4	В3	нет	В8	В7	В9	В10	В11	В12	С3	В14	В15

Целью единого государственного экзамена является установление уровня освоения выпускниками федерального компонента государственного образовательного стандарта среднего (полного) общего образования по предмету. Используемые при этом контрольные

измерительные материалы позволяют соотнести результаты, показанные отдельными экзаменуемыми, путем выставления за работу количественной оценки по стобалльной шкале. Таким образом, становится возможным использовать результаты ЕГЭ для конкурсного отбора абитуриентов вузов.

Результаты единого государственного экзамена по информатике и ИКТ признаются образовательными учреждениями среднего профессионального и высшего профессионального образования как результаты вступительных испытаний по информатике и ИКТ. Согласно приказу Министерства образования и науки Российской Федерации от 29.10.2009 №505 (в редакции приказов от 24.01.2011 №86 и от 03.10.2011 №2433), ЕГЭ по информатике и ИКТ входил в перечень вступительных испытаний для 129 специальностей высшего образования, не только непосредственно связанных с ИКТ и вычислительной техникой, но и многих общеинженерных, технологических специальностей, а также для физико-математических специальностей классических и педагогических университетов. Экзамен по информатике и ИКТ входит в перечень специальностей картографического и гидрометеорологического направлений. При этом ни для одной из специальностей информатика и ИКТ не является профильным общеобразовательным предметом (им для большинства специальностей является математика, также в таком качестве для трех специальностей выступает физика, для шести специальностей – география).

Согласно положению, единый государственный экзамен проверяет знания и умения выпускников по предмету «Информатика и ИКТ» по результату обучения в старшей школе. Однако, согласно Государственным образовательным стандартам 2004 года, изучение информатики и ИКТ начинается в основной школе (8-9 классы). Естественно, что полученное в рамках основной школы образование по предмету является фундаментом для обучения на этапе старшей школы, без овладения содержанием и достижения требований, предусмотренных стандартом основной школы, невозможно достижение требований стандарта среднего (полного) общего образования. Тем более что информатика и ИКТ могут изучаться в старшей школе на базовом или профильном уровне, а требования базового и профильного стандарта различаются очень значительно.

Структура и объем учебного плана по информатике в образовательных учреждениях разных типов и видов существенно варьируется: от 240 часов в старших классах информационно-технологического профиля до 70 часов базового курса в классах гуманитарных профилей (и то и другое предусмотрено стандартом). В этой связи контрольные измерительные материалы содержат задания, рассчитанные как на выпускников профильных классов, так и на тех, кто прослушал только базовый курс для старшей школы (то есть задания, проверяющие достижение требований как базового, так и профильного стандарта).

Каждое задание экзаменационной работы характеризуется одновременно проверяемым содержанием учебного предмета, требованием к подготовке выпускника, установленным федеральным образовательным стандартом по предмету, достижению которого проверяется с помощью этого задания, а также уровнем сложности и характером осуществляемой деятельности (воспроизведение знаний и умений, применение их в стандартной либо новой ситуации). Структура экзаменационной работы моделируется исходя из принципов всесторонней проверки: задания экзаменационной работы должны проверять все существенные элементы содержания предмета и достижение всех требований к уровню подготовки, которые возможно проверить в формате ЕГЭ.

В КИМ ЕГЭ по информатике не включены задания, требующие простого воспроизведения знания терминов, понятий, величин, правил (такие задания слишком просты для выполнения). При выполнении любого из заданий КИМ от экзаменуемого требуется решить тематическую задачу: либо прямо использовать известное правило, алгоритм, умение, либо выбрать из общего количества изученных понятий и алгоритмов наиболее подходящее и применить его в известной или новой ситуации.

Знание теоретического материала проверяется косвенно через понимание используемой терминологии, взаимосвязей основных понятий, размерностей единиц и т.д., при выполнении экзаменуемыми практических заданий по различным темам предмета. Таким образом, в КИМ по информатике и ИКТ проверяется освоение теоретического материала из разделов:

- единицы измерения информации;
- принципы кодирования;
- системы счисления;
- моделирование;
- понятие алгоритма, его свойств, способов записи;
- основные алгоритмические конструкции;
- основные понятия, используемые в информационных и коммуникационных технологиях.

Введение раздела 2 кодификатора («Перечень требований к уровню подготовки выпускников») потребовало появления еще одной характеристики используемых в КИМ заданий: по видам умений и способам действий. Отметим, что стандарты образования по информатике содержат значительное число требований к использованию приобретенных знаний и умений в практической жизни. Однако, установленная стандартизированная бланковая форма единого государственного экзамена, ориентированная на проверку, в первую очередь, теоретических знаний, не позволяет проверить выполнение этих требований в полном объеме. Использование в процессе экзамена компьютеров для выполнения заданий экзаменационной работы позволит увеличить долю заданий, проверяющих использование знаний и умений в практической деятельности.

Контрольные измерительные материалы ЕГЭ по информатике и ИКТ проверяют знания и умения в трех видах ситуаций: воспроизведения, применения знаний в стандартной либо новой ситуации.

В работе 6 (из общего количества 32) заданий *первого вида* (требующих воспроизведения знаний), они входят в первую и вторую часть работы. Эти задания решаются в одно-два действия, и предполагают формальное выполнение изученного алгоритма или применение правила (подстановку значений в формулу). Задания первого вида могут быть как базового, так и повышенного уровня сложности. За выполнение таких заданий можно получить 6 первичных баллов из 40 возможных.

Задания *второго вида* (требующие умений применять свои знания в стандартной ситуации), входящие во все три части экзаменационной работы, предусматривают использование комбинации правил или алгоритмов, совершение последовательных действий, однозначно приводящих к верному результату. Предполагается, что экзаменуемые в процессе изучения школьного курса информатики приобрели достаточный опыт в решении подобных задач. К этому типу, в частности, относится одно из заданий третьей части работы (задание С2), требующее формальной записи изученного в школе алгоритма обработки массива на языке программирования, либо на естественном языке. Это задание относится к высокому уровню сложности, т.к. комплексно проверяет владение синтаксисом языка программирования, знание проверяемого алгоритма, умение пользоваться оператором присваивания и конструкциями цикла и ветвления. Задания второго вида встречаются в экзаменационной работе чаще всего (15 заданий из 32, за них можно получить 16 первичных баллов из 40 возможных).

Следующие умения проверяются через их применение в стандартной ситуации:

- подсчитывать информационный объем сообщения;
- искать кратчайший путь в графе, осуществлять обход графа;
- осуществлять перевод из одной системы счисления в другую;
- использовать стандартные алгоритмические конструкции при программировании;

- формально исполнять алгоритмы, записанные на естественных и алгоритмических языках, в том числе на языках программирования;
- формировать для логической функции таблицу истинности и логическую схему;
- оценивать результат работы известного программного обеспечения;
- оперировать массивами данных;
- формулировать запросы к базам данных и поисковым системам.

Задания *третьего вида*, проверяющие умения применять знания в новой ситуации, входят во вторую и третью часть работы (всего 11 заданий из 32, дают максимально 18 первичных баллов из 40). Они предполагают решение учащимися творческой задачи: какие изученные правила и алгоритмы следует применить, в какой последовательности это следует сделать, какие данные использовать. К этому типу относятся задания на анализ результата исполнения алгоритма при различных исходных данных, на поиск и устранение ошибок в алгоритмах, на самостоятельное написание программ, задания, предполагающие прогнозирование результатов поиска в Интернет.

На уровне применения в новой для экзаменуемого ситуации проверяются следующие сложные умения:

- анализировать однозначность двоичного кода;
- анализировать обстановку исполнителя алгоритма;
- определять основание системы счисления по свойствам записи чисел;
- определять мощность адресного пространства компьютерной сети по маске подсети в протоколе TCP/IP;
- осуществлять преобразования логических выражений;
- моделировать результаты поиска в Интернет;
- анализировать текст программы с точки зрения соответствия записанного алгоритма поставленной задаче и изменять его в соответствии с заданием;
- находить и обосновывать выигрышную стратегию для игры двух лиц с полной информацией;
- реализовывать сложный алгоритм с использованием современных систем программирования.

Выполнение каждого задания (или каждого элемента задания при политомической оценке) оценивается одним баллом, независимо от сложности задания и степени его новизны для экзаменуемого. Полученные баллы суммируются, что позволяет выделить группы выпускников со схожим уровнем подготовки и, в результате, осуществить конкурсный отбор абитуриентов в учреждения профессионального образования. Минимальная граница первичных баллов, позволяющая получить сертификат ЕГЭ по предмету, определяется исходя из содержания базового стандарта. В то же время КИМ должен обеспечивать адекватную оценку компетентностей выпускников с высоким уровнем подготовки, поэтому он содержит задания высокого уровня сложности, требующие применения знаний и умений в новой для экзаменуемого ситуации.

Содержание экзамена составлено таким образом, чтобы на результат существенным образом не влияло то, по какой программе или учебно-методическому комплексу велось преподавание в конкретном образовательном учреждении, какое программное обеспечение использовалось, хотя отсутствие тех или иных предусмотренных стандартом элементов содержания в конкретном учебном курсе будет, естественно, влиять на итоговую оценку подготовки выпускников. Также очевидно, что полностью исключить влияние компьютеризации учебного процесса в образовательном учреждении на результаты ЕГЭ по информатике и ИКТ невозможно, но содержание экзаменационной работы позволяло выпускникам, изучавшим информатику и ИКТ в «безмашинном» варианте, преодолеть минимальную границу и получить достаточный для поступления на непрофильную техническую специальность балл.

Таким образом, ЕГЭ по информатике и ИКТ решает одновременно две задачи: во-первых, оценить усвоение выпускниками содержания учебной программы и достижение предусмотренных стандартом требований, и, во-вторых, — ранжировать экзаменуемых по уровню подготовки к продолжению образования в сфере информатики и компьютерных технологий.

Как уже было сказано, в экзаменационной работе используются три вида заданий: с выбором одного из четырех предложенных ответов, с кратким ответом и с развернутым ответом. Использование заданий с выбором ответа позволяет проверить материал, который не проверяется заданиями с кратким ответом (в силу содержательной невозможности формулирования ответа в виде числа или краткой записи). В то же время, при анализе результатов выполнения заданий с выбором ответа надо учитывать возможность случайного угадывания ответа.

Задания с кратким ответом предполагают, в подавляющем большинстве случаев, ответ в виде натурального числа. Ответы такого формата распознаются компьютером практически безошибочно. В 2013 г. на экзамене не использовались задания, предполагающие ответ в виде набора чисел, разделенных запятыми (в прежние годы такие задания существовали). Все задания с кратким ответом предполагают дихотомическую оценку («верно» - «неверно»), неполный или частично правильный ответ невозможен.

Задания с развернутым ответом проверяются экспертами по единым критериям. Это задания комплексные, проверяющие совокупность знаний и умений. В связи с этим во всех заданиях с развернутым ответом используется политомическая оценка. В силу трудоемкости заданий с развернутым ответом (их в работе всего 4), они имеют повышенный (1 задание) или высокий (3 задания) уровень сложности и предназначены для хорошо подготовленных экзаменуемых. Следует ответить, что доля участников экзамена, не приступивших к выполнению заданий с развернутым ответом, с каждым годом снижается, но остается достаточно высокой.

При разработке критериев оценивания заданий с развернутым ответом учитывается факт разнообразия учебных программ и изучаемых языков программирования. При выполнении заданий С2 и С4, предполагающих написание программы, экзаменуемый может использовать любой язык программирования, требуется только точно назвать версию языка. В связи с этим критерии оценивания пишутся в общей форме, и не зависят от выбранного языка программирования. Этот подход предъявляет высокие требования к квалификации экспертов, проверяющих задания: они должны быть готовы проверять задания, выполненные на различных языках программирования.

Также во всех заданиях, содержащих фрагменты алгоритмов, записанных на формальном языке, они предъявляются на четырёх различных языках программирования: школьном Алгоритмическом языке, Си, Паскале и Бейсике.

Критерии оценивания заданий с развернутым ответом учитывают невозможность для экзаменуемых проверить решение в процессе отладки в среде программирования и поэтому толерантны к синтаксическим ошибкам, не искажающим замысел автора программы.

Далее сформулированы общие рекомендации по подготовке к ЕГЭ 2014 г.

Во-первых, ЕГЭ по информатике – экзамен по выбору, но, тем не менее, выпускников, сдающих этот экзамен, по уровню исходной подготовки можно разделить на три группы: базовый уровень; хороший уровень и отличный уровень.

Экзаменуемые с базовым уровнем подготовки, как правило, выполняют только задания базового уровня сложности (от 10 до 15 таких заданий). Эти задания охватывают (на базовом уровне) основной материал курса информатики и ИКТ, в том числе, темы «Двоичное представление чисел», «Файловая система персональных компьютеров», «Базы данных», «Электронные таблицы», «Кодирование текстовой информации», «Основы логики», «Основы теории алгоритмов». Работа экзаменуемых этой группы происходит, в основном, на уровне воспроизведения и применения знаний в стандартной ситуации,

Абитуриенты этого уровня будут испытывать затруднения при изучении информатики и ИКТ в профильных учреждениях высшего профессионального образования.

При подготовке к ЕГЭ ученикам этой группы разумно сосредоточиться на темах, проверяемых заданиями А1 – А9, В1 – В6. Вместе с тем предпочтительная стратегия состоит в том, чтобы более глубоко изучить курс в целом и поднять свой уровень знаний с базового до хорошего.

Экзаменуемые с хорошим уровнем подготовки, как правило, выполняют все задания, кроме наиболее сложных заданий (А12, В15, С2, С3, С4). Говоря о группе в целом, следует обратить внимание на тему «алгоритмы и программирование», которой посвящены 4 из 5 наиболее трудных задач ЕГЭ 2013 г. Кроме того, ученики этой группы имеют, как правило, пробелы в отдельных темах – индивидуальные для каждого ученика. Эти проблемные темы должны быть своевременно выявлены с помощью тестовых работ и тщательно разобраны. Подчеркнем, что речь идет именно об изучении тем, а не о натаскивании на отдельные задания.

Экзаменуемые с отличным уровнем подготовки показывают хорошее знание всех разделов курса информатики и ИКТ и готовность к продолжению образования на профильных специальностях учреждений высшего профессионального образования. Однако даже среди этой группы процент выполнения задачи С4, требующей самостоятельно написать программу, достаточно низок (некоторым оправданием этого является то, что разработка программы ведется на бумаге, без использования привычной программной среды). Таким образом, резерв в повышении результатов этой группы состоит в изучении программирования (задача С4) и уменьшении количества потерянных баллов в других задачах.

Во-вторых, следует учитывать неравномерность изучения различных тем в образовательных учреждениях. Рассматривая отдельные темы, можно отметить хорошее, в целом, решение экзаменуемыми заданий по темам «Системы счисления» и «Файловая система» и, одновременно, недостаточно высокие результаты по ряду других тем. Следует обратить внимание на изучение понятийного аппарата и основных формул, связанных с организацией и функционированием компьютерных сетей, передачей данных, кодированием звуковых и графических данных. При обучении на профильном уровне следует уделить большее внимание разработке программ в рамках требований, предъявляемых стандартом образования (задача С4). Как при профильном, так и при базовом обучении, следует максимальное внимание уделять решению задач, в том числе – решению практических задач на построение алгоритмов с помощью компьютера. Большую помощь при этом могут оказать практикумы, включающие наборы задач по разным темам и допускающие выполнение обучающимися самопроверки (см., например, доклад К.Ю.Полякова <http://kpolyakov.narod.ru/school/doklad.htm#vluki13> ). При обучении важно обращать внимание учащихся не только на простые факты и формулы, но и на более глубокие связи между объектами и понятиями. Для проверки глубины освоения материала в ЕГЭ есть задания (А1, А3, А8, В4, В5, В9, В13, В14, В15), решение которых, основанного на понимании особенностей задачи, существенно проще решения, основанного на прямом применении заученных формул.

При подготовке выпускников к единому государственному экзамену учителям следует подробнее объяснять учащимся цели этого испытания и структуру экзаменационной работы. Так как экзамен используется и для оценки уровня усвоения образовательной программы, и для ранжирования подготовки абитуриентов к продолжению обучения на профильных специальностях вузов, экзаменационная работа содержит набор заданий различной сложности, расположенных по возрастанию сложности и преследующих различные цели. Будущему участнику экзамена надо четко определиться с тем, какие цели он ставит и, соответственно, в какую из групп по уровню результатов планирует попасть. Как показывают результаты экзамена, только небольшая часть его участников показали необходимый для продолжения образования на



профильных специальностях уровень подготовки. Эта подготовка включает умение использовать электронные таблицы для обработки статистических данных, в том числе результатов научных исследований, умение самостоятельно разрабатывать программы на языках программирования для решения практических задач обработки массивов данных, умение использовать ресурсы Интернет для поиска и систематизации информации. Поэтому следует обратить внимание на такие разделы кодификатора содержания, как 1.3.2 (Математические модели), 1.5.2 (Цепочки (конечные последовательности), деревья, списки, графы, матрицы (массивы), псевдослучайные последовательности), 1.5.6 (Сортировка), 3.4.1 (Математическая обработка статистических данных), 3.5.2 (Использование инструментов поисковых систем, формирование запросов). Надо иметь в виду, что учреждения высшего профессионального образования заинтересованы в абитуриентах, чья подготовка соответствует следующим требованиям кодификатора требований: 1.1.1 (Проводить вычисления в электронных таблицах), 1.1.5 (Создавать программы на языке программирования), 2.9 (Проводить статистическую обработку данных с помощью компьютера).

Следует понимать, что требования учреждений высшего профессионального образования к подготовке абитуриентов профильных специальностей предполагают уровень подготовки, соответствующий профильному курсу информатики и ИКТ, поэтому выпускникам с базовой подготовкой не следует рассчитывать на высокий результат ЕГЭ. Задания базового уровня обеспечивали экзаменуемому максимум 15 первичных баллов, что соответствует «удовлетворительному» уровню, недостаточному для продолжения образования на профильной специальности.

При подготовке к ЕГЭ задания демонстрационной версии экзамена надо рассматривать только как ориентиры, показывающие примерные образцы заданий, которые могут стоять на соответствующих позициях. Натаскивание абитуриентов на определенный тип заданий может привести к тому, что они будут обескуражены заданием, немного отличающимся по формату от того, которое было ими многократно разобрано.

В качестве ресурсов, которые полезно использовать при подготовке к ЕГЭ по информатике, укажем такие бесплатные ресурсы, как ресурс К.Ю.Полякова <http://kpolyakov.narod.ru/> (разделы <http://kpolyakov.narod.ru/school/ege.htm> и <http://kpolyakov.narod.ru/school/kumir.htm>), ресурс <http://ege.yandex.ru/> (раздел <http://ege.yandex.ru/informatics/>), а также специализированный ресурс по информатике и математике <http://ege-go.ru/>.

В заключение отметим, что, к сожалению, для информатики не существует открытого банка задач, подобного тому, который есть, например, для математики. Создание такого банка необходимо для развития образования в области информатики в нашей стране. Попытка создания открытого банка задач, который можно использовать при изучении информатики, в настоящее время ведется на сайте <http://ege-go.ru/> (раздел <http://ege-go.ru/zadaniya/>), однако эта работа пока далека от завершения.

По опыту единого государственного экзамена 2012 – 2013 годов, наибольшее затруднение вызывают задания А10, В13, В14, В15. Разберем подробно эти задания.

**Задание В13** относится к разделу «Алгоритмы». Оно проверяет умение построить дерево путей вычислений, а также умение анализировать такое дерево, например, вычислять количество путей, удовлетворяющих заданным свойствам. Задание может выглядеть, например, так: «Дано описание исполнителя, который работает с натуральными числами. Команды этого исполнителя преобразуют данное число в новое число, которое больше исходного. Программа для такого исполнителя – последовательность команд. Требуется подсчитать количество программ, которые переводят число  $a$  в число  $b$ .»

Сложность задания – повышенная; примерное время решения около 20 минут.

*Пример условия (демо-версия 2013 г.)*

«У исполнителя Увеличитель две команды, которым присвоены номера:



его один раз (например, после умножения 3 на 2), можно подсчитать, что есть 7 способов получить число 23 из числа 6. Строить второй раз поддерево с корнем 6 не нужно! Решение с помощью построения таблицы (см. п. 3.2.4) и есть способ систематически избегать лишней работы.

Отметим, что, если бы вместо числа 23 взять, скажем, число 50 (и, заведомо, если взять 100), то, пожалуй, задача станет неподъемной. А описанным ниже методом решения с помощью заполнения таблиц это сделать довольно легко.

*Второй способ решения: заполнение таблицы*

Будем решать поставленную задачу последовательно для чисел 3, 4, ..., 23. То есть для каждого из чисел определим, сколько программ исполнителя существует для получения этого числа из числа 3. Количество программ, которые преобразуют число 3 в число  $n$ , будем обозначать через  $R(n)$ . Для удобства будем заполнять такую таблицу (см. рис. 2а). Для удобства таблица разбита на две части – для чисел от 3 до 13 и для чисел от 14 до 23.

Число $n$	3	4	5	6	7	8	9	10	11	12	13
Из каких чисел можно получить число $n$ за один шаг											
Количество программ, с помощью которых можно получить число $n$ из числа 3 (не обязательно за один шаг)											

Число	14	15	16	17	18	19	20	21	22	23
Из чего можно получить число $n$ за один шаг										
Количество программ, с помощью которых можно получить число $n$ из числа 3 (не обязательно за один шаг)										

Рис 2а.

В таблице средняя строка показывает, из каких чисел может быть получено данное число за одно действие. Таблицу удобно заполнять так. Двигаясь от меньших чисел к большему, то есть в порядке 3, 4, ..., 22, для каждого числа отмечаем, какие числа могут быть получены из этого числа. На рис. 2б показана верхняя часть таблицы после обработки чисел 3 и 4; на рис. 2в – эта же часть после того, как дополнительно обработано число 5.

На рисунке 2г средняя строка таблицы заполнена полностью. Если число не делится на два, то оно может быть получено только из предыдущего с помощью команды **прибавь 1**. Если число на 2 делится, то вариантов последней команды два: **прибавь 1** и

умножь на 2,

Число $n$	3	4	5	6	7	8	9	10	11	12	13
Из каких чисел можно получить число $n$ за один шаг		3	4	3		4					
Количество программ, с помощью которых можно получить число $n$ из числа 3 (не обязательно за один шаг)											

Рис. 2б

Число $n$	3	4	5	6	7	8	9	10	11	12	13
Из каких чисел можно получить число $n$ за один шаг		3	4	3, 5		4		5			
Количество программ, с помощью которых можно получить число $n$ из числа 3 (не обязательно за один шаг)											

Рис. 2в

Теперь перейдем к заполнению нижней строки.

Число 3 у нас уже есть, значит, его можно получить с помощью “пустой” программы. Любая непустая программа увеличит исходное число, т.е. даст число, больше 3. Значит,  $R(3) = 1$ . Для каждого следующего числа рассмотрим, из какого числа оно может быть получено за одну команду исполнителя.

Если число не делится на два, то оно может быть получено только из предыдущего с помощью команды **прибавь 1**. Значит, количество искомых программ для такого числа равно количеству программ для предыдущего числа:  $R(n) = R(n-1)$ . Если число на 2 делится, то вариантов последней команды два: **прибавь 1** и **умножь на 2**, тогда  $R(n) = R(n-1) + R(n/2)$ .

В общем виде это можно записать так:

$$R(n) = R(p1) + R(p2),$$

где  $p1, p2$  – числа, из которых число  $n$  можно получить за одно действие. Понятно, что если у число  $n$ , не 2, а, один «предшественник», то формула будет выглядеть  $R(n) = R(p)$ .

*Замечание.* Можно придумать исполнителя, у которого будет больше двух команд. «Формула сложения» остается верной для случая любого количества предшественников у числа  $n$ .

Число $n$	3	4	5	6	7	8	9	10	11	12	13
Из каких чисел можно получить число $n$ за один шаг		3	4	3, 5	6	4, 7	8	5, 9	10	6, 11	12
Количество программ, с помощью которых можно получить число $n$ из числа 3 (не обязательно за один шаг)											

Число	14	15	16	17	18	19	20	21	22	23
Из чего можно получить число $n$ за один шаг	7, 13	14	8, 16	16	9, 17	18	10, 19	20	11, 21	22
Количество программ, с помощью которых можно получить число $n$ из числа 3 (не обязательно за один шаг)										

Рис 2г.

Пользуясь формулой сложения, двигаясь от меньших чисел к большим, можно заполнить нижнюю строку таблицы. Как сказано выше,  $R(3) = 1$ . Далее, по формуле сложения и используя уже заполненную среднюю строку, получаем:  $R(4) = R(3) = 1$ ;  $R(5) = R(4) = 1$ ;  $R(6) = R(3) + R(5) = 1 + 1 = 2$  и т.д. Полностью заполненная таблица приведена на рисунке 3.

Число $n$	3	4	5	6	7	8	9	10	11	12	13
Из каких чисел можно получить число $n$ за один шаг		3	4	3, 5	6	4, 7	8	5, 9	10	6, 11	12
Количество программ, с помощью которых можно получить число $n$ из числа 3 (не обязательно за один шаг)	1	1	1	$1+1=2$	2	$1+2=3$	3	$1+3=4$	4	$2+4=6$	6

Число	14	15	16	17	18	19	20	21	22	23
Из чего можно получить число $n$ за один шаг	7, 13	14	8, 16	16	9, 17	18	10, 19	20	11, 21	22
Количество программ, с помощью которых можно получить число $n$ из числа 3 (не обязательно за один шаг)	$2+6=8$	8	$3+8=11$	11	$3+11=14$	14	$4+14=18$	18	$4+18=22$	22

Рис 3.

**Ответ:** 22.

*Замечание.* Как видим, количество программ быстро растет с ростом числа  $n$ . Например, для  $n = 24$  получим  $R(n) = 30$ ;  $n = 26$  получим  $R(n) = 36$ ;  $n = 30$  получим  $R(n) = 52$ . При заполнении таблицы дополнительная работа невелика, а количество вершин в дереве, которое нужно построить, увеличилось более, чем в два раза.

**Задание В14** также относится к разделу «Алгоритмы». Оно посвящено анализу алгоритма, записанного с использованием вспомогательных алгоритмов. В нем проверяется умение ученика понять, что делает алгоритм, не выполняя полную трассировку. В том числе - умение распознавать стандартные алгоритмы обработки данных (поиск экстремума, суммирование и т.п., см. «Кодификатор элементов содержания и требований к уровню подготовки выпускников общеобразовательных учреждений для проведения в 2012 году единого государственного экзамена по информатике и ИКТ»).

Задание может выглядеть, например, так. Дается описание алгоритма на четырех алгоритмических языках. Основной алгоритм – один из стандартных алгоритмов, предусмотренных в классификаторе (определение минимума или максимума массива, суммы массива и т.п.) Используется несложный (несколько строк) вспомогательный алгоритм. Требуется определить значение, которое одна из переменных основного алгоритма принимает после выполнения алгоритма.

*Пример условия из демонстрационного варианта ЕГЭ-2012.*

«Определите, какое число будет напечатано в результате выполнения следующего алгоритма, изображенного на рис.4 (для Вашего удобства алгоритм представлен на четырех языках).»

Бейсик	Паскаль
<pre> a = -20: b = 20 M = p1: R = F(p1): M = R  FOR t = a TO b IF F(t) &lt; R THEN M = t R = F(t) END IF NEXT t PRINT M  FUNCTION F (x) F = (x - 1) * (x - 3) </pre>	<pre> Var a,b,t,M,R :integer; Function F(x : integer): integer; begin F:=4*(x-1)*(x-3); end; BEGIN a:=-20; b:=20; M:=a; R:=F(a); for t:= a to b do begin if (F(t)&lt;R) then begin </pre>

END FUNCTION	<pre> M:=t; R:=F(t); end; end; write(M); END.</pre>
<b>Си</b>	<b>Алгоритмический язык</b>
<pre> int F(int x) {     return (x-1)*(x-3); } void main() {     int a, b, t, M, R;     a = -20; b = 20;     M = a; R = F(a);     for (t=a; t&lt;=b; t++){         if ( F(t)&lt;R ) {             M = t; R = F(t);         }     }     printf("%d", M); } </pre>	<pre> <b>алг</b> <b>нач</b>     <b>цел</b> a, b, t, R, M     a:= -20; b:= 20     M:= a; R:= F(a)     <b>нц для t от a до b</b>         <b>если</b> F(t)&lt; R             <b>то</b>                 M:= t; R:= F(t)         <b>все</b>     <b>кц</b>     <b>вывод</b> M <b>кон</b>  <b>алг цел F(цел x)</b> <b>нач</b>     <b>знач</b> := 4*(x-1)*(x-3) <b>кон</b> </pre>

Рис. 4

*Набросок решения.* Для краткости мы ограничимся анализом записи алгоритма на Школьном Алгоритмическом Языке, см. рис. 5. На рисунке строки перенумерованы (при использовании системы Кумир они будут перенумерованы так же).

1	<b>алг</b>
2	<b>нач</b>
3	<b>цел</b> a, b, t, R, M
4	a:= -20; b:= 20
5	M:= a; R:= F(a)
6	<b>нц для t от a до b</b>
7	<b>если</b> F(t)< R
8	<b>то</b>
9	M:= t; R:= F(t)
10	<b>все</b>
11	<b>кц</b>
12	<b>вывод</b> M
13	<b>кон</b>
14	
15	<b>алг цел F(цел x)</b>
16	<b>нач</b>
17	<b>знач</b> := 4*(x-1)*(x-3)
18	<b>кон</b>

Рис. 5.

В алгоритме используется вспомогательный алгоритм-функция F, этот алгоритм имеет единственный аргумент типа **цел**. Значение F(x) при значении аргумента x - это

значение квадратичного многочлена  $4*(x-1)*(x-3)$ . См. строки 15 - 18.

Алгоритм перебирает все целочисленные значения  $t$  от  $a=-20$  до  $b=20$ . См. цикл от строки 6 до строки 11.

До начала цикла переменной  $M$  присваивается значение  $a$ , а переменной  $R$  - значение  $F(a)$ . См. строку 5. Внутри цикла вычисляется значение  $F(t)$  для очередного значения переменной  $t$ . Если выполнено условие  $F(t) < M$ , то перевычисляются значения  $R$  и  $M$  - так, как показано в строках 7 - 10. Это означает, что переменная  $R$  хранит текущее минимальное значение величин  $F(t)$ , а переменная  $M$  - первое (т.е. наименьшее) значение  $t$ , при котором был достигнут этот минимум. То, что хранится именно первая точка минимума, определяется тем, что в строке 7 проверяется строгое неравенство.

Таким образом, после выполнения цикла в строках 6 - 11 значение переменной  $M$  равно первой из точек минимума значений функции  $F(x)$  в целочисленных точках  $x=-20, \dots, 20$ .

Из курса алгебры (9 класс) известно, что квадратичная функция  $F(x) = 4*(x-1)*(x-3)$  достигает минимума в точке 2. Поэтому после завершения выполнения цикла переменная  $M$  будет иметь значение 2 и при выполнении команды вывод в строке 12 будет напечатано 2.

**Ответ:** 2

*Рекомендации, как разбирать задачу с учениками*

Чтобы выполнить задание, ученик должен уметь:

1. распознавать записанные на языке программирования стандартные алгоритмы обработки данных (в разобранном примере – это алгоритм поиска минимального элемента в наборе чисел);
2. распознавать вспомогательный алгоритм и понимать, что он делает;
3. разобравшись в отдельных элементах программы, понять, что делает программа в целом.

Чтобы овладеть этими умениями, необходим опыт практической работы в той или иной среде программирования. Если у учеников такой опыт уже есть, можно ограничиться разбором нескольких примеров и тренингом, чтобы «набить» руку. Кроме заданий типа В14 в качестве тренинга на распознавание стандартных алгоритмов можно давать задачи типа А12. Например, предлагать ученикам текст алгоритма и 4 варианта ответа того, что произойдет после его выполнения. Например, варианты ответов могут быть такими: 1) ищет минимальное значение, 2) ищет первую точку минимума, 3) ищет последнюю точку минимума, 4) ищет количество минимумов. Полезно дать ученикам возможность попробовать решить задачу подобного типа, а потом проверить себя, выполнив программу в реальной среде программирования.

Работая с менее подготовленными учениками, предварительно придется изучить (повторить) темы «Стандартные алгоритмы» и «Вспомогательные алгоритмы».

**Задания по логике** занимают значительное место в экзаменационной работе. В этом разделе мы разбираем задание В15, которое вызывает наибольшие трудности у сдающих экзамен, а также задание А10, которое в его нынешнем виде появилось в ЕГЭ впервые в 2013 г.

**Задание А10** - повышенной сложности. Оно проверяет умение выполнять эквивалентные преобразования логических выражений, а также умение проверять истинность высказываний о математических объектах. Высказывания могут включать элементарные утверждения ("точка  $P$  лежит внутри квадрата  $ABCD$ ", "число  $K$  делится на 5" и т.п.), логические операции (отрицание, конъюнкция, дизъюнкция, импликация, эквивалентность). Проверяется также умение понимать утверждения всеобщности (примеры: "при любом  $x$  выполнено  $x^2 - x + 1 > 0$ "; "при любом  $x$  выполнено:  $(x > 10 \vee x < 1) \rightarrow x^2 - 5x + 6 > 0$ "); и определять истинность таких высказываний для заданного множества объектов.

Задание может выглядеть, например, так.



(1) Дано выражение, зависящее от аргументов определенного типа (слов, чисел и пр.) и несколько наборов значений аргументов. Надо определить, при каком наборе аргументов выражение истинно (ложно), на скольких наборах выражение истинно (ложно) и т.п.

(2) Дан набор высказываний всеобщности относительно некоторых множеств объектов (чисел, слов, точек и т.п.). Требуется указать истинное (ложное) высказывание, количество истинных (ложных) высказываний и т.п.

*Пример задания A10.*

«На числовой прямой даны два отрезка:  $P = [21, 29]$  и  $Q = [12, 42]$ .

Выберите такой отрезок  $A$ , что формула

$((x \in P) \rightarrow (x \in A)) \wedge ((x \in A) \rightarrow (x \in Q))$

тождественно истинна, то есть принимает значение 1 при любом значении переменной  $x$ .

1)  $[7, 33]$       2)  $[17, 33]$       3)  $[7, 43]$       4)  $[17, 43]$ »

*Решение:* Первый элемент конъюнкции  $((x \in P) \rightarrow (x \in A))$  истинен для всех  $x$ , если  $P \subseteq A$ . Второй элемент конъюнкции  $((x \in A) \rightarrow (x \in Q))$  истинен для всех  $x$ , если  $A \subseteq Q$ . Таким образом, конъюнкция тождественно истинна тогда и только тогда, когда  $P \subseteq A \subseteq Q$ . Этому условию удовлетворяет только отрезок  $[18, 31]$ .

*Ответ:*  $[17, 33]$  (вариант 2).

Полезные рекомендации содержатся в статье К.Ю.Полякова, которую можно скачать по ссылке <http://kpolyakov.narod.ru/download/inf-2013-02.pdf>

**Задание B15** - высокого уровня сложности. Оно проверяет умение преобразовывать выражения, содержащие логические переменные, умение описать на естественном языке множество значений логических переменных, при которых заданный набор логических высказываний истинен. Задание может выглядеть, например, так.

«Найти количество решений системы логических уравнений. Предполагается, что ученик опишет множество решений системы, после чего подсчитает, сколько элементов есть в этом множестве.»

Для того, чтобы выполнить задание, ученик должен уметь

- преобразовывать логические выражения (включая выполнение замены переменных);
- переводить формальное описание, в виде системы логических условий, на нормальный, "человеческий" язык ;
- подсчитать число двоичных наборов, удовлетворяющих заданным условиям.

После того, как выяснено, что за наборы удовлетворяют системе, подсчет их числа относительно прост.

Наиболее трудным для усвоения, видимо, является второе из перечисленных требований – оно не формализуется, от ученика, как правило, требуется догадка.

*Пример задания B15 (демо-версия 2012 г.)*

Сколько существует различных наборов значений логических переменных  $x_1, x_2, \dots, x_9, x_{10}$ , которые удовлетворяют всем перечисленным ниже условиям?

$$((x_1 \equiv x_2) \vee (x_3 \equiv x_4)) \wedge (\neg(x_1 \equiv x_2) \vee \neg(x_3 \equiv x_4)) = 1$$

$$((x_3 \equiv x_4) \vee (x_5 \equiv x_6)) \wedge (\neg(x_3 \equiv x_4) \vee \neg(x_5 \equiv x_6)) = 1$$

$$((x_5 \equiv x_6) \vee (x_7 \equiv x_8)) \wedge (\neg(x_5 \equiv x_7) \vee \neg(x_7 \equiv x_8)) = 1$$

$$((x_7 \equiv x_8) \vee (x_9 \equiv x_{10})) \wedge (\neg(x_7 \equiv x_8) \vee \neg(x_9 \equiv x_{10})) = 1$$

В ответе **не нужно** перечислять все различные наборы значений  $x_1, x_2, \dots, x_9, x_{10}$ , при которых выполнена данная система равенств. В качестве ответа вам нужно указать количество таких наборов.

*Решение* состоит из двух этапов. Сначала попытаемся описать, как устроены все наборы значений переменных, удовлетворяющие данной системе. Далее подсчитаем число таких наборов.

*Этап 1. Как устроено множество решений*

А. Предварительный этап – упрощаем уравнения.

В системе фигурируют логические функции от следующих выражений:

$$(x_1 \equiv x_2), (x_3 \equiv x_4), (x_5 \equiv x_6), (x_7 \equiv x_8), (x_9 \equiv x_{10})$$

Подобно тому, как это делается при решении алгебраических уравнений, сделаем замену переменных:

$$\begin{aligned}t_1 &= x_1 \equiv x_2 \\t_2 &= x_3 \equiv x_4 \\t_3 &= x_5 \equiv x_6 \\t_4 &= x_7 \equiv x_8 \\t_5 &= x_9 \equiv x_{10}\end{aligned}$$

Общая формула замены ( $k=1, 2, 3, 4, 5$ ):

$$t_k = (x_{2k-1} \equiv x_{2k})$$

Получим:

$$\begin{aligned}(t_1 \vee t_2) \wedge (\neg t_1 \vee \neg t_2) &= 1 \\(t_2 \vee t_3) \wedge (\neg t_2 \vee \neg t_3) &= 1 \\(t_3 \vee t_4) \wedge (\neg t_3 \vee \neg t_4) &= 1 \\(t_4 \vee t_5) \wedge (\neg t_4 \vee \neg t_5) &= 1\end{aligned}$$

Уравнения полученной системы имеют вид ( $k=1, 2, 3, 4$ ):

$$(t_k \vee t_{k+1}) \wedge (\neg t_k \vee \neg t_{k+1}) = 1$$

Это означает, что из каждых двух переменных  $t_k$  и  $t_{k+1}$  ровно одна равна 1 и ровно одна равна нулю, т.е. эти переменные имеют разные значения. Таким образом, систему можно еще немного упростить и записать ее так:

$$\begin{aligned}\neg(t_1 \equiv t_2) &= 1 \\ \neg(t_2 \equiv t_3) &= 1 \\ \neg(t_3 \equiv t_4) &= 1 \\ \neg(t_4 \equiv t_5) &= 1\end{aligned}$$

Б. Анализ системы.

В любом решении последней системы значения переменных чередуются. Поэтому такая система имеет ровно два решения: 01010 и 10101 (первая цифра – значение переменной  $t_1$ , вторая – значение  $t_2$  и т.д.).

Далее, т.к.

$$t_k = x_{2k-1} \equiv x_{2k}$$

(здесь  $k=1, 2, 3, 4, 5$ ), то каждому значению  $t_k$  соответствуют две пары значений переменных  $x_{2k-1}$  и  $x_{2k}$ . Например,  $t_k = 1$  в двух случаях:  $\{x_{2k-1} = x_{2k} = 1\}$  и  $\{x_{2k-1} = x_{2k} = 0\}$ .

*Этап 2. Подсчет числа решений.*

Каждому из двух решений системы для переменных  $t$  соответствует  $2^5 = 32$  решения исходной системы. Поэтому исходная система имеет  $2 \cdot 32 = 64$  решения.

**Ответ:** 64

*Упражнение.* Выпишите все решения. Это немного утомительно, но полезно.

*Пример 2 (открытый сегмент банка заданий ФИПИ)*

Сколько существует различных наборов значений логических переменных  $x_1, x_2, \dots$ ,

$x_{10}$ , которые удовлетворяют всем перечисленным ниже условиям?

$$\begin{aligned}\neg(x_1 \equiv x_2) \wedge (x_1 \vee x_3) \wedge (\neg x_1 \vee \neg x_3) &= 0 \\ \neg(x_2 \equiv x_3) \wedge (x_2 \vee x_4) \wedge (\neg x_2 \vee \neg x_4) &= 0 \\ \dots \\ \neg(x_8 \equiv x_9) \wedge (x_8 \vee x_{10}) \wedge (\neg x_8 \vee \neg x_{10}) &= 0\end{aligned}$$

В ответе **не нужно** перечислять все различные наборы значений  $x_1, x_2, \dots, x_{10}$ , при которых выполнена данная система равенств. В качестве ответа вам нужно указать количество таких наборов.

Решение состоит из двух этапов. Сначала попытаемся описать, как устроены все наборы значений переменных, удовлетворяющие данной системе. Далее подсчитаем число таких наборов.

*Этап 1. Как устроено множество решений*

А. Предварительный этап – упрощаем уравнения.

Заметим, что выражение  $(a \vee b) \wedge (\neg a \vee \neg b)$  равносильно тому, что ровно одна из переменных  $a$  и  $b$  равна 1, то есть равносильно выражению  $\neg(a \equiv b)$ . Поэтому каждое выражение вида  $(x_k \vee x_{k+2}) \wedge (\neg x_k \vee \neg x_{k+2})$ , где  $k=1, \dots, 8$ , в наших уравнениях можно заменить выражением  $\neg(x_k \equiv x_{k+2})$ .

Таким образом, наша система эквивалентна системе

$$\begin{aligned}\neg(x_1 \equiv x_2) \wedge \neg(x_1 \equiv x_3) &= 0 \\ \neg(x_2 \equiv x_3) \wedge \neg(x_2 \equiv x_4) &= 0 \\ \dots \\ \neg(x_8 \equiv x_9) \wedge \neg(x_8 \equiv x_{10}) &= 0\end{aligned}$$

Далее,  $\neg a \wedge \neg b = 0$  означает, что, если  $\neg a$  истинно, то  $\neg b$  истинным быть не может. Т.е.  $\neg a \wedge \neg b = 0$  эквивалентно  $\neg a \rightarrow b = 1$ .

Поэтому систему можно записать в следующем виде

$$\begin{aligned}\neg(x_1 \equiv x_2) \rightarrow (x_1 \equiv x_3) &= 1 \\ \neg(x_2 \equiv x_3) \rightarrow (x_2 \equiv x_4) &= 1 \\ \dots \\ \neg(x_8 \equiv x_9) \rightarrow (x_8 \equiv x_{10}) &= 1\end{aligned}$$

Б. Анализ системы.

Каждое из уравнений полученной системы имеет вид ( $k = 1, \dots, 8$ ):

$$\neg(x_k \equiv x_{k+1}) \rightarrow (x_k \equiv x_{k+2}) = 1$$

Иными словами, если два соседних элемента набора  $x_k$  и  $x_{k+1}$  не равны между собой, то  $x_k = x_{k+2}$ , то есть элементы  $x_{k+1}$  и  $x_{k+2}$  также не равны между собой. Таким образом, набор удовлетворяет системе, тогда и только тогда, когда он обладает следующими свойствами. В начале набора стоит несколько (может быть, одно) одинаковых значений (назовем это "головой" набора). Затем (после первого появления нового числа) значения в наборе чередуются ("хвост" набора).

Пример решения: 1111010101 (в этой последовательности первая цифра – значение переменной  $x_1$ , вторая цифра – значение переменной  $x_2$ , и т.д.)

Здесь голова набора состоит из четырех единиц, а хвост – это последовательность 01010101. в данном примере длина головы равна 4.

Важное наблюдение. Для каждой непустой головы есть ровно один хвост, образующий вместе с ней решение. Действительно, первая цифра такого хвоста – это цифра, противоположная цифрам головы. А дальше цифры в хвосте чередуются.

*Этап 2. Подсчет числа решений*

В соответствии с важным наблюдением, количество решений совпадает с

количеством возможных голов. Очевидно, существует 10 голов, состоящих из единиц (1, 11, 111, ..., 1111111111) и столько же голов, состоящих из нулей.

**Ответ:** 20.

*Замечание.* Как видим, сложность решения задачи не зависит от числа переменных и уравнений. Если понятно, как устроено множество решений, подсчитать количество решений для аналогичной системы, скажем, с 20-ю переменными, не сложнее, чем в уже рассмотренном случае.

*Рекомендации для учителей и учеников*

Разбирать эту задачу стоит только с учениками, которые достаточно свободно владеют преобразованиями логических выражений. Отметим несколько полезных преобразований (они встречались в разобранных примерах):

$$\neg a \vee b \quad \text{равносильно} \quad a \rightarrow b$$

$$(a \rightarrow b) \wedge (a \rightarrow b) \quad \text{равносильно} \quad a \equiv b$$

$$(\neg a \vee b) \wedge (a \vee \neg b) \quad \text{равносильно} \quad a \equiv b$$

$$(a \vee b) \wedge (\neg a \vee \neg b) \quad \text{равносильно} \quad \neg(a \equiv b).$$

Кроме того, полезно потренироваться в выполнении замен в логических выражениях. Отметим, что это делается точно так же, как и замены в уравнениях, которые встречаются в курсе математики.

Первая цель при выполнении задания В15 - понять, что собой представляет множество решений системы. Для этого систему бывает полезно преобразовать (упростить) систему, используя тождественные преобразования и замены переменных. Затем – подсчитать количество элементов во множестве решений.

Во многих случаях система состоит из однотипных уравнений, каждое из которых связывает небольшое число переменных (две-три-четыре), притом что в системе может быть 10 и более переменных. Обычно, количество переменных не является источником сложности, оно является параметром решения. Если не получается решить задачу в общем виде, можно попробовать перебрать все решения для системы с небольшим количеством переменных. Это может подсказать, как выглядит решение в общем виде.

Если понятно, как выглядит множество решений, подсчет их количества – несложная комбинаторная задача. Сильные ученики могут сообразить, как провести подсчет, даже не обладая специальными знаниями. Стоит повторить формулы произведения возможностей и формулу суммы арифметической прогрессии.

Выше разобрано несколько примеров. Другие полезные примеры и рекомендации можно найти на сайтах, указанных выше.

Таким образом, план подготовки может быть примерно таким.

1) Повторить логические преобразования и элементы комбинаторики.

2) Порешать задачи и попрактиковаться в переводе формального описания, в виде системы логических условий на обычный, "человеческий" язык.

В заключение следует отметить, что год от года результаты выполнения ЕГЭ по информатике и ИКТ экзаменуемыми средней и сильной групп подготовки растут. Это показывает, что хорошая подготовка к экзамену, тренировка в решении экзаменационных заданий различных типов – верный путь к успеху на экзамене.