

**РЕЗУЛЬТАТЫ  
ЕДИНОГО ГОСУДАРСТВЕННОГО ЭКЗАМЕНА  
ПО ИНФОРМАТИКЕ И ИКТ  
ВЫПУСКНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ  
УЧРЕЖДЕНИЙ АЛТАЙСКОГО КРАЯ 2014 г.**

**Аналитический отчет  
предметной комиссии**

**Барнаул  
2014**

**Результаты** единого государственного экзамена по информатике и ИКТ выпускников общеобразовательных учреждений Алтайского края 2014 г. : аналитический отчет предметной комиссии. – Барнаул, 2014. – 111 с.

**Составитель:**

*М.В. Афонина*, председатель Алтайской краевой предметной комиссии ЕГЭ по ИНФОРМАТИКЕ и ИКТ, заведующий кафедрой теоретических основ информатики Алтайской государственной педагогической академии (ФГБОУ ВПО «АлтГПА»), кандидат педагогических наук

В отчёте представлен анализ информации, полученной в период подготовки и проведения Единого государственного экзамена в Алтайском крае в 2014 году, а также методические рекомендации для учителей общеобразовательных учреждений по устранению и предотвращению типичных ошибок и недочётов при подготовке обучающихся к экзамену по информатике и ИКТ в форме ЕГЭ.

Содержащаяся в отчёте информация предназначена для широкого круга специалистов в сфере образования. Методические рекомендации в первую очередь ориентированы на учителей информатики и ИКТ общеобразовательных учреждений, могут быть использованы учащимися старших классов при изучении информатики и подготовке к ЕГЭ по информатике и ИКТ.

Статистическая информация предоставлена сотрудниками Регионального центра обработки информации Единого государственного экзамена в Алтайском крае.

© М.В. Афонина. 2014  
© Алтайская государственная педагогическая академия. 2014  
© Главное управление образования и молодёжной политики Алтайского края. 2014

## Содержание

Введение .....	4
1. Численность и состав предметной комиссии .....	5
2. Обеспеченность нормативной и методической документацией по организации деятельности предметной комиссии .....	12
3. Взаимодействие с Главным управлением образования и молодёжной политики Алтайского края и РЦОИ ЕГЭ .....	13
4. Характеристика участников ЕГЭ по информатике и ИКТ 2014 года .....	14
5. Основные результаты экзамена по информатике и ИКТ 2014 года .....	16
6. Результаты выполнения заданий по основным содержательным разделам учебного предмета «Информатика и ИКТ» .....	19
6.1. Анализ выполнения части А .....	26
6.2. Анализ выполнения части В .....	40
6.3. Анализ выполнения части С .....	64
7. Типичные ошибки, допущенные участниками экзамена при выполнении заданий третьей части (С) .....	66
7.1. Задание С1 .....	66
7.2. Задание С2 .....	71
7.3. Задание С3 .....	80
7.4. Задание С4 .....	92
8. Влияние на результаты экзамена в форме ЕГЭ используемого учебника .....	102
9. Сведения о работе конфликтной комиссии .....	103
10. Общие выводы и рекомендации .....	104
Литература .....	110
Информационные ресурсы .....	110

## Введение

Единый государственный экзамен (ЕГЭ) является единственной формой объективной оценки качества подготовки лиц, освоивших образовательные программы среднего (полного) общего образования. Выполнение учащимися заданий контрольных измерительных материалов ЕГЭ позволяет установить уровень освоения ими федерального государственного образовательного стандарта.

Результаты единого государственного экзамена признаются образовательными учреждениями, в которых реализуются образовательные программы среднего (полного) общего образования, как результаты государственной (итоговой) аттестации, а образовательными учреждениями среднего профессионального образования и образовательными учреждениями высшего профессионального образования как результаты вступительных испытаний по соответствующим общеобразовательным предметам. Таким образом, ЕГЭ является основной формой вступительных экзаменов в большинство российских вузов.

Несмотря на то, что информатика и ИКТ не является обязательной дисциплиной по которой проводится государственная итоговая аттестация, приходится учитывать то, что в соответствии с приказом Минобрнауки России от 28.10.2009 N 505 (ред. от 13.12.2012) информатика и ИКТ входит в перечень вступительных испытаний для 129 специальностей высшего образования. Кроме того, IT-специалисты в настоящее время пользуются спросом на рынке труда, а профессии, связанные с IT-технологиями являются наиболее престижными. Поэтому число желающих среди учащихся средних школ сдавать ЕГЭ по информатике и ИКТ с каждым годом растет.

Прошедший в 2014 г. ЕГЭ по информатике и ИКТ выявил ряд тенденций в развитии в Алтайском крае среднего образования в области информатики и ИКТ, а так же типичные погрешности учащихся при выполнении заданий. Выявление недочетов в образовании в области информатики и ИКТ и проектирование действий по их предупреждению являются актуальными задачами для учреждений общего образования.

Предлагаемый вниманию читателей отчет содержит анализ информации, полученной в период подготовки и проведения Единого государственного экзамена в Алтайском крае в 2014 году, а также методические рекомендации для учителей общеобразовательных учреждений по устранению и предотвращению типичных ошибок и недочетов при подготовке обучающихся к экзамену по информатике и ИКТ в форме ЕГЭ. В данной публикации использованы материалы, накопленные в ходе работы Алтайской краевой предметной комиссии ЕГЭ по информатике и ИКТ.

Содержащаяся в отчете информация предназначена для широкого круга специалистов в сфере образования. Методические рекомендации в первую очередь ориентированы на учителей информатики и ИКТ общеобразовательных учреждений, могут быть использованы учащимися старших классов при изучении информатики и подготовке к ЕГЭ по информатике и ИКТ.

Статистическая информация предоставлена сотрудниками Регионального центра обработки информации ЕГЭ в Алтайском крае.

## 1. Численность и состав предметной комиссии

Одной из важнейших составляющих процедуры ЕГЭ является оценивание экзаменационных работ участников экзамена, осуществляемое региональными предметными комиссиями ЕГЭ (далее – РПК) по проверке заданий с развернутыми ответами. От результатов работы РПК в огромной степени зависит обеспечение принципов справедливого оценивания. Именно поэтому представляется крайне важным добиться максимальной согласованности в подходах к оцениванию экзаменационных работ, минимизировать вероятность несправедливого оценивания.

В соответствии с действующей процедурой ЕГЭ оценивание работ происходит во время проверки работ (первая, вторая, третья проверки). Кроме того, итоговая оценка может измениться в процессе рассмотрения апелляций на результаты оценивания, т.е. в процессе работы региональной конфликтной комиссии (РКК). Таким образом, любые механизмы совершенствования работы РПК должны быть направлены на оптимизацию всего комплекса процедур, включая рассмотрение апелляций.

Состав предметной комиссии (подкомиссии) ГЭК по информатике и ИКТ утверждён приказом Главного Управления образования и молодежной политики Алтайского края № 1759 от 19.03.2014 и насчитывает 29 человек, включая председателя комиссии (кандидат педагогических наук, заведующий кафедрой теоретических основ информатики, доцент Алтайской государственной педагогической академии М.В. Афонина) и заместителя председателя (старший преподаватель кафедры теоретических основ информатики Алтайской государственной педагогической академии Е.И. Апольских).

Председатель предметной комиссии в январе 2014 года принял участие в работе очных семинаров по согласованию подходов к оцениванию работ ЕГЭ, предназначенных для экспертов, претендующих на позиции руководителей региональных предметных комиссий по информатике и ИКТ 2014 году, организатором которых явилось федеральное государственное бюджетное научное учреждение «Федеральный институт педагогических измерений». Семинары проводились при поддержке Федеральной службы по надзору в сфере образования и науки. Обсуждённые на семинаре проблемы совершенствования системы оценки качества образования и пути их решения привели к обновлению содержания мероприятий по повышению квалификации экспертов и учителей информатики и ИКТ общеобразовательных учреждений.

Председатель предметной комиссии принял участие в ряде проводимых мероприятий Российского, а также краевого значения, направленных на решение задач совершенствования системы оценки качества образования в области информатики и ИКТ, в том числе следующих:

- XII Всероссийская конференция «Преподавание информационных технологий в Российской Федерации» (Организаторы: высшая школа Информационных Технологий и Информационных Систем Казанского феде-

рального университета (ИТИС КФУ), при поддержке Правительства Республики Татарстан, Министерства образования и науки РФ, Министерства связи и массовых коммуникаций РФ, Российского Союза ректоров; форма участия: доклад «Подготовка учащихся средней школы к ЕГЭ по информатике и ИКТ»; период проведения: 15.05.2014 - 16.05.2014)

- IV научно-практическая конференция краевых профессиональных объединений педагогов «Профессиональные сообщества педагогов как ресурс развития системы образования края» (Организаторы: Главное управление образования и молодежной политики Алтайского края, ФГБОУ ВПО «АлтГПА», АК ИПКРО, форма участия: выступление с докладом «Перспективы развития системы проверки итоговых знаний учащихся по информатике», период проведения: 7 ноября 2013);

- Краевой вебинар «Подготовка выпускников общеобразовательных учреждений к государственной итоговой аттестации по информатике и ИКТ в форме ЕГЭ» 25.02.2014 г. при поддержке Главного управления образования и молодежной политики Алтайского края, АК ИПКРО (форма участия: выступление с лекцией, проведение семинара);

- Курсы повышения квалификации членов предметных комиссий ЕГЭ по информатике на базе ФГБОУ ВПО «Алтайская государственная педагогическая академия» (13-31 марта 2014 г., приказ Главного управления образования и молодежной политики Алтайского края №1523 от 12.03.2014);

- Совещание с председателями предметных комиссий ЕГЭ по общеобразовательным предметам по вопросу совершенствования технологий проведения ЕГЭ в Алтайском крае в 2014 году (Главное управление образования и молодежной политики Алтайского, 6 марта 2014 г.).

К экспертам, привлекаемым для работы в составе РПК по информатике и ИКТ, предъявлялись требования, которые были выдержаны для всех членов РПК:

- наличие высшего образования;
- соответствие квалификационным требованиям, указанным в квалификационных справочниках, и (или) профессиональных стандартах;
- наличие опыта работы в организациях, осуществляющих образовательную деятельность и реализующих образовательные программы среднего общего, среднего профессионального или высшего образования (не менее 5 лет);

- наличие документа, подтверждающего получение дополнительного профессионального образования, включающего в себя практические занятия по оцениванию образцов экзаменационных работ в соответствии с критериями оценивания по соответствующему учебному предмету, определяемыми Рособрнадзором.

В состав РПК по информатике и ИКТ включены опытные учителя общеобразовательных учреждений и учреждения НПО (13 человек), преподаватели вузов (16 человек).

Несмотря на стабильность состава экспертного корпуса, необходимость ежегодного обучения экспертов и повышения их квалификации не теряет актуальности и вызвана рядом причин, основными среди которых являются:

- 1) изменения в структуре и содержании ЕГЭ по информатике и ИКТ;
- 2) содержательное совершенствование критериев оценивания заданий части С, требующих развёрнутых ответов от участников экзамена;
- 3) необходимость учёта опыта работы экспертов в предыдущие годы, обобщения позитивного опыта работы экспертов РПК Алтайского края и других регионов;
- 4) обязательное присвоение экспертам статусов: ведущий, старший, основной;
- 5) целесообразность детального анализа результатов выполнения учащимися экзаменационной работы в предыдущем году с целью определения предупреждающих и корректирующих мероприятий по преодолению типичных ошибок участников экзамена.

Для подготовки экспертов предметной комиссии по информатике и ИКТ применялся электронный курс «Экспертная оценка заданий ЕГЭ по информатике с развернутой формой ответов», разработанный в среде управления электронным обучением Moodle (авторы М.В. Афолина и Е.И. Апольских) с использованием методических материалов Федерального института педагогических измерений (г. Москва) и материалов, полученных в период проведения ЕГЭ по информатике и ИКТ в Алтайском крае в 2013 году. Кроме того, обучающиеся использовали тренинги и тесты интернет-системы дистанционной подготовки экспертов «Эксперт-ЕГЭ».

Основной целью подготовки явилось: формирование и развитие профессиональной компетентности специалистов в области проверки и оценки выполнения заданий с развернутым ответом экзаменационных работ ЕГЭ.

Используемые в подготовке ресурсы позволяют автоматизировано решать следующие виды задач:

- проведение дистанционной подготовки с использованием заранее подготовленной информации о составе участников и учебных материалов;
- обеспечение авторизации участников;
- контроль над ходом проведения дистанционной подготовки;
- подготовка информации о результатах дистанционной подготовки.

Обучение экспертов приёмам оценивания ответов учащихся, включая знакомство с критериями оценивания, прохождение различных тренингов по оцениванию ответов учащихся, сравнение с эталонными оценками осуществлено в марте-апреле 2014 г.

По результатам прохождения экспертами итогового тестирования в электронном курсе из 29 человек, входящих в комиссию, статус ведущего эксперта присвоен 3 соискателям, 20 человек получили статус старшего эксперта, а 6 человек – статус основного эксперта.

Ведущие эксперты вошли в состав предметной комиссии по информатике и информационно-коммуникационным технологиям, создаваемой Рособнадзором на 2014 год (приказ №609 от 30.04.2014).

Для обеспечения результативности, эффективности и гибкости процесса оценивания на основном этапе проведения ЕГЭ по информатике и ИКТ было проведено инструктивное занятие в первый день проверки экзаменационных работ, которое решало следующие учебные задачи:

- способствовать формированию у экспертов знаний о содержании нормативных документов, регламентирующих разработку контрольных измерительных материалов ЕГЭ, процедуру проверки и оценки ответов выпускников на задания с развернутым ответом экзаменационных работ ЕГЭ по информатике и ИКТ, права и обязанности экспертов;
- актуализировать умения экспертов: работать с инструкциями, регламентирующими процедуру проверки и оценки ответов выпускников на задания с развернутым ответом; проверять и объективно оценивать ответы выпускников на задания с развернутым ответом; вести рабочую документацию эксперта.

В ходе инструктивного занятия проходил обмен мнениями экспертов по проблемам оценивания экзаменационных работ ЕГЭ, способствовавший выработке единых подходов к проверке и оцениванию решений задач с развернутым ответом, достижению однозначности толкования требований критериев оценивания, предотвращению разночтений при проверке решений, осуществлённых неотражёнными в критериях оценивания способами.

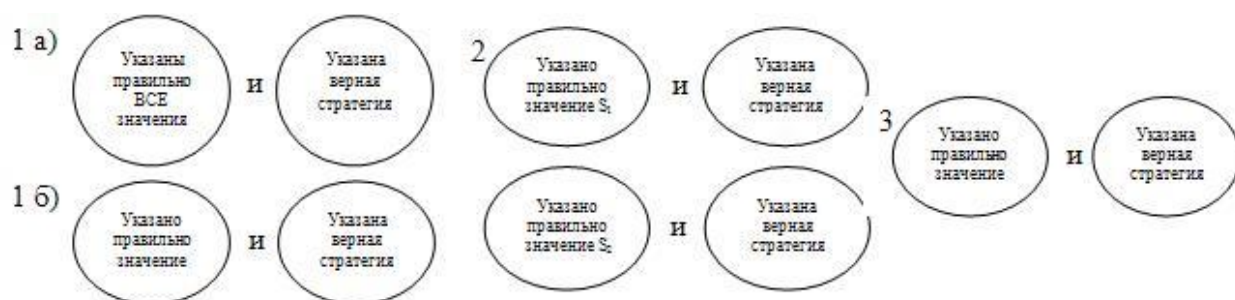
Экспертам были предложены памятки и критерии оценивания работ. Например, для упрощения процедуры определения баллов за задания С1 и С3 по соответствующим критериям были предложены таблица и схемы представленные на рисунках 1-4.

Верно выполненных действий	Нет ложно названных ошибок	1 ложная ошибка	2 и > ложных ошибок
4	3	2	1
3	2	1	1
2	1	1	1
1	0	0	0
0	0	0	0

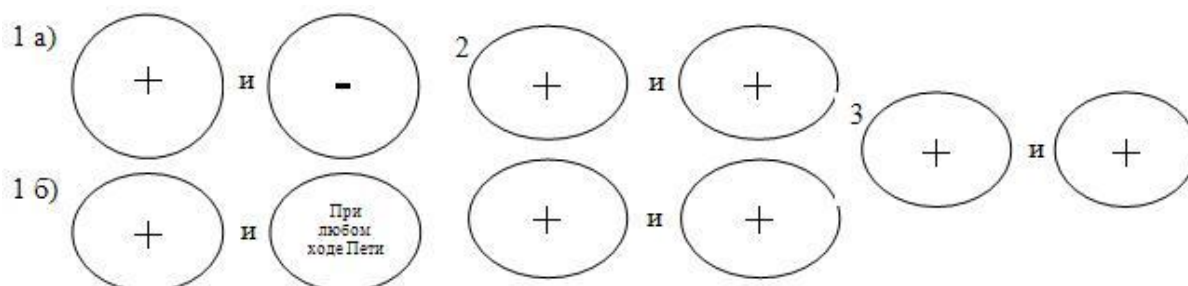
Баллы

**Рисунок 1.** Система выбора баллов при оценивании задания С1

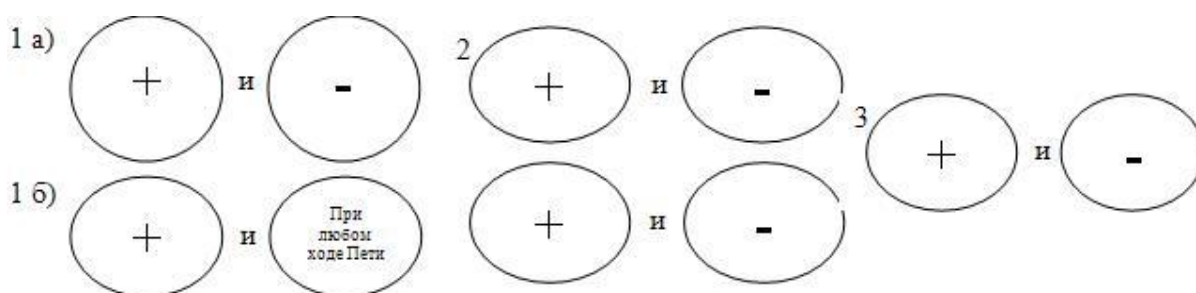




**Рисунок 2.** Методика оценивания задания С3



**Рисунок 3.** Схема 3-х балльной работы С3 при ЧАСТИЧНО выполненном задании 1



**Рисунок 4.** Схема работы С3 на 2 балла

Организация управления предметной комиссией осуществлялась председателем и его заместителем. Председателем комиссии М.В. Афонинной выполнялась работа по уточнению состава комиссии, организации очно-заочного и дистанционного обучения экспертов ЕГЭ и подготовке методических материалов к ней, составлению ведомостей, связи с РЦОИ, связи с Главным управлением образования и молодежной политики Алтайского края, связи с учителями общеобразовательных учреждений и преподавателями вузов – экспертами ЕГЭ, подготовке итогового отчёта о работе.

Заместитель председателя предметной комиссии Е.И. Апольских участвовала в организации учёбы экспертов; в разработке электронного учебного курса «Экспертная оценка заданий ЕГЭ по информатике с развернутой формой ответов» в среде управления электронным обучением Moodle; во взаимодействии с РЦОИ, анализе результатов ЕГЭ.

Работа комиссии на основном этапе ЕГЭ организовывалась 10 июня (день сдачи ЕГЭ по информатике и ИКТ – 9 июня). На проверку явились 15 приглашённых экспертов – члены РПК по проверке выполнения заданий с развернутым ответом экзаменационных работ ЕГЭ по информатике и ИКТ. Из 14 участников проверки 9 – преподаватели вузов (из которых 4 – препода-

ватели Алтайской государственной педагогической академии, 5 – Алтайского государственного университета), 5 – учителя информатики и ИКТ общеобразовательных учреждений.

Председатель предметной комиссии осуществлял консультационную поддержку и помощь при затруднениях членов комиссии в ходе проверки.

Перед проведением третьей проверки экзаменационных работ председателем предметной комиссии был проведён инструктаж, основной задачей которого являлась выработка согласованных подходов к оцениванию «нестандартных» решений участников экзамена, разрешение спорных ситуаций оценивания работ участников экзамена.

Стоит отметить невысокий процент работ выпускников 2014 года, ушедших на третью проверку – 5,97%. Это показывает высокий уровень согласованности подходов к оцениванию работ.

В процессе работы экспертов **выявлены следующие проблемы**, на которые, считаю, необходимо обратить внимание организаторам ГИА и председателю ГЭК в регионе:

1. Использование учащимся языков и сред программирования, а также процедур и функций из различных библиотек выбранной среды программирования при написании программы никак не ограничено, а следовательно при проверке работы может оказаться, что эксперт не владеет данным языком или не сталкивался с используемой экзаменуемым процедурой (функцией, оператором) и не имеет возможности объективно и компетентно оценить работу. При этом пакеты работ распределены строго по экспертам, и передать работу на проверку другому эксперту не представляется возможным. Это серьезным образом накладывает определенные обязанности на председателя предметной комиссии (его заместителя), которые выполняют функции консультанта во время проверки. Так же это увеличивает требования к ним по владению абсолютно всеми языками и средами программирования, равно как и методами. Кроме того, при формировании РПК председатель вынужден вести поиск экспертов-универсалов в этом плане, но сформировать комиссию только из таких экспертов практически нереализуемо. Не запрещено назначать во время работы комиссии консультантов по отдельным вопросам, не являющихся председателями или заместителями председателей предметных комиссий, но **вопросы оплаты** такого вида деятельности во время работы предметной комиссии **не решены**.

2. При выполнении задания на написание программы (составление алгоритма), например, в задании С4, учащийся может выбрать абсолютно любой метод решения задачи, оптимальный или нет. При этом длину программы (алгоритма), его сложность и запутанность предвидеть нельзя. Но, учитывая, что в критериях на один балл (задание С4) достаточно увидеть здоровое зерно и понимание учащимся, каким должен быть ход решения (его этапы), то эксперт должен досконально разбираться в написанном тексте, каким бы запутанным он ни был, даже если понятно, что задача правильно экзаменуемым не решена и программа не работает. Зачастую эта работа занимает много времени (достигало до 45 минут на проверку одной работы). Счи-

таю, что в таком случае, необходимо продумать **дифференцированную систему оплаты** для проверки работ экспертами по разным предметам. Стоит, например, обратить внимание на тот факт, что часто члены предметной комиссии по информатике и ИКТ являются так же экспертами по математике или физике (учителя математики и информатики, учителя физики и информатики, преподаватели кафедр прикладной математики и программирования и пр.). Существуют случаи, что один и тот же специалист за шести часовую работу в предметной комиссии по математике проверяет 200-300 работ, а за то же время работы в предметной комиссии по информатике и ИКТ успевает проверить не более 80 работ. При этом оплата за проверку одной работы одинакова. Таким образом, у председателя предметной комиссии возникает проблема для стимулирования участия в работе предметной комиссии по информатике и ИКТ опытных специалистов.

3. В резервное время и во «вторую» волну по информатике и ИКТ сдают ЕГЭ не более десятка человек, при этом на проверку заданий третьей части эксперт может получить всего одну или несколько работ, а экспертов при этом должно быть не менее двух. Таким образом, учитывая мизерную стоимость проверки одной работы, в летнее, отпускное время эксперты тратят несколько часов (учитывая дорогу, проверку, заполнение протоколов) на работу, которая фактически не оплачивается. Стоит в таких случаях оплачивать не одну работу, а **время работы эксперта**, например как участие в работе конфликтной комиссии и оплата при этом должна быть ощутимой!

## **2. Обеспеченность нормативной и методической документацией по организации деятельности предметной комиссии**

На федеральном уровне все нормативные и методические материалы, связанные с ЕГЭ, располагаются на сайте Федерального института педагогических измерений <http://www.fipi.ru> и доступны для использования.

На региональном уровне нормативные акты, методические материалы, дополнительные сведения сконцентрированы на сайте информационной поддержки государственной итоговой аттестации в Алтайском крае <http://ege.edu22.info/>.

Для слаженной работы всех членов комиссии и отработки единых подходов к оцениванию творческих работ учащихся предназначены рекомендуемые экспертам учебно-методические материалы для председателей и членов региональных предметных комиссий по проверке выполнения заданий с развернутым ответом экзаменационных работ ЕГЭ 2014 года «Методические рекомендации по оцениванию выполнения заданий единого государственного экзамена с развёрнутым ответом: Информатика и ИКТ», разработанных Федеральным институтом педагогических измерений. Использование указанных материалов способствует: пониманию концептуальной основы ЕГЭ - 2014; обеспечению единства принципов и согласованности действий при оценивании экзаменационных работ выпускников школы по информатике и ИКТ; уяснению и осмыслению критериев оценивания заданий с развернутым ответом; обеспечению максимальной однородности результатов проверки; совершенствованию методики подготовки экспертов ЕГЭ.

Для обеспечения работы предметной комиссии по информатике и ИКТ в день проведения экзамена на основном этапе ЕГЭ председателем предметной комиссии, его заместителями с привлечением опытных экспертов предметной комиссии были детально проанализированы критерии оценивания экзаменационных работ и подготовлены оригинал-макеты методических материалов для осуществления оценочной деятельности каждого эксперта. Подготовленные методические материалы позволили исключить дублирование информации, способствовали созданию комфортных условий для результативной работы экспертов, что явилось одним из факторов повышения качества оценивания работ участников экзамена.

С целью выявления типичных погрешностей участников экзамена при решении задач третьей части экзаменационной работы были разработаны задания и рекомендации для экспертов по осуществлению анализа работ участников экзамена, подготовки выводов по осуществлению предупреждающих и корректирующих действий. Результаты такого рода аналитической деятельности экспертов нашли отражения в настоящем отчете.

### **3. Взаимодействие с Главным управлением образования и молодёжной политики Алтайского края и РЦОИ ЕГЭ**

В течение года основные вопросы и проблемы ЕГЭ оперативно решаются совместно с Главным управлением образования и молодёжной политики Алтайского края. Взаимодействие со специалистами, занимающимися вопросами ЕГЭ, позволяет своевременно решать возникающие проблемы, получать от них компетентные разъяснения и рекомендации по осуществлению деятельности, необходимую административную и методическую поддержку. Систематично Главным управлением образования и молодёжной политики Алтайского края проводятся совещания председателей предметных комиссий. Текущее взаимодействие осуществляется по электронной почте, специалисты оперативно и своевременно отвечают на вопросы и организуют методическую помощь.

Отлаженное взаимодействие с отделом дошкольного и общего образования Главного управления образования и молодёжной политики Алтайского края (начальник отдела – Дроздова Ирина Николаевна) позволило оперативно решать организационно-методические вопросы, актуальные в текущем году и касающиеся подготовки экспертов предметной комиссии, планирования и сопровождения на всех этапах деятельности членов предметной комиссии.

Следует отметить также хорошо организованную и чёткую деятельность Регионального центра по обработке информации Единого государственного экзамена, что особенно важно при проведении процедуры проверки экзаменационных работ выпускников. Сотрудники центра согласовывают график удобный для работы членов комиссии; своевременно готовят и предоставляют в распоряжение предметной комиссии все необходимые материалы для проверки; необходимую статистическую информацию по результатам ЕГЭ по информатике и ИКТ; оперативно обрабатывают предоставляемую информацию. Заседания предметной комиссии проходили в обстановке взаимопонимания со стороны РЦОИ и представителей Главного управления образования и молодёжной политики Алтайского края.

Заседания конфликтной комиссии проходили в обстановке доброжелательности по отношению к лицам, подавшим апелляции, при взаимопонимании экспертов предметной комиссии, представителей Главного управления образования и молодёжной политики Алтайского края, общеобразовательных учреждений Алтайского края. Никаких эксцессов и нестандартных ситуаций за время рассмотрения конфликтной комиссией апелляций по работам по информатике и ИКТ не произошло.

Характеризуя взаимодействие с Главным управлением образования и молодёжной политики Алтайского края и РЦОИ ЕГЭ в целом, следует отметить результативность, эффективность и гибкость в осуществлении процессов нормативного, организационного, технологического и методического обеспечения процедур, связанных с ЕГЭ по информатике и ИКТ.

#### 4. Характеристика участников ЕГЭ по информатике и ИКТ 2014 года

Единый государственный экзамен по информатике и ИКТ в июне 2014 года (основной этап) из 11950 выпускников общеобразовательных учреждений в Алтайском крае сдавали 519 человек, что составляет 4,34%. В 2013 году сдавали 515 выпускников из 12708, или 4,05%. Таким образом, наблюдается рост доли выпускников, выбравших информатику и ИКТ в качестве дополнительного предмета для прохождения итоговой аттестации.

Статистические данные по распределению основных показателей ЕГЭ по информатике и ИКТ в июне 2014 по районам представлены в таблице 1.

*Таблица 1*

##### Средний балл и распределение оценок ЕГЭ по ОУ Алтайского края

Район	Число участников	Средний балл	Мин. балл	Макс. балл	Не пре-одолели минимум (чел.)	Не пре-одолели минимум (%)
<b>Алтайский край</b>	<b>519</b>	<b>58,96</b>	<b>5</b>	<b>100</b>	<b>45</b>	<b>8,67</b>
Алейский район	1	49,00	49	49	0	0,00
Алтайский район	1	30,00	30	30	1	100,00
Бийский район	1	65,00	65	65	0	0,00
Благовещенский район	2	61,00	54	68	0	0,00
Бурлинский район	5	53,60	30	65	1	20,00
Быстроистокский район	1	81,00	81	81	0	0,00
Волчихинский район	1	49,00	49	49	0	0,00
Егорьевский район	5	62,80	49	75	0	0,00
Завьяловский район	3	62,00	45	81	0	0,00
Залесовский район	3	43,67	40	49	0	0,00
Змеиногорский район	4	43,25	40	47	0	0,00
Заринский район	1	42,00	42	42	0	0,00
Калманский район	3	37,00	10	54	1	33,33
Косихинский район	1	30,00	30	30	1	100,00
Красногорский район	5	40,80	25	47	1	20,00
Кулундинский район	1	52,00	52	52	0	0,00
Курьинский район	2	52,50	50	55	0	0,00
Локтевский район	2	60,00	57	63	0	0,00
Мамонтовский район	6	65,00	52	78	0	0,00
Михайловский район	14	55,57	25	78	1	7,14
Немецкий национальный район	2	47,00	47	47	0	0,00
Новичихинский район	1	42,00	42	42	0	0,00
Павловский район	1	67,00	67	67	0	0,00
Панкрушихинский район	2	52,00	47	57	0	0,00
Петропавловский район	2	48,00	42	54	0	0,00
Поспелихинский район	5	53,20	35	70	1	20,00
Рубцовский район	2	52,00	49	55	0	0,00
ЗАТО Сибирский	3	51,33	35	62	1	33,33
Смоленский район	1	44,00	44	44	0	0,00
Советский район	5	40,40	5	57	1	20,00
Солонешенский район	3	47,67	42	52	0	0,00

Район	Число участников	Средний балл	Мин. балл	Макс. балл	Не пре-одолели минимум (чел.)	Не пре-одолели минимум (%)
Солтонский район	3	61,00	52	76	0	0,00
Табунский район	1	55,00	55	55	0	0,00
Тальменский район	3	51,67	45	60	0	0,00
Тогульский район	1	10,00	10	10	1	100,00
Топчихинский район	2	31,00	20	42	1	50,00
Третьяковский район	4	61,25	50	78	0	0,00
Троицкий район	1	63,00	63	63	0	0,00
Угловский район	1	76,00	76	76	0	0,00
Усть-Пристанский район	4	55,75	44	62	0	0,00
Хабарский район	2	36,00	30	42	1	50,00
Целинный район	1	20,00	20	20	1	100,00
Чарышский район	1	57,00	57	57	0	0,00
Шипуновский район	7	57,00	35	80	1	14,29
Шелаболихинский район	1	15,00	15	15	1	100,00
г. Алейск	9	60,56	40	88	0	0,00
г. Барнаул	210	64,44	20	100	11	5,24
г. Белокураха	5	57,40	45	65	0	0,00
г. Бийск	64	55,48	20	80	8	12,50
г. Заринск	11	58,18	30	88	1	9,09
г. Камень-на-Оби	15	48,20	10	76	3	20,00
г. Новоалтайск	10	44,30	20	63	2	20,00
г. Рубцовск	37	55,86	5	81	5	13,51
г. Славгород	4	53,00	45	70	0	0,00
г. Яровое	1	60,00	60	60	0	0,00
Краевые ОУ	31	71,10	47	100	0	0,00
Негосударственные ОУ	1	49,00	49	49	0	0,00

Традиционно, наиболее массовый характер выбора информатики и ИКТ для сдачи ЕГЭ показывают выпускники ОУ г. Барнаула, г. Бийска, г. Рубцовска. Однако, если в г. Барнауле наблюдается рост популярности ЕГЭ по информатике и ИКТ (210 участников в 2014 г. и 144 – в 2013 г.), то в г. Бийске (64 участника в 2014 г. и 74 – в 2013 г.) и г. Рубцовске (37 участников в 2014 г. и 62 – в 2013 г.) констатируется спад. Кроме того, в г. Рубцовске в прошлом году участники ЕГЭ показали баллы выше среднего по Алтайскому краю ( $75,35 > 65,12$ ), а в текущем – баллы заметно ниже среднего по региону ( $55,86 < 58,96$ ).

Учитывая статистику, необходимо принять меры, позволяющие в 2015 году популяризировать среди выпускников ОУ среднего образования Алтайского края предмет «Информатика и ИКТ» в качестве дополнительного для сдачи ЕГЭ. Особенно актуальным это является в связи с принятием в РФ «Стратегии развития отрасли информационных технологий в Российской Федерации на 2014-2020 год и на перспективу до 2025 года» и «Концепции развития математического образования в Российской Федерации».

## 5. Основные результаты экзамена по информатике и ИКТ 2014 года

Количество выпускников, не набравших минимальный балл (40 баллов), составило 45 человек, или 8,67%. В сравнении с 2013 годом (26 выпускников, 5,04%) наблюдаем существенное увеличение данного показателя.

Следует отметить, что средний балл ЕГЭ по информатике и ИКТ в 2014 году равен 58,96, тогда как в 2013 году средний балл был равен 65,12. Так видим значительное снижение среднего балла, при этом доля выпускников, набравших баллы выше среднего значения в текущем году, не возросла, а напротив, снизилась (264, или 50,87%, в 2013 году – 264, или 51,26%).

Основные результаты экзамена в сравнении за последние 2 года представлены в таблице 2.

Таблица 2

### Сравнение показателей результатов ЕГЭ по информатике и ИКТ выпускников ОУ Алтайского края

Год	Доля выпускников текущего года в составе сдававших ЕГЭ, чел. / %	Доля участников, не набравших мин. кол-во баллов ЕГЭ, чел. / %	100-балльников, чел. / %	Средний тестовый балл
2013	515 / 4,05	26 / 5,04	5	65,12
2014	519 / 4,34	45 / 8,67	3	58,96

В таблице 3 представлено распределение количества участников ЕГЭ по информатике и ИКТ 2014 года по тестовым баллам.

Таблица 3

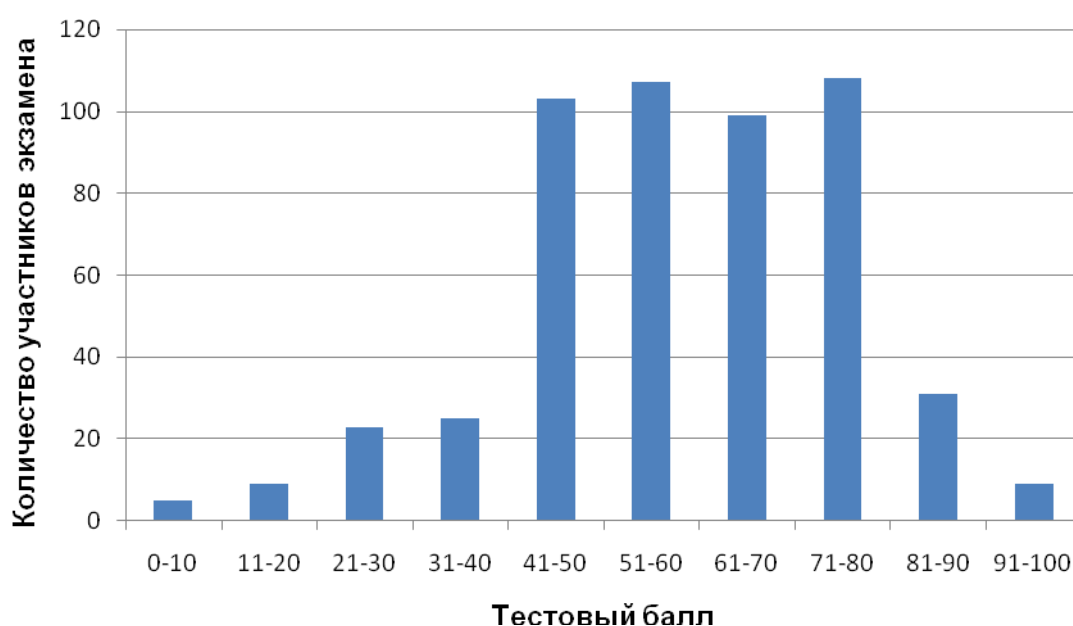
### Распределение количества участников ЕГЭ 2014 года (информатика и ИКТ) по полученным тестовым баллам

Баллы	Количество участников
5	2
10	3
15	1
20	8
25	7
30	16
35	8
40	17
42	18
44	11
45	25
47	19
49	17
50	13
52	14
54	14
55	22
57	24
58	16
60	17



Баллы	Количество участников
62	19
63	22
65	14
67	17
68	13
70	14
71	28
73	15
75	16
76	20
78	16
80	13
81	11
83	11
84	2
88	7
91	3
94	1
97	2
100	3

На рисунке 5 наглядно представлено распределение количества участников ЕГЭ по шкале тестовых баллов.



**Рисунок 5.** Распределение количества участников ЕГЭ по шкале тестовых баллов

Как видно на диаграмме, распределение по шкале тестовых баллов аппроксимирует к нормальному распределению, с некоторым смещением вправо, что говорит о приемлемом уровне выполнения заданий. Основная доля выпускников находится в зоне от 41 до 80 баллов, что характеризует степень подготовленности учащихся как удовлетворительную.

Итак, что доля выпускников, выбравших информатику и ИКТ для сдачи ЕГЭ увеличилась, то есть желание учащихся выявить итоговый результат подготовки по информатике растет, и в целом результаты демонстрируют

удовлетворительный уровень подготовки учащихся. Однако, наблюдается отрицательная динамика изменения основных показателей ЕГЭ по информатике и ИКТ, несмотря на то, что отсутствуют существенные изменения в содержании ЕГЭ-2014 по сравнению с прошлогодним. Стабилизация содержания экзаменационной работы должна была способствовать осуществлению планомерной подготовки участников экзамена. Таким образом, напрашивается вывод об отсутствии системы такой подготовки в Алтайском крае и требуется проведение работы по повышению качества подготовки учащихся по информатике и ИКТ и методической подготовки учителей информатики и ИКТ.

Некоторым образом изменение показателей в отрицательную сторону можно оправдать тем, что в 2013 году мы имели дело с «Утечкой информации» в СМИ о содержании контрольных измерительных материалов ЕГЭ, в 2014 году такой проблемы массово не было выявлено, и к результатам текущего года большая степень доверия.

Учитывая статистику, необходимо принять меры, позволяющие к ЕГЭ 2015 года:

1. Повысить квалификацию учителей информатики и ИКТ в 2014-2015 году на курсах повышения квалификации по вопросам подготовки к ЕГЭ и оценивания результатов выполнения заданий открытой формы.
2. Повысить качество подготовки учащихся по информатике и ИКТ в 2014-2015 году.
3. Создать банк методических рекомендаций по решению конкретных заданий контрольно-измерительных материалов ЕГЭ.
4. Создать банк типичных ошибок учащихся на ЕГЭ с примерами работ и рекомендаций по решению трудных заданий.
5. На заседаниях школьных методобъединений в начале учебного года следует актуализировать выше указанный материал и предоставить учителям возможность использования методических рекомендаций по подготовке к ЕГЭ по информатике и ИКТ в учебном процессе.
6. Регулярно осуществлять сетевые консультации учителей информатики и ИКТ по актуальным вопросам подготовки к ЕГЭ.

## **6. Результаты выполнения заданий по основным содержательным разделам учебного предмета «Информатика и ИКТ»**

Экзаменационная работа по информатике и ИКТ 2014 года 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.

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

Часть 1 (А) экзаменационной работы содержала 9 заданий базового уровня сложности и 4 задания повышенного уровня сложности.

Часть 2 (В) содержала 6 заданий базового уровня, 8 заданий повышенного уровня, 1 задание высокого уровня сложности.

Задания части 3 (С) относились к повышенному и высокому уровням (соответственно 1 и 3).

Задания базового уровня ориентированы на проверку знаний и умений инвариантной составляющей курса информатики, преподающегося в классах и учебных заведениях всех профилей. Правильное решение таких заданий позволяло получить только 37,5% первичных баллов (15 из 40). Правильный ответ экзаменуемого на половину заданий базового уровня позволяет получить минимально необходимый результат для участия в конкурсном отборе для поступления в вуз. В 2014 г. Рособранзором был установлен минимальный уровень в 8 первичных баллов.

Из заданий базового уровня 5 заданий посвящены математическим основам информатики (А1, А2, А3, А9, В4), 5 заданий – информационно-коммуникационным технологиям (А4, А6, А7, А8, В3) и 5 заданий посвящены основам теории алгоритмов и программирования (А5, В1, В2, В5, В6).

Элементы содержания заданий базового уровня по разделам предмета «Информатика и ИКТ»:

### **Раздел «Математические основы информатики»**

Тема «Измерение и кодирование информации, информационные процессы»:

- процесс передачи информации: источник и приемник информации, сигнал, кодирование и декодирование, искажение информации;
- дискретное (цифровое) представление информации, единицы измерения количества информации.

Тема «Системы счисления»:

- двоичное представление информации.

Тема «Моделирование»:

– описание (информационная модель) реального объекта и процесса в виде схемы, карты, таблицы, графика и/или формулы.

Тема «Основы логики»:

– высказывания, логические операции, кванторы, истинность высказывания.

Проверяемые умения или способы действий:

– оценивать числовые параметры информационных объектов и процессов;

– интерпретировать результаты, получаемые в ходе моделирования реальных процессов;

– строить модели объектов, систем и процессов в виде таблицы истинности для логического высказывания.

**Раздел «Информационно-коммуникационные технологии»**

Тема «Архитектура компьютера. Хранение и поиск информации в памяти ЭВМ»

– файловая система организации данных.

Тема «Технологии обработки числовой информации»

– построение электронных таблиц и обработка числовых данных с использованием формул;

– использование инструментов решения статистических и расчетно-графических задач.

Тема «Технологии хранения, поиска и сортировки информации в базах данных»

– системы управления базами данных, организация баз данных.

Тема «Технологии создания и обработки графической и мультимедийной информации»

– ввод и обработка звуковых объектов.

Проверяемые умения или способы действий:

– осуществлять поиск и отбор информации;

– проводить вычисления в электронных таблицах;

– представлять и анализировать табличную информацию в виде графиков и диаграмм;

– создавать и использовать структуры хранения данных;

– оценивать объем памяти, необходимый для хранения информации.

**Раздел «Основы теории алгоритмов и программирование»**

Тема «Алгоритмы и исполнители»

– формализация понятия алгоритма;

– построение алгоритмов в заданной системе команд исполнителя и практические вычисления;

– индуктивное определение объектов (рекурсия).

Тема «Программирование»

– типы данных в языке программирования;

– основные конструкции языка программирования.

Проверяемые умения или способы действий:

- строить информационные модели объектов, систем и процессов в виде алгоритмов;
- читать и отлаживать программы на языке программирования.

Задания повышенного уровня направлены на проверку освоения 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 математическим основам информатики (A10, A11, B7, B9, B10), 2 задания – информационно-коммуникационным технологиям (B11 и B12) и 6 заданий посвящены основам теории алгоритмов и программирования (A12, A13, B8, B13, B14, C1).

Элементы содержания заданий повышенного уровня сложности по разделам предмета «Информатика и ИКТ»:

### **Раздел «Математические основы информатики»**

Тема «Измерение и кодирование информации, информационные процессы»:

- дискретное (цифровое) представление информации, единицы измерения количества информации;
- скорость передачи информации.

Тема «Системы счисления»:

- позиционные системы счисления.

Тема «Моделирование»:

- описание (информационная модель) реального объекта и процесса в виде схемы, карты, таблицы, графика и/или формулы.

Тема «Основы логики»:

- высказывания, логические операции, кванторы, истинность высказывания.

Проверяемые умения или способы действий:

- вычислять логическое значение сложного высказывания по известным значениям элементарных высказываний;
- оценивать объем памяти, необходимый для хранения информации;
- оценивать скорость передачи и обработки информации;
- использовать готовые модели, оценивать их соответствие реальному объекту и целям моделирования;
- строить информационные модели объектов, систем и процессов в виде алгоритмов.

### **Раздел «Информационно-коммуникационные технологии»**

Тема «Архитектура компьютерных сетей и технологии поиска информации в сети»

- программная и аппаратная организация компьютеров и компьютерных систем;
- использование инструментов поисковых систем (формирование за-

просов).

Проверяемые умения или способы действий:

- работать с распространенными автоматизированными информационными системами, решать задачи, связанные с адресацией сетей;
- осуществлять поиск и отбор информации.

### **Раздел «Основы теории алгоритмов и программирование»**

#### Тема «Алгоритмы и исполнители»

- исполнение и анализ алгоритма;
- эквивалентные алгоритмические модели.

#### Тема «Программирование»

- основные конструкции языка программирования;
- работа с массивами (заполнение, считывание, поиск, сортировка и др.);
- программы, содержащие циклы и ветвление;
- программы с процедурами и функциями.

Проверяемые умения или способы действий:

- использовать готовые модели, оценивать их соответствие реальному объекту и целям моделирования;
- строить информационные модели объектов, систем и процессов в виде алгоритмов;
- интерпретировать результаты, получаемые в ходе моделирования реальных процессов;
- читать и отлаживать программы на языке программирования.

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 (В15) и три – основам теории алгоритмов и программирования (С2, С3, С4).

Элементы содержания заданий высокого уровня сложности по разделам предмета «Информатика и ИКТ»:

### **Раздел «Математические основы информатики»**

#### Тема «Основы логики»:

- высказывания, логические операции, кванторы, истинность высказывания.

Проверяемые умения или способы действий:

- вычислять логическое значение сложного высказывания по известным значениям элементарных высказываний.

### **Раздел «Основы теории алгоритмов и программирование»**

#### Тема «Алгоритмы и исполнители»

- разработка выигрышной стратегии, цепочки (конечные последовательности), деревья, списки, графы, матрицы (массивы), псевдослучайные последовательности.

Тема «Программирование»

- построение алгоритмов и практические вычисления;
- основные этапы разработки программ, разбиение задачи на подзадачи.

Проверяемые умения или способы действий:

- строить информационные модели объектов, систем и процессов в виде алгоритмов;
- создавать программы на языке программирования по их описанию.

В таблице 4 представлено распределение заданий по основным содержательным разделам учебного предмета «Информатика и ИКТ».

*Таблица 4*

**Распределение заданий по основным содержательным разделам учебного предмета «Информатика и ИКТ»**

Раздел	Тема	Число заданий / % от всего их колва (32)	Процент максимального первичного балла от максимального первичного балла за всю работу (40) , %	Номера заданий
1. Математические основы информатики	1.1. Измерение и кодирование информации, информационные процессы	4 / 12,50	10,00	A9, B4, A11, B10
	1.2. Системы счисления	2 / 6,25	5,00	A1 , B7
	1.3. Моделирование	2 / 6,25	5,00	A2, B9
	1.4. Основы логики	3 / 9,37	7,50	A3, A10, B15
2. Информационно-коммуникационные технологии	2.1. Архитектура компьютера. Хранение и поиск информации в памяти ЭВМ	1 / 3,13	2,50	A4
	2.2. Технологии обработки числовой информации	2 / 6,25	5,00	A7, B3
	2.3. Технологии хранения, поиска и сортировки информации в базах данных	1 / 3,13	2,50	A6
	2.4. Технологии создания и обработки графической и мультимедийной информации	1/ 3,13	2,50	A8
	2.5. Архитектура компьютерных сетей и технологии поиска информации в сети	2 / 6,25	5,00	B11, B12
	3. Основы теории алгоритмов и программирование	3.1. Алгоритмы и исполнители	6 / 18,74	20,00
	3.2. Программирование	8 / 25,00	35,00	B2, B5, A12, B8, B14, C1, C2, C4

Таким образом, содержание экзаменационной работы охватывает учебный материал всех разделов образовательной области «Информатика и ИКТ». Содержание и структура экзаменационной работы дают возможность достаточно полно проверить комплекс умений и способов действий по предмету.

Таблица 4 может служить основой для составления программы факультатива (элективного курса) в 10-11 классах по подготовке к ЕГЭ по информатике и ИКТ или программы подготовительных курсов в условиях довузовской подготовки.

В таблице 5 приведен процент выполнения заданий ЕГЭ.

*Таблица 5*

**Процент выполнения заданий ЕГЭ по информатике и ИКТ в 2014 г. (%)**

Кол-во баллов	X	0	1	2	3	4
<b>Задания</b>						
<b>Часть А</b>						
<b>A01</b>	0,19	14,64	85,16	–	–	–
<b>A02</b>	0,19	11,75	88,05	–	–	–
<b>A03</b>	0,19	18,30	81,50	–	–	–
<b>A04</b>	0	10,6	89,40	–	–	–
<b>A05</b>	0,19	34,49	65,32	–	–	–
<b>A06</b>	0	8,86	91,14	–	–	–
<b>A07</b>	0,39	30,25	69,36	–	–	–
<b>A08</b>	0	32,56	67,44	–	–	–
<b>A09</b>	0	43,35	56,65	–	–	–
<b>A10</b>	0,19	55,30	44,51	–	–	–
<b>A11</b>	0,77	51,64	47,59	–	–	–
<b>A12</b>	0,96	40,27	58,77	–	–	–
<b>A13</b>	0,96	46,82	52,22	–	–	–
<b>Часть В</b>						
<b>B01</b>	1,73	9,83	88,44	–	–	–
<b>B02</b>	0,58	14,07	85,36	–	–	–
<b>B03</b>	2,70	19,27	78,03	–	–	–
<b>B04</b>	6,17	51,45	42,39	–	–	–
<b>B05</b>	2,12	34,68	63,20	–	–	–
<b>B06</b>	6,94	80,54	12,52	–	–	–
<b>B07</b>	9,63	63,01	27,36	–	–	–
<b>B08</b>	13,68	47,21	39,11	–	–	–



Кол-во баллов Задания	X	0	1	2	3	4
<b>В09</b>	0,39	32,95	66,67	–	–	–
<b>В10</b>	6,94	40,85	52,22	–	–	–
<b>В11</b>	4,43	37,76	57,80	–	–	–
<b>В12</b>	3,85	55,68	40,46	–	–	–
<b>В13</b>	4,82	61,85	33,33	–	–	–
<b>В14</b>	23,12	50,87	26,01	–	–	–
<b>В15</b>	26,2	66,86	6,94	–	–	–
<b>Часть С</b>						
<b>С01</b>	42,00	9,63	7,32	16,76	24,28	–
<b>С02</b>	48,55	11,37	8,86	31,21	–	–
<b>С03</b>	30,64	4,82	13,49	12,52	38,54	–
<b>С04</b>	79,58	5,97	6,55	5,39	1,16	1,35

Как видно из таблицы, более 50% экзаменующихся правильно выполнили все задания части А, за исключением заданий А10 (справились 44,51%) и А11 (справились 47,59%). Следовательно, наибольшую трудность в этой части вызвали задания повышенного уровня сложности по темам «Измерение и кодирование информации, информационные процессы» и «Основы логики», в которых проверялись умения оценивать объем памяти, необходимый для хранения информации, а так же проводить анализ и вычислять логическое значение сложного высказывания по известным значениям элементарных высказываний.

Из 15 заданий второй части только с половиной справились 50% и более участников экзаменов, это задания В1-В3, В5 базового уровня сложности и В9-В11 повышенного уровня сложности. Хуже всего (менее 30%) выпускники школ справились с заданиями В6 – на выполнение рекурсивного алгоритма вычисления функции, В7 – на определение основания системы счисления, в которой записано число, где требуется показать умения составлять информационную модель в виде алгоритма, В14 – на выполнение программы с процедурами или функциями, В15 – на анализ системы логических уравнений. При этом в 2013 году перечисленные задания были выполнены более чем 50% выпускников ОУ, кроме задания В15. Задание В15 традиционно является одним из наиболее трудных на ЕГЭ по информатике и ИКТ.

Приступали к выполнению заданий 3 части (С) традиционно не выше 60% экзаменуемых, при этом на максимальный балл задания С1-С3 выполнили лишь 20-30%. К выполнению самого сложного задания С4 (на разработку алгоритма и программы) приступали только 20,42% экзаменуемых, а получили за него максимальный балл 1,35 процентов. Эта статистика практически в точности повторяет результаты ЕГЭ прошлого года.

## 6.1. Анализ выполнения части А

Анализ результатов выполнения части А показал, что в 2014 году выпускники общеобразовательных учреждений Алтайского края гораздо хуже справились с заданиями базового уровня сложности, чем в 2013 году.

В таблице 6 представлены результаты освоения разделов учебного предмета «Информатика и ИКТ» при выполнении заданий части А с подробной характеристикой заданий.

Таблица 6

### Результаты освоения разделов учебного предмета «Информатика и ИКТ» (Часть А)

Раздел	Тема	№ задания	Характеристика задания	% выполнивших задание, 2014	% выполнивших задание, 2013	Прирост
Математические основы информатики	Изменение и кодирование информации, информационные процессы	A9	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> процесс передачи информации: источник и приемник информации, сигнал, кодирование и декодирование, искажение информации</p> <p><i>Проверяемые умения:</i> интерпретировать результаты, получаемые в ходе моделирования реальных процессов</p> <p><i>Пример задания:</i> Для передачи по каналу связи сообщения, состоящего только из символов А, Б, В и Г, используется неравномерный (по длине) код: А – 0; Б – 10; В – 110. Каким кодовым словом нужно кодировать символ Г, чтобы длина его была минимальной, а код при этом допускал однозначное разбиение кодированного сообщения на символы? 1) 1    2) 1110    3) 111    4) 11</p>	56,65	86,80	-30,15
		A11	<p>Повышенный уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> дискретное (цифровое) представление информации, единицы измерения количества информации</p> <p><i>Проверяемые умения:</i> оценивать объем памяти, необходимый для хранения информации</p> <p><i>Пример задания:</i> Для регистрации на сайте некоторой страны пользователю требуется придумать пароль. Длина пароля – ровно 7 символов. В качестве символов используются десятичные цифры и 26 различных букв местного алфа-</p>	47,59	69,90	-22,31

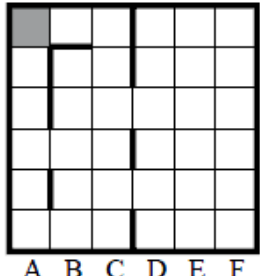
Раздел	Тема	№ задания	Характеристика задания	% выполненных задание, 2014	% выполненных задание, 2013	Прирост																																																	
			<p>вита, причём все буквы используются в двух начертаниях: как строчные, так и прописные (регистр буквы имеет значение!). Под хранение каждого такого пароля на компьютере отводится минимально возможное и одинаковое целое количество байтов, при этом используется посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством битов.</p> <p>Определите объём памяти, который занимает хранение 65 паролей.</p> <p>1) 390 байт      2) 455 байт 3) 520 байт      4) 325 байт</p>																																																				
	Системы счисления	A1	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> двоичное представление информации</p> <p><i>Проверяемые умения:</i> оценивать числовые параметры информационных объектов и процессов</p> <p><i>Пример задания:</i> Наибольшим десятичным числом, которое в двоичной системе счисления можно записать с помощью трёх цифр, является число</p> <p>1) 6      2) 7      3) 8      4) 12</p>	<b>85,16</b>	71,65	13,51																																																	
	Моделирование	A2	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> описание (информационная модель) реального объекта и процесса в виде схемы, карты, таблицы, графика и/или формулы</p> <p><i>Проверяемые умения:</i> интерпретировать результаты, получаемые в ходе моделирования реальных процессов</p> <p><i>Пример задания:</i> Между населёнными пунктами А, В, С, D, E, F построены дороги, протяжённость которых приведена в таблице. (Отсутствие числа в таблице означает, что прямой дороги между пунктами нет.)</p> <table border="1" data-bbox="579 1771 1064 1939"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <th>A</th> <td></td> <td>1</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <th>B</th> <td>1</td> <td></td> <td>10</td> <td>7</td> <td>10</td> <td></td> </tr> <tr> <th>C</th> <td></td> <td>10</td> <td></td> <td></td> <td>8</td> <td></td> </tr> <tr> <th>D</th> <td></td> <td>7</td> <td></td> <td></td> <td>2</td> <td></td> </tr> <tr> <th>E</th> <td></td> <td>10</td> <td>8</td> <td>2</td> <td></td> <td>5</td> </tr> <tr> <th>F</th> <td></td> <td></td> <td></td> <td></td> <td>5</td> <td></td> </tr> </tbody> </table> <p>Определите длину кратчайшего пути между пунктами А и F (при условии, что передвигаться можно только по построенным дорогам).</p>		A	B	C	D	E	F	A		1					B	1		10	7	10		C		10			8		D		7			2		E		10	8	2		5	F					5		<b>88,05</b>	81,94	6,11
	A	B	C	D	E	F																																																	
A		1																																																					
B	1		10	7	10																																																		
C		10			8																																																		
D		7			2																																																		
E		10	8	2		5																																																	
F					5																																																		

Раздел	Тема	№ задания	Характеристика задания	% выполненных задание, 2014	% выполненных задание, 2013	Прирост																																
			1) 10    2) 14    3) 15    4) 16																																			
	Основы логики	A3	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> высказывания, логические операции, кванторы, истинность высказывания</p> <p><i>Проверяемые умения:</i> строить модели объектов, систем и процессов в виде таблицы истинности для логического высказывания</p> <p><i>Пример задания:</i> Дан фрагмент таблицы истинности выражения F.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>x1</th> <th>x2</th> <th>x3</th> <th>x4</th> <th>x5</th> <th>x6</th> <th>x7</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td>0</td> <td>1</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> </tr> </tbody> </table> <p>Каким выражением может быть F?</p> <p>1) <math>\neg x1 \vee \neg x2 \vee \neg x3 \vee x4 \vee \neg x5 \vee x6 \vee x7</math>  2) <math>\neg x1 \wedge x2 \wedge \neg x3 \wedge x4 \wedge x5 \wedge \neg x6 \wedge \neg x7</math>  3) <math>x1 \wedge \neg x2 \wedge \neg x3 \wedge x4 \wedge \neg x5 \wedge x6 \wedge x7</math>  4) <math>x1 \vee \neg x2 \vee \neg x3 \vee \neg x4 \vee \neg x5 \vee x6 \vee x7</math></p>	x1	x2	x3	x4	x5	x6	x7	F	1	0	0	1	0	1	1	1	1	1	1	0	1	0	0	0	0	1	1	1	1	0	0	0	<b>81,50</b>	85,24	-3,74
x1		x2	x3	x4	x5	x6	x7	F																														
1	0	0	1	0	1	1	1																															
1	1	1	0	1	0	0	0																															
0	1	1	1	1	0	0	0																															
		A10	<p>Повышенный уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> высказывания, логические операции, кванторы, истинность высказывания</p> <p><i>Проверяемые умения:</i> вычислять логическое значение сложного высказывания по известным значениям элементарных высказываний, анализировать множества элементов, удовлетворяющих высказыванию</p> <p><i>Пример задания:</i> Выберите такой отрезок A, что формула  <math>((x \in A) \wedge (x \in P)) \rightarrow (x \in Q)</math>  тождественно истинна, т.е. принимает значение 1 при любом значении переменной x.</p> <p>1) [24, 49]                      2) [29, 90]  3) [45, 120]                      4) [91, 140]</p>	<b>44,51</b>	57,86	-13,35																																
<b>Информационно-коммуникационные технологии</b>	Архитектура компьютера. Хранение и поиск информации в памяти ЭВМ	A4	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> файловая система организации данных</p> <p><i>Проверяемые умения:</i> осуществлять поиск и отбор информации</p> <p><i>Пример задания:</i> Для групповых операций с файлами используются <b>маски имён файлов</b>. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которой также могут встречать-</p>	<b>89,40</b>	89,71	-0,31																																

Раздел	Тема	№ задания	Характеристика задания	% выпол- нивших задание, 2014	% выпол- нивших задание, 2013	При- рост																																				
			<p>ся следующие символы. Символ «?» (вопросительный знак) означает ровно один произвольный символ. Символ «*» (звездочка) означает любую последовательность символов произвольной длины, в том числе «*» может задавать и пустую последовательность. Определите, какое из указанных имён файлов удовлетворяет маске: <b>b?ar*.?xt</b> 1) baara.txt                    2) blar.txt 3) bar.ext                        4) blar.txt</p>																																							
	Технологии обработки числовой информации	A7	<p>Базовый уровень сложности <i>Проверяемые элементы содержания:</i> построение электронных таблиц и обработка числовых данных с использованием формул <i>Проверяемые умения:</i> проводить вычисления в электронных таблицах <i>Пример задания:</i> Коле нужно с помощью электронных таблиц построить таблицу двузначных чисел от 10 до 49. Для этого сначала в диапазоне В1:К1 он записал числа от 0 до 9, и в диапазоне А2:А5 он записал числа от 1 до 4. Затем в ячейку В2 записал формулу двузначного числа (А2 – число десятков; В1 – число единиц), после чего скопировал её во все ячейки диапазона В2:К5. В итоге получил таблицу двузначных чисел. На рисунке ниже представлен фрагмент этой таблицы.</p> <table border="1" data-bbox="582 1467 1013 1680"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> </tr> </thead> <tbody> <tr> <th>1</th> <td></td> <td>0</td> <td>1</td> <td>2</td> <td>3</td> </tr> <tr> <th>2</th> <td>1</td> <td>10</td> <td>11</td> <td>12</td> <td>13</td> </tr> <tr> <th>3</th> <td>2</td> <td>20</td> <td>21</td> <td>22</td> <td>23</td> </tr> <tr> <th>4</th> <td>3</td> <td>30</td> <td>31</td> <td>32</td> <td>33</td> </tr> <tr> <th>5</th> <td>4</td> <td>40</td> <td>41</td> <td>42</td> <td>43</td> </tr> </tbody> </table> <p>Какая формула была записана в ячейке В2? 1) =\$A2*10+\$B1 2) =A\$2*10+\$B1 3) =\$A2*10+B\$1 4) =A2*10+B1</p>		A	B	C	D	E	1		0	1	2	3	2	1	10	11	12	13	3	2	20	21	22	23	4	3	30	31	32	33	5	4	40	41	42	43	<b>69,36</b>	63,69	5,67
	A	B	C	D	E																																					
1		0	1	2	3																																					
2	1	10	11	12	13																																					
3	2	20	21	22	23																																					
4	3	30	31	32	33																																					
5	4	40	41	42	43																																					
	Технологии хранения, поиска	A6	<p>Базовый уровень сложности <i>Проверяемые элементы содержания:</i> системы управления базами данных, организация баз данных <i>Проверяемые умения:</i></p>	<b>91,14</b>	86,80	4,34																																				

Раздел	Тема	№ зада-ния	Характеристика задания	% выпол-нивших задание, 2014	% выпол-нивших задание, 2013	При-рост																																																																											
	и сор-тировки инфор-мации в базах данных		<p>создавать и использовать структуры хранения данных</p> <p><i>Пример задания:</i></p> <p>Ниже представлены две таблицы из базы данных. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информа-ция представлена значением поля ID в соответствующей строке таблицы 1. Определите на основании приведён-ных данных фамилию и инициалы дяди Гресс О.С.</p> <p><i>Пояснение: дядей считается родной брат отца или матери.</i></p> <table border="1" style="display: inline-table; margin-right: 20px;"> <caption>Таблица 1</caption> <thead> <tr> <th>ID</th> <th>Фамилия И.О.</th> <th>Пол</th> </tr> </thead> <tbody> <tr><td>14</td><td>Грач Н.А.</td><td>Ж</td></tr> <tr><td>24</td><td>Петренко И.П.</td><td>М</td></tr> <tr><td>25</td><td>Петренко П.И.</td><td>М</td></tr> <tr><td>26</td><td>Петренко П.П.</td><td>М</td></tr> <tr><td>34</td><td>Ерёма А.И.</td><td>Ж</td></tr> <tr><td>35</td><td>Ерёма В.С.</td><td>Ж</td></tr> <tr><td>36</td><td>Ерёма С.С.</td><td>М</td></tr> <tr><td>44</td><td>Лебедь А.С.</td><td>Ж</td></tr> <tr><td>45</td><td>Лебедь В.А.</td><td>М</td></tr> <tr><td>46</td><td>Гресс О.С.</td><td>Ж</td></tr> <tr><td>47</td><td>Гресс П.О.</td><td>М</td></tr> <tr><td>54</td><td>Клычко А.П.</td><td>Ж</td></tr> <tr><td>64</td><td>Крот П.А.</td><td>Ж</td></tr> <tr><td>...</td><td>...</td><td>...</td></tr> </tbody> </table> <table border="1" style="display: inline-table;"> <caption>Таблица 2</caption> <thead> <tr> <th>ID Родителя</th> <th>ID Ребёнка</th> </tr> </thead> <tbody> <tr><td>24</td><td>25</td></tr> <tr><td>44</td><td>25</td></tr> <tr><td>25</td><td>26</td></tr> <tr><td>64</td><td>26</td></tr> <tr><td>24</td><td>34</td></tr> <tr><td>44</td><td>34</td></tr> <tr><td>34</td><td>35</td></tr> <tr><td>36</td><td>35</td></tr> <tr><td>14</td><td>36</td></tr> <tr><td>34</td><td>46</td></tr> <tr><td>36</td><td>46</td></tr> <tr><td>25</td><td>54</td></tr> <tr><td>64</td><td>54</td></tr> <tr><td>...</td><td>...</td></tr> </tbody> </table> <p>1) Петренко И.П. 2) Петренко П.И. 3) Лебедь В.А. 4) Гресс П.О.</p>	ID	Фамилия И.О.	Пол	14	Грач Н.А.	Ж	24	Петренко И.П.	М	25	Петренко П.И.	М	26	Петренко П.П.	М	34	Ерёма А.И.	Ж	35	Ерёма В.С.	Ж	36	Ерёма С.С.	М	44	Лебедь А.С.	Ж	45	Лебедь В.А.	М	46	Гресс О.С.	Ж	47	Гресс П.О.	М	54	Клычко А.П.	Ж	64	Крот П.А.	Ж	...	...	...	ID Родителя	ID Ребёнка	24	25	44	25	25	26	64	26	24	34	44	34	34	35	36	35	14	36	34	46	36	46	25	54	64	54	...	...			
ID	Фамилия И.О.	Пол																																																																															
14	Грач Н.А.	Ж																																																																															
24	Петренко И.П.	М																																																																															
25	Петренко П.И.	М																																																																															
26	Петренко П.П.	М																																																																															
34	Ерёма А.И.	Ж																																																																															
35	Ерёма В.С.	Ж																																																																															
36	Ерёма С.С.	М																																																																															
44	Лебедь А.С.	Ж																																																																															
45	Лебедь В.А.	М																																																																															
46	Гресс О.С.	Ж																																																																															
47	Гресс П.О.	М																																																																															
54	Клычко А.П.	Ж																																																																															
64	Крот П.А.	Ж																																																																															
...	...	...																																																																															
ID Родителя	ID Ребёнка																																																																																
24	25																																																																																
44	25																																																																																
25	26																																																																																
64	26																																																																																
24	34																																																																																
44	34																																																																																
34	35																																																																																
36	35																																																																																
14	36																																																																																
34	46																																																																																
36	46																																																																																
25	54																																																																																
64	54																																																																																
...	...																																																																																
	Техно-логии созда-ния и обра-ботки графич-еской и мультиме-дийной инфор-мации	A8	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> ввод и обработка звуковых объектов</p> <p><i>Проверяемые умения:</i> оценивать объем памяти, необходи-мый для хранения информации</p> <p><i>Пример задания:</i></p> <p>Производится четырёхканальная (квадро) звукозапись с частотой дискретизации 48 кГц и 32-битным разрешением. Запись длится 2 мину-ты, её результаты записываются в файл, сжатие данных не производит-ся. Какая из приведённых ниже вели-чин наиболее близка к размеру полу-ченного файла?</p> <p>1) 15 Мбайт                      2) 27 Мбайт 3) 42 Мбайт                      4) 88 Мбайт</p>	<b>67,44</b>	74,76	-7,32																																																																											
<b>Осно-вы теории алго-ритмов и про-</b>	Алго-ритмы и ис-полни-тели	A5	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> алгоритм, свойства алгоритма</p> <p><i>Проверяемые умения:</i> выполнять алгоритм</p> <p><i>Пример задания:</i></p>	<b>65,32</b>	78,83	-13,51																																																																											

Раздел	Тема	№ задания	Характеристика задания	% выпол- нивших задание, 2014	% выпол- нивших задание, 2013	При- рост
грам- миро- вание			<p>В некоторой информационной системе информация кодируется двоичными шестиразрядными словами. При передаче данных возможны их искажения, поэтому в конец каждого слова добавляется седьмой (контрольный) разряд таким образом, чтобы сумма разрядов нового слова, считая контрольный, была чётной. Например, к слову 110011 справа будет добавлен 0, а к слову 101100 – 1.</p> <p>После приёма слова производится его обработка. При этом проверяется сумма его разрядов, включая контрольный. Если она нечётна, это означает, что при передаче этого слова произошёл сбой, и оно автоматически заменяется на зарезервированное слово 0000000. Если она чётна, это означает, что сбоя не было или сбоев было больше одного. В этом случае принятое слово не изменяется.</p> <p>Исходное сообщение 1100101 1001011 0011000 было принято в виде 1100111 1001110 0011000.</p> <p>Как будет выглядеть принятое сообщение после обработки?</p> <p>1) 1100111 1001011 0011000 2) 1100111 1001110 0000000 3) 0000000 0000000 0011000 4) 0000000 1001110 0011000</p>			
		A13	<p>Повышенный уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> исполнение и анализ алгоритма</p> <p><i>Проверяемые умения:</i> интерпретировать результаты, получаемые в ходе моделирования реальных процессов</p> <p><i>Пример задания:</i> Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости, включает в себя 4 команды-приказа и 4 команды – проверки условия. Команды-приказы: вверх, вниз, влево, вправо. При выполнении любой из этих команд РОБОТ перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →. Если РОБОТ начнёт движение в сторону находящейся рядом с ним стены,</p>	52,22	74,37	-22,15

Раздел	Тема	№ задания	Характеристика задания	% выпол- нивших задание, 2014	% выпол- нивших задание, 2013	При- рост
			<p>то он разрушится, и программа прервётся.</p> <p>Другие 4 команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ: <b>сверху свободно, снизу свободно, слева свободно, справа свободно.</b></p> <p>Цикл          ПОКА <i>условие</i>              <i>последовательность команд</i>          КОНЕЦ ПОКА</p> <p>выполняется, пока условие истинно.</p> <p>В конструкции          ЕСЛИ <i>условие</i>              ТО <i>команда1</i>              ИНАЧЕ <i>команда2</i>          КОНЕЦ ЕСЛИ</p> <p>выполняется <i>команда1</i> (если условие истинно) или <i>команда2</i> (если условие ложно).</p> <p>Сколько клеток лабиринта соответствуют требованию, что, начав движение в ней и выполнив предложенную программу, РОБОТ уцелеет и остановится в закрашенной клетке (клетка A1)?</p>  <p>1) 14    2) 18    3) 20    4) 22</p>			
Программирование		A12	<p>Повышенный уровень сложности</p> <p><i>Проверяемые элементы содержания:</i>          работа с массивами (заполнение, считывание, поиск, сортировка, массовые операции и др.)</p> <p><i>Проверяемые умения:</i>          читать и отлаживать программы на языке программирования</p> <p><i>Пример задания:</i>          Ниже приведён фрагмент программы, записанный на четырёх языках программирования.</p> <p>Массив A одномерный; в программе рассматривается его фрагмент, соответствующий значениям индекса от 1 до n.</p>	<b>58,77</b>	76,12	-17,35



Раздел	Тема	№ задания	Характеристика задания	% выполнивших задание, 2014	% выполнивших задание, 2013	Прирост
			<p><b>Паскаль</b></p> <pre> j := 1; for i := 1 to n do begin   if A[i] &lt; A[j] then     j := i end; s := j; </pre> <p>Чему будет равно значение переменной s после выполнения данного фрагмента программы?</p> <ol style="list-style-type: none"> <li>1) минимальному элементу в массиве A</li> <li>2) индексу минимального элемента в массиве A (наименьшему из таких индексов, если минимальных элементов несколько)</li> <li>3) индексу минимального элемента в массиве A (наибольшему из таких индексов, если минимальных элементов несколько)</li> <li>4) количеству элементов, равных минимальному в массиве A</li> </ol>			

Из приведенной таблицы видно, что в разделе «Математические основы информатики» на высоком уровне выпускники выполняют задания на представление чисел в различных системах счисления (A1 – 85,16%), на работу с информационной моделью в виде таблицы (A2 – 88,05%), на построение таблицы истинности для логического выражения (A3 – 81,5%). В разделе «Информационно-коммуникационные технологии» высокий процент выполнимости у заданий на поиск и отбор информации в файловой системе (A4 – 89,4%), на поиск информации в базе данных (A6 – 91,14%). Хотя, по сравнению с прошлым годом наблюдается понижение процента справившихся с этими заданиями выпускников.

Однако в каждой части (А, В и С) Единого государственного экзамена по русскому языку можно говорить о заданиях, которые вызывают у учащихся определённые трудности при их выполнении. Анализ таких заданий позволяет выявить типичные ошибки, причины затруднений и спланировать виды деятельности для их устранения.

В таблице 7 представлены «проблемные темы» освоения разделов учебного предмета «Информатика и ИКТ» при выполнении заданий части А, расположенные в порядке убывания их сложности для учащихся. В таблице выделены те задания, процент выполнения которых составил менее 60, что говорит о низкой сформированности как теоретических знаний по проверяемому разделу дисциплины, так и практических умений и навыков.

**«Проблемные темы» освоения разделов учебного предмета  
«Информатика и ИКТ» (Часть А)**

№ задания	Спецификация задания	% выполнивших задание
А10	Основы логики <i>Проверяемые элементы содержания:</i> высказывания, логические операции, кванторы, истинность высказывания <i>Проверяемые умения:</i> вычислять логическое значение сложного высказывания по известным значениям элементарных высказываний, анализировать множества элементов, удовлетворяющих высказыванию	44,57
А11	Измерение и кодирование информации, информационные процессы <i>Проверяемые элементы содержания:</i> дискретное (цифровое) представление информации, единицы измерения количества информации <i>Проверяемые умения:</i> оценивать объем памяти, необходимый для хранения информации	47,59
А13	Алгоритмы и исполнители <i>Проверяемые элементы содержания:</i> исполнение и анализ алгоритма <i>Проверяемые умения:</i> интерпретировать результаты, получаемые в ходе моделирования реальных процессов	52,22
А9	Измерение и кодирование информации, информационные процессы <i>Проверяемые элементы содержания:</i> процесс передачи информации: источник и приемник информации, сигнал, кодирование и декодирование, искажение информации <i>Проверяемые умения:</i> интерпретировать результаты, получаемые в ходе моделирования реальных процессов	56,65
А12	Программирование <i>Проверяемые элементы содержания:</i> работа с массивами (заполнение, считывание, поиск, сортировка, массовые операции и др.) <i>Проверяемые умения:</i> читать и отлаживать программы на языке программирования	58,77

Ниже приведен анализ «проблемных» тем.

**А09.** Измерение и кодирование информации, информационные процессы: процесс передачи информации – источник и приемник информации, сигнал, кодирование и декодирование, искажение информации. Процент выполнения – **56,65%** (2013 г. процент выполнения – 86,8%). Наблюдаем понижение процента выполнимости по сравнению с 2013 годом более чем на 30.

Для выполнения задания учащиеся должны владеть следующими понятиями:

- кодирование – это перевод информации с одного языка на другой (запись в другой системе символов, в другом алфавите);
- кодированием называют перевод информации с естественного языка на формальный, а декодированием – обратный переход;

- кодирование может быть равномерное и неравномерное: при равномерном кодировании все символы кодируются кодами равной длины; при неравномерном кодировании разные символы могут кодироваться кодами разной длины, это затрудняет декодирование, но позволяет уменьшить объем информации в сообщении за счет использования более коротких кодов для символов, чаще встречающихся в тексте;

- закодированное сообщение можно однозначно декодировать с начала, если выполняется условие Фано: никакое кодовое слово не является началом другого кодового слова;

- закодированное сообщение можно однозначно декодировать с конца, если выполняется обратное условие Фано: никакое кодовое слово не является окончанием другого кодового слова;

- условие Фано – это достаточное, но не необходимое условие однозначного декодирования.

Наиболее типичной ошибкой при выполнении заданий данного вида является допущение неоднозначного кодирования. Основной причиной является незнание теорем (условий) Фано. Для правильного выполнения задания желательно, чтобы учащиеся владели приемом построения графа в виде дерева, для выявления занятых вершин, стоящих на пути другого кода, рисунок 6 – Г и Д.

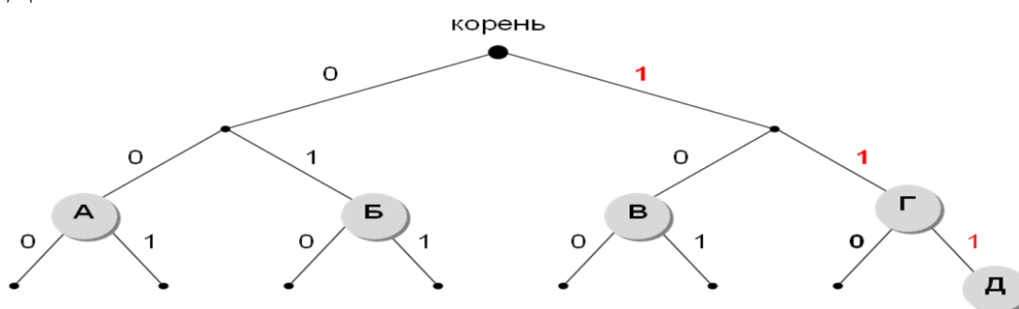


Рисунок 6. Дерево неравномерных кодов

Часто при выборе ответа экзаменующиеся забывают об условии минимальности длины кода.

Кроме того, многие учащие просто не имеют понятия о неравномерных кодах и правилах их построения, встречаясь в школьном курсе информатики только с равномерным кодированием.

Учащиеся, не владеющие понятием «декодирование», выполняя задание на декодирование сообщения, применяют неверный алгоритм действий.

При подготовке к выполнению таких заданий, необходимо сформировать понятия «равномерное кодирование», «неравномерное кодирование», «декодирование». Затем, необходимо познакомить учащихся с теоремами Фано и отработать решение задач на проверку однозначности кодов, задач на минимизацию длины кода. При решении задач на получение неравномерных кодов рекомендуется применять дерево и выполнять следующий план действий: 1. Получить равномерные коды; 2. Если на предыдущем уровне дерева есть свободные позиции, то переносим на него возможное количество символов.

лов, так, чтобы выполнялось условие Фано; 3. Выписываем полученные коды.

**A10. Основы логики: высказывания, логические операции, кванторы, истинность высказывания.** Процент выполнения – **44,51%** (2013 г. процент выполнения – 57,86%). Наблюдаем понижение процента выполняемости по сравнению с 2013 годом более чем на 10.

Это задание традиционно сложное для выполнения на ЕГЭ.

Для выполнения задания учащиеся должны знать:

- что является высказыванием;
- как обозначаются высказывания;
- что такое логическое выражение;
- условные обозначения логических операций;
- таблицы истинности логических операций «И», «ИЛИ», «НЕ», «импликация»;
- приоритеты логических операций;
- законы логики, особенно полезен закон устранения импликации и законы де Моргана.

Кроме этого, для выполнения данного задания необходимо владеть понятием «множества» и операциями над множествами, хорошо знать числовые множества. Полезно уметь представлять множества графически на числовой оси и с помощью диаграмм Эйлера.

Типичные ошибки экзаменуемых при выполнении данного задания – это ошибки в законах логики в процессе упрощения исходного логического выражения, а так же ошибки определения принадлежности элементов пересечению и объединению числовых множеств.

Частой ошибкой является «забывание» про отрицание выражения. При этом отрицанием выражения типа  $X > 5$  экзаменуемые ошибочно считают  $X < 5$ , а не  $X \leq 5$  как требуется.

Следовательно, при подготовке учащихся необходимо обратить на данные замечания внимание.

**A11. Измерение и кодирование информации, информационные процессы: дискретное (цифровое) представление информации, единицы измерения количества информации.** Процент выполнения – **47,59%** (2013 г. процент выполнения – 69,9%). Наблюдаем понижение процента выполняемости по сравнению с 2013 годом более чем на 20.

Для выполнения задания у учащихся должна быть сформирована система понятий и умений:

- «Алфавит» – совокупность всех различных символов, которая используется для записи сообщения;
- «Мощность алфавита» ( $M$ ) – это количество символов в алфавите;
- Количество всех возможных «слов» (символьных цепочек) длиной  $N$  (без учета смысла) в алфавите имеет мощностью  $M$  равно  $Q = M^N$ ; для двоичного кодирования (мощность алфавита  $M = 2$  символа) получаем из-

вестную формулу:  $Q = 2^N$ ;

- С помощью  $K$  бит можно закодировать  $Q = 2^K$  различных вариантов (чисел);

- Информационный объем сообщения ( $I$ ) равен произведению количества символов в нем ( $K$ ) на число бит, необходимых для кодирования одного символа ( $N$ ):  $I = N \cdot K$ .

Кроме этого учащимся важно знать единицы измерения информации и уметь выполнять перевод из одних единиц измерения в другие. Особенно важно при работе с «большими» числами уметь пользоваться степенями двойки и хорошо владеть свойствами степеней и дробей, таблица 8.

Таблица 8

### Единицы измерения информации

Единица	Обозначение	Значение в «соседних» единицах	Значение в битах (по степеням 2)
1 байт	Б (B)	8 бит	$2^3$
1 Килобайт	Кб (Kb)	1024 байт	$2^{13}$
1 Мегабайт	Мб (Mb)	1024 килобайт	$2^{23}$
1 Гигабайт	Гб (Gb)	1024 мегабайт	$2^{33}$
1 Терабайт	Тб (Tb)	1024 гигабайт	$2^{43}$
1 Петабайт	Пб (Pb)	1024 терабайт	$2^{53}$
1 Эксабайт	Эб (Eb)	1024 петабайт	$2^{63}$
1 Зеттабайт	Зб (Zb)	1024 эксабайт	$2^{73}$
1 Йоттабайт	Йб (Yb)	1024 зеттабайт	$2^{83}$

Типичными ошибками экзаменуемых при выполнении данного задания являются:

- 1) не выполнение условия использования минимального ЦЕЛОГО количества бит (байт) для кодирования значений;
- 2) при выборе ответов «путают» единицы измерения;
- 3) «забывают» различать регистры используемых символов;
- 4) информационный вес символа (глубину кодирования) «путают» с длиной кода.

При подготовке учащихся необходимо сформировать умения решать задачи на определение возможного количества кодов при известной мощности алфавита и длине кода. Затем важно сформировать понятие обратной задачи – на определение глубины кодирования и научить её решать. После этого необходимо рассматривать задачи на определение объема сообщения, содержащего известное количество кодовых слов. Полезно рассмотреть и обратную задачу – на определение количества закодированных элементов в сообщении известного объема.

**A12. Программирование: работа с массивами (заполнение, считывание, поиск, сортировка, массовые операции и др.).** Процент выполнения – **58,77%** (2013 г. процент выполнения – 76,12%). Наблюдаем понижение процента выполняемости задания по сравнению с 2013 годом более чем на 17.

Для выполнения задания учащимся необходимо знать:

- формат записи и работу цикла с параметром;
- что такое массив – это набор однотипных элементов, имеющих общее имя и расположенных в памяти рядом;
- для обращения к элементу массива используют квадратные скобки, запись  $A[i]$  обозначает элемент массива  $A$  с номером (индексом)  $i$ ;
- матрица (двухмерный массив) – это прямоугольная таблица однотипных элементов;
- если матрица имеет имя  $A$ , то обращение  $A[i,k]$  обозначает элемент, расположенный на пересечении строки  $i$  и столбца  $k$ ;
- элементы, у которых номера строки и столбца совпадают, расположены на главной диагонали;
- выше главной диагонали расположены элементы, у которых номер строки меньше номера столбца;
- ниже главной диагонали расположены элементы, у которых номер строки больше номера столбца;
- при использовании вложенных циклов сначала выполняется полностью тело внутреннего цикла, при этом параметр «перебирает» все значения от начального заданного до конечного, и только после этого управление передается внешнему циклу и меняется значение его параметра.

Кроме перечисленных знаний учащиеся должны владеть трассировкой (пошаговой отладкой) программ.

Типичной и одной из основных ошибок экзаменуемых при выполнении данного задания является неумение работать с вложенными циклами и неверное присваивание значений параметрам циклов при трассировке.

При работе с массивами данных экзаменуемые часто «путают» элемент массива с его индексом.

При подготовке учащихся рекомендуется составлять трассировочные таблицы, содержащие все переменные, при этом важно обратить внимание на изменение значений  $i$  и  $A[i]$ .

**A13. Алгоритмы и исполнители: исполнение и анализ алгоритма.** Процент выполнения – **52,22%** (2013 г. процент выполнения – 74,37%). Наблюдаем понижение процента выполняемости задания по сравнению с 2013 годом более чем на 20.

Для выполнения задания у учащихся должны быть сформированы понятия:

- «алгоритм», «исполнитель», «система команд исполнителя»,

«среда исполнителя», «отказы».

Учащиеся должны знать:

- основные алгоритмические конструкции – следование, ветвление, цикл.

Учащиеся должны уметь:

- выполнять пошагово представленный алгоритм, с учетом системы команд исполнителя и отказов его работы;
- анализировать действия исполнителя в указанных условиях.

Типичные ошибки экзаменуемых при выполнении данного задания:

1) чаще всего выполняют не общий анализ алгоритма с рассмотрением возможных вариантов начальной позиции, а перебирают варианты исхода при всех возможных начальных позициях исполнителя, при этом теряют по невнимательности некоторые решения;

2) невнимательно относятся к условиям продолжения (выхода) из цикла;

3) не учитывают порядок выполнения операций внутри цикла при анализе возможных вариантов прохода по лабиринту;

4) не умеют работать с вложенными циклами;

5) не учитывают начальной позиции при которой исполнитель не выполнив ни одного действия останется на месте, она тоже может включаться в решение.

При подготовке учащихся важно отработать различные варианты заданий на выполнение и невыполнение условий ветвления и продолжения (выхода) цикла. Нужно обратить внимание на то, что внешний цикл может выполняться более одного раза, и, что цикл может не выполниться ни разу; неучет этого обстоятельства приводит к потере решений.

При выполнении заданий типа А13 важно не перебирать все возможные варианты, их может оказаться довольно много, а выполнить анализ: правильно определить свойства, по которым клетку можно считать «кандидатом», затем посчитать количество таких клеток.

При выборе «кандидата» важно обращать внимание на порядок выполнения внутренних циклов (например, сначала Робот идет вправо до препятствия, и только затем – вниз). При изменении этого порядка изменится и результат, в частности, изменятся условия, определяющие особую клетку.

### **Выводы по выполнению заданий части А**

1. В результате анализа выполнения заданий первой части ЕГЭ по информатике и ИКТ в 2014 году можно констатировать слабое владение даже на базовом уровне абитуриентами темой «Измерение и кодирование информации, информационные процессы», а так же низкий уровень знаний учащихся с базовой подготовкой зафиксирован по программированию. На изучение теоретических основ информатики и тему «Кодирование информации»

следует особо обратить внимание, важно сформировать понимание приемов равномерного и неравномерного кодирования и применение этих знаний при решении задач прикладного значения. Владение основами программирования необходимо абитуриентам, выбирающим информатику как профильную дисциплину для изучения, а следовательно нельзя недооценивать его роль в содержании предмета «Информатика и ИКТ». Следует отработать основные приемы работы с хранимыми данными простых типов в памяти ЭВМ (присваивание, переприсваивание, ввод, вывод). Обратить внимание необходимо на отработку приемов оперирования массивами однотипных данных, а эти операции невозможны без владения понятием «цикл».

2. Твердые знания на базовом уровне показали экзаменуемые по темам «системы счисления» и «моделирование». Ровные и уверенные знания на базовом уровне сложности продемонстрированы по разделу «информационно-коммуникационные технологии». То есть, отмечавшиеся ранее пробелы в изучении отдельных тем этого раздела закрыты, уровень знаний участников ЕГЭ по информатике соответствует требованиям государственного стандарта образования.

3. По теме «основы логики» можно констатировать уверенное владение на базовом уровне и низкое – на повышенном уровне. Особое внимание следует обратить на овладение законами логики и приемами упрощения логического выражения.

4. По теме «алгоритмы и исполнители» так же можно констатировать уверенное владение на базовом уровне и низкое – на повышенном, тогда как, тема занимает центральное положение в школьном курсе информатики и должна быть хорошо отработана за долгие годы развития предмета содержанием обучения.

## 6.2. Анализ выполнения части В

Результаты выполнения части В демонстрируют аналогичную динамику изменений по сравнению с прошлым годом, что и результаты выполнения части А. Хотя здесь темп убыли процента выполнимости более сглажен.

В таблице 9 представлены результаты освоения разделов учебного предмета «Информатика и ИКТ» при выполнении заданий части В с подробной характеристикой заданий.

Таблица 9


### Результаты освоения разделов учебного предмета «Информатика и ИКТ» (Часть В)

Раздел	Тема	№ задания	Характеристика задания	% выполнивших задание, 2014	% выполнивших задание, 2013	Прирост
Математические основы	Изменение и кодирование	В4	Базовый уровень сложности <i>Проверяемые элементы содержания:</i> дискретное (цифровое) представление информации, единицы измерения ко-	42,39	50,87	-8,48



Раздел	Тема	№ задания	Характеристика задания	% выпол- нивших задание, 2014	% выпол- нивших задание, 2013	При- рост
инфор- фор- мати- ки	инфор- мации, инфор- маци- онные процес- сы		личества информации <i>Проверяемые умения:</i> оценивать числовые параметры ин- формационных объектов и процессов <i>Пример задания:</i> Для передачи аварийных сигналов договорились использовать специаль- ные цветные сигнальные ракеты, за- пускаемые последовательно. Одна последовательность ракет – один сиг- нал; в каком порядке идут цвета – существенно. Какое количество раз- личных сигналов можно передать при помощи запуска ровно пяти таких сигнальных ракет, если в запасе име- ются ракеты трёх различных цветов (ракет каждого вида неограниченное количество, цвет ракет в последова- тельности может повторяться)?			
		V10	Повышенный уровень сложности <i>Проверяемые элементы содержания:</i> скорость передачи информации <i>Проверяемые умения:</i> оценивать скорость передачи и обра- ботки информации <i>Пример задания:</i> Документ объёмом 16 Мбайт можно передать с одного компьютера на дру- гой двумя способами. А. Сжать архиватором, передать архив по каналу связи, распаковать. Б. Передать по каналу связи без ис- пользования архиватора. Какой способ быстрее и насколько, если: – средняя скорость передачи данных по каналу связи составляет $2^{21}$ бит в секунду; – объём сжатого архиватором доку- мента равен 25% исходного; – время, требуемое на сжатие доку- мента, – 12 секунд, на распаковку – 3 секунды? В ответе напишите букву А, если быстрее способ А, или Б, если быст- рее способ Б. Сразу после буквы напиши- те число, обозначающее, на сколько секунд один способ быстрее другого. Так, например, если способ Б быстрее способа А на 23 секунды, в ответе нужно написать В23.	<b>52,22</b>	60,19	-7,97
		Систе-	V7	Повышенный уровень сложности	<b>27,36</b>	75,15

Раздел	Тема	№ задания	Характеристика задания	% выполнивших задание, 2014	% выполнивших задание, 2013	Прирост
	мы счисления		<p><i>Проверяемые элементы содержания:</i> позиционные системы счисления</p> <p><i>Проверяемые умения:</i> строить информационные модели объектов, систем и процессов в виде алгоритмов</p> <p><i>Пример задания:</i> Укажите наименьшее основание системы счисления, в которой запись десятичного числа 30 имеет ровно три значащих разряда.</p>			
	Моделирование	В9	<p>Повышенный уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> описание (информационная модель) реального объекта и процесса в виде схемы, карты, таблицы, графика и/или формулы</p> <p><i>Проверяемые умения:</i> использовать готовые модели, оценивать их соответствие реальному объекту и целям моделирования</p> <p><i>Пример задания:</i> На рисунке – схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Л?</p>	<b>66,67</b>	62,52	4,15
	Основы логики	В15	<p>Высокий уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> высказывания, логические операции, кванторы, истинность высказывания</p> <p><i>Проверяемые умения:</i> вычислять логическое значение сложного высказывания по известным значениям элементарных высказываний</p> <p><i>Пример задания:</i> Сколько существует различных наборов значений логических переменных <math>x_1, x_2, \dots, x_{10}</math>, которые удовлетворяют всем перечисленным ниже условиям?</p>	<b>6,94</b>	40,19	-33,25

Раздел	Тема	№ задания	Характеристика задания	% выпол- нивших задание, 2014	% выпол- нивших задание, 2013	При- рост												
			$\neg(x_1 \equiv x_2) \wedge ((x_1 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3)) = 0$ $\neg(x_2 \equiv x_3) \wedge ((x_2 \wedge \neg x_4) \vee (\neg x_2 \wedge x_4)) = 0$ ... $\neg(x_8 \equiv x_9) \wedge ((x_8 \wedge \neg x_{10}) \vee (\neg x_8 \wedge x_{10})) = 0$															
Инфор- маци- онно- ком- муни- каци- онные техно- логии	Техно- логии обра- ботки число- вой ин- форма- ции	В3	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> использование инструментов решения статистических и расчетно-графических задач</p> <p><i>Проверяемые умения:</i> представлять и анализировать табличную информацию в виде графиков и диаграмм</p> <p><i>Пример задания:</i> Дан фрагмент электронной таблицы.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td>А</td> <td>В</td> <td>С</td> </tr> <tr> <td>1</td> <td>8</td> <td></td> <td>6</td> </tr> <tr> <td>2</td> <td>=(B1+1)/(2*A1)</td> <td>=1/(B1+1)</td> <td>=3/(2*B1+C1)</td> </tr> </table> <p>Какое целое число должно быть запи- сано в ячейке В1, чтобы построенная после выполнения вычислений диа- грамма по значени- ям диапазона ячеек А2:С2 соответство- вала рисунку?</p>  <p>Известно, что все значения диапазона, по которым построена диаграмма, имеют один и тот же знак.</p>		А	В	С	1	8		6	2	=(B1+1)/(2*A1)	=1/(B1+1)	=3/(2*B1+C1)	<b>78,03</b>	80,78	-2,75
		А	В	С														
1	8		6															
2	=(B1+1)/(2*A1)	=1/(B1+1)	=3/(2*B1+C1)															
Архи- тектура компь- ютер- ных сетей и техно- логии поиска инфор- мации в сети	В11	<p>Повышенный уровень сложности</p> <p><i>Проверяемые элементы содержания:</i> программная и аппаратная организа- ция компьютеров и компьютерных систем</p> <p><i>Проверяемые умения:</i> работать с распространенными авто- матизированными информационными системами, решать задачи, связанные с адресацией сетей</p> <p><i>Пример задания:</i> В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес. Ад- рес сети получается в результате при- менения поразрядной конъюнкции к заданному IP-адресу узла и маске. По заданным IP-адресу узла и маске определите адрес сети.</p>	<b>57,80</b>	64,27	-6,47													

Раздел	Тема	№ задания	Характеристика задания	% выполнивших задание, 2014	% выполнивших задание, 2013	Прирост																																
			<p>IP-адрес узла: 64.128.208.194            Маска: 255.255.224.0            При записи ответа выберите из приведенных в таблице чисел четыре элемента IP-адреса сети и запишите в нужном порядке соответствующие им буквы без использования точек.</p> <table border="1"> <tr> <td>A</td> <td>B</td> <td>C</td> <td>D</td> <td>E</td> <td>F</td> <td>G</td> <td>H</td> </tr> <tr> <td>0</td> <td>64</td> <td>128</td> <td>192</td> <td>194</td> <td>208</td> <td>224</td> <td>255</td> </tr> </table> <p>Пример.            Пусть искомый IP-адрес: 192.168.128.0, и дана таблица</p> <table border="1"> <tr> <td>A</td> <td>B</td> <td>C</td> <td>D</td> <td>E</td> <td>F</td> <td>G</td> <td>H</td> </tr> <tr> <td>128</td> <td>168</td> <td>255</td> <td>8</td> <td>127</td> <td>0</td> <td>17</td> <td>192</td> </tr> </table> <p>В этом случае правильный ответ будет записан в виде: HBAF</p>	A	B	C	D	E	F	G	H	0	64	128	192	194	208	224	255	A	B	C	D	E	F	G	H	128	168	255	8	127	0	17	192			
A	B	C	D	E	F	G	H																															
0	64	128	192	194	208	224	255																															
A	B	C	D	E	F	G	H																															
128	168	255	8	127	0	17	192																															
		B12	<p>Повышенный уровень сложности  <i>Проверяемые элементы содержания:</i>            использование инструментов поисковых систем (формирование запросов)  <i>Проверяемые умения:</i>            осуществлять поиск и отбор информации  <i>Пример задания:</i>            В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ « », а для логической операции «И» – символ «&amp;». В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.</p> <table border="1"> <thead> <tr> <th>Запрос</th> <th>Найдено страниц (в тысячах)</th> </tr> </thead> <tbody> <tr> <td>хоккей &amp; футбол &amp; волейбол</td> <td>80</td> </tr> <tr> <td>футбол &amp; волейбол</td> <td>260</td> </tr> <tr> <td>хоккей &amp; волейбол</td> <td>230</td> </tr> </tbody> </table> <p>Компьютер печатает количество страниц (в тысячах), которое будет найдено по следующему запросу:  <i>(хоккей   футбол) &amp; волейбол</i>            Укажите целое число, которое напечатает компьютер. Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.</p>	Запрос	Найдено страниц (в тысячах)	хоккей & футбол & волейбол	80	футбол & волейбол	260	хоккей & волейбол	230	40,46	58,77	-15,85																								
Запрос	Найдено страниц (в тысячах)																																					
хоккей & футбол & волейбол	80																																					
футбол & волейбол	260																																					
хоккей & волейбол	230																																					
Основы теории алгоритмов и программирование	Алгоритмы и исполнители	B1	<p>Базовый уровень сложности  <i>Проверяемые элементы содержания:</i>            построение алгоритмов в заданной системе команд исполнителя и практические вычисления  <i>Проверяемые умения:</i>            строить информационные модели объектов, систем и процессов в виде алгоритмов</p>	88,44	89,71	-1,27																																

Раздел	Тема	№ задания	Характеристика задания	% выпол- нивших задание, 2014	% выпол- нивших задание, 2013	При- рост
			<p><i>Пример задания:</i> У исполнителя Квадратор две коман- ды, которым присвоены номера: <b>1. возведи в квадрат,</b> <b>2. прибавь 1.</b> Первая из них возводит число на экране в квадрат, вторая увеличивает его на 1. Запишите порядок команд в программе, которая преобразует чис- ло 1 в число 10 и содержит не более 4 команд. Указывайте лишь номера ко- манд. (Например, программа <b>2122</b> – это про- грамма <b>прибавь 1,</b> <b>возведи в квадрат,</b> <b>прибавь 1,</b> <b>прибавь 1.</b> Эта программа преобразует число 3 в число 18.)</p>			
		В6	<p>Базовый уровень сложности <i>Проверяемые элементы содержания:</i> индуктивное определение объектов (рекурсия) <i>Проверяемые умения:</i> строить информационные модели объектов, систем и процессов в виде алгоритмов <i>Пример задания:</i> Алгоритм вычисления значения функции <math>F(n)</math>, где <math>n</math> – натуральное число, задан следующими соотноше- ниями: <math>F(n) = 1</math> при <math>n \leq 2</math>; <math>F(n) = F(n - 1) + 2 \times F(n - 2)</math> при <math>n &gt; 2</math>. Чему равно значение функции <math>F(7)</math>? В ответе запишите только натураль- ное число.</p>	<b>12,52</b>	66,60	-54,08
		В13	<p>Повышенный уровень сложности <i>Проверяемые элементы содержания:</i> эквивалентные алгоритмические мо- дели <i>Проверяемые умения:</i> строить информационные модели объектов, систем и процессов в виде алгоритмов <i>Пример задания:</i> У исполнителя Удвоитель две коман- ды, которым присвоены номера: <b>1. прибавь 1,</b> <b>2. умножь на 2.</b> Первая из них увеличивает число на экране на 1, вторая удваивает его.</p>	<b>33,33</b>	53,98	-20,65

Раздел	Тема	№ задания	Характеристика задания	% выпол- нивших задание, 2014	% выпол- нивших задание, 2013	При- рост
			Программа для Удвоителя – это последовательность команд. Сколько есть программ, которые число 2 преобразуют в число 22?			
	Программирование	B2	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i></p> <ul style="list-style-type: none"> <li>– типы данных в языке программирования;</li> <li>– основные конструкции языка программирования</li> </ul> <p><i>Проверяемые умения:</i></p> <p>читать и отлаживать программы на языке программирования</p> <p><i>Пример задания:</i></p> <p>Определите значение переменной c после выполнения следующего фрагмента программы. Ответ запишите в виде целого числа.</p> <pre> Паскаль a := 25; b := 12; a := 3 * b - a; if a &gt; b then   c := 2 * a - b else   c := 2 * a + b; </pre>	<b>85,36</b>	83,50	1,86
		B5	<p>Базовый уровень сложности</p> <p><i>Проверяемые элементы содержания:</i></p> <p>основные конструкции языка программирования (циклы)</p> <p><i>Проверяемые умения:</i></p> <p>читать и отлаживать программы на языке программирования</p> <p><i>Пример задания:</i></p> <p>Запишите число, которое будет напечатано в результате выполнения программы.</p> <pre> Паскаль var n, s: integer; begin   n := 0;   s := 0;   while s &lt;= 325 do   begin     s := s + 10;     n := n + 3   end;   write(n) end. </pre>	<b>63,20</b>	70,87	-7,67
		B8	<p>Повышенный уровень сложности</p> <p><i>Проверяемые элементы содержания:</i></p> <p>программы, содержащие циклы и ветвление</p> <p><i>Проверяемые умения:</i></p> <p>читать и отлаживать программы на языке программирования</p>	<b>39,11</b>	40,19	-1,8

Раздел	Тема	№ задания	Характеристика задания	% выпол- нивших задание, 2014	% выпол- нивших задание, 2013	При- рост
			<p><i>Пример задания:</i> Ниже на четырёх языках записан алгоритм. Получив на вход число <math>x</math>, этот алгоритм печатает два числа: <math>a</math> и <math>b</math>. Укажите наименьшее из таких чисел <math>x</math>, при вводе которых алгоритм печатает сначала 13, а потом 5.</p> <pre> Паскаль var x, a, b, c: integer; begin   readln(x);   a := 0; b := 10;   while x&gt;0 do   begin     c := x mod 10;     a := a+c;     if c&lt;b then b := c;     x := x div 10;   end;   writeln(a); write(b); end. </pre>			
		B14	<p>Повышенный уровень сложности <i>Проверяемые элементы содержания:</i> программы с процедурами и функциями <i>Проверяемые умения:</i> читать и отлаживать программы на языке программирования <i>Пример задания:</i> Напишите в ответе число, которое будет напечатано в результате выполнения следующего алгоритма.</p> <pre> Паскаль var a,b,t,M,R :integer; Function F(x:integer):integer; begin   F := 2*(x*x-16)*(x*x-16)+5 end;  begin   a := -11; b := 11;   M := a; R := F(a);   for t := a to b do begin     if (F(t) &lt; R) then begin       M := t;       R := F(t)     end   end;   write(M+6) end. </pre>	<b>26,01</b>	55,15	40,19

Из приведенной таблицы видно, что при выполнении заданий открытой формы части В в разделах «Математические основы информатики» и «Информационно-коммуникационные технологии» экзаменуемые не продемонстрировали высокого уровня знаний. В разделе «Основы теории алгоритмов и программирование» высокий процент выполнимости отметим у заданий базового уровня сложности В1 (88,44%) на построение алгоритма в заданной системе команд исполнителя и В2 (85,36%) на чтение и отладку программы с ветвлением. Выполнимость таких заданий в 2013 году было на том же уровне, что и в 2014.

Стабильным остается уровень выполняемости заданий В9 (66,67%) на оценку реального объекта с помощью модели в виде графа, В3 (78,03%) на представление и анализ табличной информации в виде графиков и диаграмм и В5 (63,20%) на чтение и отладку программы, содержащей цикл.

Однако, в данном разделе задания вызвали у экзаменуемых большие трудности, чем в части А.

В таблице 10 представлены «проблемные темы» освоения разделов учебного предмета «Информатика и ИКТ» при выполнении заданий части В, расположенные в порядке убывания их сложности для учащихся. В таблице выделены те задания, процент выполнения которых составил менее 60.

Таблица 10

**«Проблемные темы» освоения разделов учебного предмета  
«Информатика и ИКТ» (Часть В)**

<b>№ Задания</b>	<b>Спецификация задания</b>	<b>% выполнивших задание</b>
В15	Основы логики <i>Проверяемые элементы содержания:</i> высказывания, логические операции, кванторы, истинность высказывания <i>Проверяемые умения:</i> вычислять логическое значение сложного высказывания по известным значениям элементарных высказываний	6,94
В6	Алгоритмы и исполнители <i>Проверяемые элементы содержания:</i> индуктивное определение объектов (рекурсия) <i>Проверяемые умения:</i> строить информационные модели объектов, систем и процессов в виде алгоритмов	12,52
В14	Программирование <i>Проверяемые элементы содержания:</i> программы с процедурами и функциями <i>Проверяемые умения:</i> читать и отлаживать программы на языке программирования	26,01
В7	Системы счисления <i>Проверяемые элементы содержания:</i> позиционные системы счисления <i>Проверяемые умения:</i> строить информационные модели объектов, систем и процессов в виде алгоритмов	27,36
В13	Алгоритмы и исполнители <i>Проверяемые элементы содержания:</i> эквивалентные алгоритмические модели <i>Проверяемые умения:</i> строить информационные модели объектов, систем и процессов в виде алгоритмов	33,33
В8	Программирование <i>Проверяемые элементы содержания:</i> программы, содержащие циклы и ветвление <i>Проверяемые умения:</i> читать и отлаживать программы на языке программирования	39,11



V12	Архитектура компьютерных сетей и технологии поиска информации в сети <i>Проверяемые элементы содержания:</i> использование инструментов поисковых систем (формирование запросов) <i>Проверяемые умения:</i> осуществлять поиск и отбор информации	40,46
V4	Измерение и кодирование информации, информационные процессы <i>Проверяемые элементы содержания:</i> дискретное (цифровое) представление информации, единицы измерения количества информации <i>Проверяемые умения:</i> оценивать числовые параметры информационных объектов и процессов	42,39
V10	Измерение и кодирование информации, информационные процессы <i>Проверяемые элементы содержания:</i> скорость передачи информации <i>Проверяемые умения:</i> оценивать скорость передачи и обработки информации	52,22
V11	Архитектура компьютерных сетей и технологии поиска информации в сети <i>Проверяемые элементы содержания:</i> программная и аппаратная организация компьютеров и компьютерных систем <i>Проверяемые умения:</i> работать с распространенными автоматизированными информационными системами, решать задачи, связанные с адресацией сетей	57,8

Ниже приведен анализ «проблемных» тем.

**V04.** Измерение и кодирование информации, информационные процессы: дискретное (цифровое) представление информации, единицы измерения количества информации. Процент выполнения – **42,39%** (2013 г. процент выполнения – 50,87%). Наблюдаем незначительное понижение процента выполнения по сравнению с 2013 годом (-8,48%).

Для выполнения задания учащиеся должны владеть тем же материалом, что и для выполнения задания A11 (см. п. 6.1.).

Типичной ошибкой при выполнении данного задания является неверное определение мощности алфавита (числа различных символов, используемых для кодирования).

Для подготовки учащихся к решению таких задач, необходимо сформировать у них основные знания и способы действий (см. A11, п. 6.1.) и подобрать для тренинга задачи разнообразного содержания.

**V06.** Алгоритмы и исполнители: индуктивное определение объектов (рекурсия). Процент выполнения задания очень низкий – **12,52%** (2013 г. процент выполнения – 66,6%). Наблюдаем значительное понижение процента выполняемости по сравнению с 2013 годом (-54,08%).

Для правильного решения задачи необходимо знать:

- рекурсия – способ задания (определения) объекта через обращение к самому себе;
- рекурсивное задание функции – способ задания функции, при

котором значение функции при определенном значении аргументов выражается через уже заданные значения функции при других значениях аргументов;

- чтобы задать функцию рекурсивно, нужно задать условие остановки рекурсии (базовый случай или несколько базовых случаев) и рекуррентную формулу.

При нахождении значения рекурсивно заданной функции самым коротким путем является способ, при котором начинаем вычисления с наименьшего значения и продвигаемся шаг за шагом к нужному значению, используя уже вычисленные на предыдущих шагах числа.

Например:

$$F(n) = 1 \text{ при } n \leq 2;$$

$$F(n) = F(n - 1) + 2 \times F(n - 2) \text{ при } n > 2.$$

Чему равно значение функции  $F(7)$ ?

Решение:

$$F(0)=F(1)=F(2)=1;$$

$$F(3)=F(3-1)+2F(3-2)=F(2)+2 \times F(1)=1+2 \times 1=3;$$

$$F(4)=F(4-1)+2 \times F(4-2)=F(3)+2 \times F(2)=3+2 \times 1=5;$$

$$F(5)=F(5-1)+2 \times F(5-2)=F(4)+2 \times F(3)=5+2 \times 3=11;$$

$$F(6)=F(6-1)+2 \times F(6-2)=F(5)+2 \times F(4)=11+2 \times 5=21;$$

$$F(7)=F(7-1)+2 \times F(7-2)=F(6)+2 \times F(5)=21+2 \times 11=43.$$

Ответ: 43

Типичными ошибками являются арифметические ошибки и «потеря» слагаемого, особенно если экзаменуемый выбирает способ решения – построение рекуррентной формулы, начиная расчет от искомого значения, расписывая его через «младшие» значения и приводя подобные слагаемые, пока не дойдет до выполнения условия выхода из рекурсии. Выполняется так называемый стек.

Пример:

$$\begin{aligned} F(7) &= F(6) + 2 \times F(5) = F(5) + 2 \times F(4) + 2 \times (F(4) + 2 \times F(3)) = F(5) + 4 \times F(4) + \\ &+ 4 \times F(3) = F(4) + 2 \times F(3) + 4 \times (F(3) + 2 \times F(2)) + 4 \times (F(2) + 2 \times F(1)) = \\ &= F(4) + 6 \times F(3) + 12 \times F(2) + 8 \times F(1) = F(4) + 6 \times F(3) + 20 = F(3) + 2 \times F(2) + \\ &+ 6 \times (F(2) + 2 \times F(1)) + 20 = F(2) + 2 \times F(1) + 2 + 18 + 20 = 3 + 2 + 18 + 20 = 43 \end{aligned}$$

Данный способ сложнее и запутанней, при этом на него тратится больше времени, что нежелательно в условиях экзамена.

**В07. Позиционные системы счисления.** Процент выполнения задания низкий – **27,36%** (2013 г. процент выполнения – 75,15%). Произошло очень сильное понижение процента выполняемости по сравнению с 2013 годом, более, чем на 40.

Данное задание повышенного уровня сложности и не решается в одно, два действия.

Для того, чтобы справиться с этим заданием, учащемуся необходимо владеть понятиями: «позиционная система счисления», «разряды», «базис», «основание системы счисления», «сокращенная форма записи числа», «развернутая форма записи числа».

Учащиеся должны уметь легко записывать числа в любой позиционной системе счисления в развернутой форме (расписывая по степеням основания) и знать правило перевода таким способом чисел из любой системы счисления в десятичную.

В данном задании важно иметь навык построения информационных моделей объектов в виде алгоритмов. Безусловно, для того, чтобы справиться с данным заданием на ЕГЭ, учащийся должен иметь опыт выполнения таких заданий и знать приемы рассуждений при решении.

Рассмотрим пример решения такого задания.

*Укажите наименьшее основание системы счисления, в которой запись десятичного числа 30 имеет ровно три значащих разряда.*

Решение:

1. Пусть  $N$  – искомое основание, это может быть только натуральное число. Так как запись числа содержит ровно 3 цифры, то его можно представить так:  $a_2N^2 + a_1N + a_0 = 30$ , где  $a_2, a_1, a_0$  – цифры в новой системе счисления.

2. Из шага 1 очевидно, что  $N^2 \leq 30$ , но явно  $30 < N^3$ , так как число не имеет четвертого разряда, следовательно  $N^2 \leq 30 < N^3$ .

3. Таким образом, нужно найти натуральное число (наименьшее), квадрат которого меньше или равен 30, а куб больше 30. Подходят только 4 и 5:  $4^2 = 16 \leq 30 < 64 = 4^3$  и  $5^2 = 25 \leq 30 < 125 = 5^3$ .

4. Наименьшим из приведенных чисел является 4, таким образом, верный ответ – 4.

Типичными ошибками являются:

- рассмотрение условия ограничения для искомого числа только сверху  $30 < N^3$  и упущение из виду ограничения снизу, тогда как в данном задании важно ограничение именно снизу (ищем наименьшее подходящее);
- иногда учащийся удовлетворяется первым подходящим значением и упускает из вида все остальные подходящие варианты.

При подготовке учащихся следует обратить на это внимание, но прежде всего, естественно важно раскрыть сам прием рассуждений или, подвести учащихся к нему.

**В08. Программирование: циклы и ветвления.** Задание повышенного уровня сложности. Процент выполнения – **31,11%** (2013 г. процент выполнения – 40,19%). Произошло незначительное понижение процента выполняемости по сравнению с 2013 годом (-1,8).

Учащиеся должны знать:

- структуру программы на выбранном языке программирования;
- как работают операторы присваивания, ввода, вывода, циклы и условные операторы в языке программирования;
- арифметические операции и операции целочисленного деления (div) и взятия остатка (mod).

Основной сложностью в задании В08 является то, что здесь нужно не просто выполнить программу по шагам, это как раз не даст ответа, а проде-

лать анализ для выбора подходящего исходного значения. План выполнения задания:

1. Разбираем по шагам принцип работы программы (можно выполнить трассировку на примере любого данного) для определения связи выводимых в конце работы программы чисел с исходным значением.

2. Проводим анализ, каким должно быть входное значение для выполнения указанных условий.

Частыми ошибками экзаменуемых являются:

- путаница в операциях целочисленного деления (div) и взятия остатка (mod) – выполняют с точностью до «наоборот»;

- неумение работать с условием продолжения цикла или принятие цикла с «предусловием» за условный оператор;

- выбор в качестве ответа первого подходящего числа, забыв про то, что оно может быть не одно.

При подготовке учащихся важно не только добиться знания операторов, но и умений читать программу и выполнять её по шагам.

**В10. Измерение и кодирование информации, информационные процессы: скорость передачи информации.** Задание повышенного уровня сложности. Процент выполнения – **52,22%** (2013 г. процент выполнения – 60,19%). Произошло незначительное понижение процента выполняемости по сравнению с 2013 годом (-7,97).

При решении таких задач учащиеся должны знать:

- объем переданной информации  $Q$  вычисляется по формуле  $Q = q \cdot t$ , где  $q$  – пропускная способность канала (скорость), измеряемая в битах в секунду или других единицах, а  $t$  – время передачи;

- при сжатии данных объем информации уменьшается;

- единицы измерения информации (таблица 8).

Задача, в общем, не должна вызывать особых трудностей, если учащиеся хорошо владеют математикой: могут определять проценты от целого в числовом выражении; умеют работать с дробями; умеют работать со степенями. Но именно в этой части допускаются типичные ошибки:

- в таких заданиях используются большие значения и проще работать со степенями, сокращая их в дробях, но учащиеся чаще всего допускают арифметические ошибки в таких расчетах;

- проявляется неумение учащихся находить проценты от целого, например, если при сжатии файла остается 75% от его объема, то учащиеся не видят, что остается  $\frac{3}{4}$  от файла, а если и приходят к этому числу, то часто не умножают на него, а делят;

- забывают прибавить время на сжатие файла и распаковку архива;

- делают ошибки при переводе из одних единиц измерения информации в другие;

- забывают привести к одной единице измерения.

При подготовке учащихся следует учесть данные замечания.

**В11. Архитектура компьютерных сетей и технологии поиска информации в сети: адресация компьютерных сетей.** Процент выполнения задания – **57,8%** (2013 г. процент выполнения – 64,27%). Наблюдаем незначительное понижение процента выполняемости по сравнению с 2013 годом (-6,47%).

Для правильного решения задачи необходимо знать:

- Компьютерная сеть образуется при соединении двух и более компьютеров между собой каналами передачи данных.
- Для создания компьютерной сети требуется специальная аппаратура – сетевое оборудование – и специальные программы – сетевое программное обеспечение.
- Узел – это устройство, соединённое с другими устройствами компьютерной сети. Узлами могут быть компьютеры, специальные сетевые устройства и т.п. Компьютер, являющийся узлом, называют также хостом.
- Протоколами передачи данных называют установленные своды правил по которым передается, получается, кодируется и декодируется информация, например, IP (Internet protocol) – протокол, отвечающий за доставку сообщения по заданному адресу, организует в сети работу с логической адресацией.
- Для того чтобы информация, передаваемая по сети, достигала нужных адресатов, каждый сетевой узел получает уникальный адрес – IP-адрес. IP-адрес представляет собой 32-битное двоичное число, представленное четырьмя группами по 8 бит (октет), разделенных точками, например: 11000000.10101000.01100100.01000101. Для удобства восприятия и сокращения записи принято также использовать десятично-точечную форму представления IP-адресов: каждый октет заменяют соответствующим ему десятичным числом из диапазона [0; 255], разделяя числа точками, так в десятично-точечном представлении IP-адрес, приведенный выше, запишется: 192.168.100.69.
- IP-адрес сетевого узла состоит из двух логических частей – номера сети, которой он принадлежит, и номера узла в этой сети. По существующему соглашению IP-адрес узла занимает 32 бита, при этом его старшие биты хранят номер сети, а младшие – номер узла в сети. Значения номеров сетей и узлов, все биты которых равны 0 или все биты которых равны 1 не используются для задания IP-адресов, они зарезервированы для специальных целей. Если, например, на номер узла в сети выделен один байт (8 бит), то номера узлов могут принимать значения от 1 (00000001) до 254 (11111110), всего  $2^8 - 2 = 254$  значения.
- В 1985 году в двухуровневую иерархию IP-адресов (сеть – узел) была введена новая составляющая – подсеть.

<b>Двухуровневая структура IP-адресов</b>		
Номер сети	Номер узла (хоста)	
<b>Трёхуровневая структура IP-адресов</b>		
Номер сети	Номер подсети	Номер узла (хоста)

Ниже представлен IP-адрес 192.168.19.132, в котором два левых бита

номера узла в сети отведены на определение подсети.

Номер сети	Подсеть	Номер узла в сети
1 1 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 1 0 0 1 1	1 0	0 0 0 1 0 0
три байта	2 бита	6 бит

При таком распределении битов адреса сеть может быть разбита на 4 подсети, в каждой из которых может быть  $2^{6-2} = 62$  узла.

<b>Подсеть 1: 192.168.19.0</b>
1 1 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0
<b>Подсеть 2: 192.168.19.64</b>
1 1 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 1 0 0 1 1 0 1 0 0 0 0 0 0
<b>Подсеть 3: 192.168.19.128</b>
1 1 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 1 0 0 1 1 1 0 0 0 0 0 0 0
<b>Подсеть 4: 192.168.19.192</b>
1 1 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 1 0 0 1 1 1 1 0 0 0 0 0 0

Эти четыре подсети могут иметь следующие действующие адреса узлов:

- 192.168.19.1 – 192.168.19.62
- 192.168.19.65 – 192.168.19.126
- 192.168.19.129 – 192.168.19.190
- 192.168.19.193 – 192.168.19.254

Напомним, что двоичные адреса узлов с одними только единицами или нулями зарезервированы, поэтому нельзя использовать адреса со следующими числами в последнем октете: 0, 63, 64, 127, 128, 191, 192 или 255.

- Для того чтобы определить, какая часть адреса относится к сети, а какая – к узлу, используется маска подсети.

- Маска подсети – 32-хразрядное двоичное число, позволяющее определить, какая часть IP-адреса узла сети относится к адресу сети, а какая к адресу самого узла в этой сети. В маске подсети старшие биты, отведенные в IP-адресе для номера сети и подсети, имеют значения равные единице; младшие биты, отведенные в IP-адресе для номера узла в подсети, имеют значения равные нулю. Чередование нулей и единиц в маске запрещено.

- Для получения адреса сети при известных IP-адресе и маске подсети, к ним применяется операция поразрядной конъюнкции (поразрядное логическое И). Биты IP-адреса, которым соответствуют единичные биты маски, определяют адрес сети; нулевые биты – адрес узла в сети. В пределах одной подсети маски подсети должны совпадать на всех узлах.

Для выполнения задания В11 учащемуся необходимо хорошо владеть представленным выше материалом и уметь выполнять поразрядную конъюнкцию двоичных кодов.

Пример 1.

Заданы маска подсети 255.255.255.192 и адрес узла 192.168.15.137.

Адрес сети равен \_\_\_\_\_.

Решение:

Переведем октеты маски и адреса узла в двоичные коды. Для определения адреса сети выполним побитовую конъюнкцию.

Маска	11111111 11111111 11111111 11000000
Узел	11000000 10101000 00001111 10001001
Поразрядная конъюнкция	11000000 10101000 00001111 10000000

Адрес сети – 192.168.15.128

Адрес узла определяется последними шестью битами (нули в маске) и равен 9.

Ответ: 192.168.15.128

Для экономии времени стоит помнить, что 255 при переводе в двоичную систему счисления превратится в восемь единиц, а при побитовой конъюнкции этого октета на какое-то число сохранится без изменения это число. Таким образом, первые три части адреса сети можно выписать сразу из адреса узла (первые три числа), а переводить в двоичную систему счисления и выполнять побитовую конъюнкцию остается только с четвертыми октетами 192 и 137.

Пример 2.

*Заданы маска подсети 255.255.255.192 и адрес узла 192.168.15.137. Непосредственный адрес узла в сети равен \_\_\_\_\_.*

Решение:

Переведем октеты маски и адреса узла в двоичные коды.

Маска	11111111 11111111 11111111 11000000
Узел	11000000 10101000 00001111 10001001

Адрес узла определяется последними шестью битами (нули в маске) и равен 9.

Ответ: 9

Общий алгоритм выполнения задания В11:

- 1) Переводим каждый октет маски в двоичную систему счисления.
- 2) Переводим каждый октет IP-адреса в двоичную систему счисления.

**Если число в двоичной системе счисления получилось маленьким (меньше 8 знаков), значит добавляем слева нули, пока число не будет с 8 знаками.**

3) Выполняем конъюнкцию соответствующих элементов.

4) Получившиеся октеты переводим обратно в десятичную систему счисления.

5) По таблице ищем совпадения и выписываем соответствующие буквы. Их и записываем в ответ.

Типичные ошибки учащихся. В результате анализа ответов учащихся, напрашивается вывод: те, кто принимался за выполнение этого задания либо отвечали правильно (около 50%), либо выбирали ответ наугад (несовпадающие варианты ответов).

Те, кто давал близкие к правильному ответы делали следующие ошибки:

- арифметические ошибки при переводе октетов в двоичные коды;

- в качестве адреса сети брали только октеты IP-адреса узла, соответствующие 255 в маске и отбрасывали остальные числа (заменяя нулями), т.о. не выполнялась полностью побитовая конъюнкция;

- путали понятия адрес сети и адрес узла в сети.

**В12. Архитектура компьютерных сетей и технологии поиска информации в сети: использование инструментов поисковых систем (формирование запросов).** Процент выполнения задания – **40,46%** (2013 г. процент выполнения – 58,77%). Наблюдаем понижение процента выполняемости по сравнению с 2013 годом (-15,85%).

Для правильного выполнения задания учащимся следует знать:

- Поисковая система – это комплекс программ и мощных компьютеров, автоматически просматривающих ресурсы Интернет и индексирующих их содержание.

- Поисковые системы обычно состоят из трех компонентов:

- поисковый робот, который перемещается по сети и собирает информацию;

- база данных, которая содержит постоянно обновляемую информацию о местонахождении веб-страниц и файлов на сотнях миллионов серверов Интернета, собираемую роботом;

- поисковый механизм, который используется как интерфейс для взаимодействия с базой данных.

- Поисковые роботы – это специальные программы, которые занимаются поиском страниц в сети, извлекают гипертекстовые ссылки на этих страницах и автоматически индексируют информацию, которую они находят для построения базы данных.

- Поисковые системы осуществляют поиск информации по ключевым словам. Ключевым словом (Keyword) документа называется слово или словосочетание, которое отражает содержание данного документа. Как правило, для успешного запроса надо ввести несколько слов для уточнения поиска.

- Службы поиска позволяют связывать ключевые слова логическими операциями и, или, искать цитаты, исключать некоторую информацию из поискового запроса и т.д. Все эти возможности определяются языком запросов конкретной поисковой системы.

Учащиеся должны знать логические операции, их обозначения и приоритет выполнения. Таким образом, с одной стороны мы имеем дело с алгеброй логики. С другой стороны мы имеем дело с понятием «множество» (речь идет о множествах страниц). Соответственно необходимы знания теории множеств, операций над множествами. Полезно владеть приемами Эйлера – применение графического метода построения модели.

Рассмотрим пример выполнения задания.

*В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.*



Запрос	Найдено страниц (в тысячах)
хоккей & футбол & волейбол	80
футбол & волейбол	260
хоккей & волейбол	230

Компьютер печатает количество страниц (в тысячах), которое будет найдено по следующему запросу:

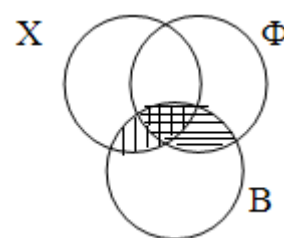
*(хоккей | футбол) & волейбол*

Укажите целое число, которое напечатает компьютер. Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Решение 1:

1. Построим диаграмму Эйлера-Венна, обозначив области начальными буквами слов «хоккей», «футбол», «волейбол» (X, Ф, В).

2. Множество элементов, удовлетворяющих запросу *хоккей & футбол & волейбол* равно  $X \cap \Phi \cap B$ , его мощность известна – 80. На диаграмме множеству соответствует область заштрихованная клеткой.



3. Множество элементов, удовлетворяющих запросу *футбол & волейбол* равно  $\Phi \cap B$ , его мощность известна – 260. На диаграмме – область заштрихованная горизонтальной линией.

4. Множество элементов, удовлетворяющих запросу *хоккей & волейбол* равно  $X \cap B$ , его мощность известна – 230. На диаграмме – область заштрихованная вертикальной линией.

5. Множество элементов, мощность которого нас интересует, удовлетворяющего запросу *(хоккей | футбол) & волейбол* равно  $(X \cup \Phi) \cap B$ . На диаграмме ему соответствует вся заштрихованная область.

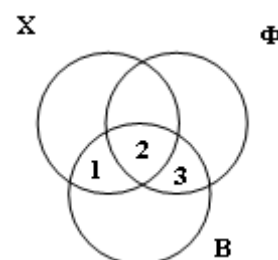
6. Если сложить мощности множеств  $\Phi \cap B$  и  $X \cap B$ , то дважды посчитаются элементы множества  $X \cap \Phi \cap B$ , вычтя их, получим мощность искомого множества:  $260+230-80=410$ .

Ответ: 410

Решение 2:

Используем другие обозначения.

1. Построим диаграмму Эйлера-Венна, обозначив области начальными буквами слов «хоккей», «футбол», «волейбол» (X, Ф, В).



2. Обозначим области, которые соответствуют каждому запросу

Запрос	Области	Количество страниц (тыс.)
<i>хоккей &amp; футбол &amp; волейбол</i>	2	80
<i>футбол &amp; волейбол</i>	2+3	260
<i>хоккей &amp; волейбол</i>	1+2	230
<i>(хоккей   футбол) &amp; волейбол</i>	1+2+3	?

3. Из таблицы следует, что в суммарный результат второго и третьего запросов область 2 входит дважды ( $2+3+1+2$ ), поэтому, удалив результат

первого запроса (2) получим  $(2+3+1+2-2=1+2+3)$ , т.е. находим результат четвертого:  $260+230-80=410$

4. Таким образом, ответ – 410

Типичные ошибки участников ЕГЭ:

- путают конъюнкцию и дизъюнкцию, и, соответственно их таблицы истинности;
- при использовании кругов Эйлера путают какими цифрами обозначены результаты каких запросов;
- неверно расставляют приоритет операций.

**В13.** Алгоритмы и исполнители: эквивалентные алгоритмические модели. Процент выполнения задания – **33,33%** (2013 г. процент выполнения – 53,98%). Наблюдаем значительное понижение процента выполняемости по сравнению с 2013 годом (-20,65%).

В данном задании требуются умения анализировать результат исполнения алгоритма и графически визуализировать действия исполнителя.

Выполняется задание методом динамического программирования – это способ решения сложных задач путем сведения их к более простым задачам того же типа. С помощью динамического программирования решаются задачи, которые требуют полного перебор вариантов:

- «подсчитайте количество вариантов...»
- «как оптимально распределить...»
- «найдите оптимальный маршрут...»

Динамическое программирование позволяет ускорить выполнение программы за счет использования дополнительной памяти; полный перебор не требуется, поскольку запоминаются решения всех задач с меньшими значениями параметров.

При выполнении задания можно использовать:

- дерево;
- рекуррентную формулу;
- таблицу.

При использовании дерева по левой ветви можно выполнять одну операцию (например, +1), а по правой другую (например, ×3). В узлах дерева записываются получаемые числа, при этом в корне – исходное число. Использование данного способа позволяет выполнять как раз полный перебор.

Выстраивая дерево и анализируя, по какой ветви можно продвигаться, для получения нужного числа, а по какой это невозможно мы можем получить все возможные пути и посчитать их количество.

Главный минус этого способа – громоздкость (особенно если необходимо выполнять более двух операций).

Применение рекуррентной формулы рассмотрим на примере.

*У исполнителя Утроитель две команды, которым присвоены номера:*

*1. прибавь 1*

*2. умножь на 3*

*Первая из них увеличивает число на экране на 1, вторая – утраивает*

его. Программа для Утроителя – это последовательность команд.

Сколько есть программ, которые число 1 преобразуют в число 14?

Решение:

1. Обозначим  $K_N$  количество разных программ для получения числа N из 1. Ясно, что  $K_2 = 1$ , так как получить 2 из 1 можно только способом 1+1. А вот  $K_3 = 2$ , так как число 3 можно получить из числа 2 прибавив 1 и из числа 1, умножив его на 3. Таким образом, всего два способа.

2. Итак, если число N не делится на 3, то оно могло быть получено только последней операцией сложения, поэтому  $K_N = K_{N-1}$ . Если же N делится на 3, то последней командой может быть как сложение, так и умножение поэтому для получения  $K_N$  нужно сложить  $K_{N-1}$  (количество программ с последней командой сложения) и  $K_{N/3}$  (количество программ с последней командой умножения). В итоге получаем:

- если N не делится на 3:  $K_N = K_{N-1}$ ;
- если N делится на 3:  $K_N = K_{N-1} + K_{N/3}$ ;
- начальное условие:  $K_1 = K_2 = 1$ .

3. Начинаем с заданного конечного числа 14: применяем первую формулу ( $K_N = K_{N-1}$ ), пока не дойдем до числа, делящегося на 3 (это 12):

$$K_{14} = K_{13} = K_{12}.$$

Далее применяем вторую формулу ( $K_N = K_{N-1} + K_{N/3}$ ):

$$K_{14} = K_{13} = K_{11} + K_4.$$

Применяем формулы для каждого слагаемого и так далее, получим:

$$K_{14} = K_{11} + K_4 = K_9 + K_3 = K_8 + K_3 + K_2 + K_1 = K_6 + K_2 + K_1 + 2 = K_5 + K_2 + 4 = K_3 + 5 = K_2 + K_1 + 5 = 7$$

4. Ответ – 7.

Табличный способ позволяет более компактно записать промежуточные значения рекуррентной формулы.

Первый и второй шаги те же, что и в предыдущем решении, а далее заполняем таблицу для всех значений от 1 до N:

N	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$K_N$	1	1	2	2	2	3	3	3	5	5	5	7	7	7

Ответ – 7.

Типичные ошибки – те же, что и в задании В6: невнимательность, арифметические ошибки.

Кроме того, при составлении рекуррентной формулы могут быть неправильно определены начальные условия или ошибочно составлена сама формула.

**В14. Программирование: программы с процедурами и функциями.** Задание повышенного уровня сложности. Процент выполнения – **26,01%** (2013 г. процент выполнения – 55,15%). Очень большое расхождение с 2013 годом: процент выполняемости задания понизился по сравнению с 2013 годом на 40.

Учащиеся должны знать:

- структуру программы на выбранном языке программирования;
- как работают основные операторы: присваивания, ввода, вывода, циклы и условные операторы в языке программирования;
- арифметические операции и операции целочисленного деления (div) и взятия остатка (mod);
- функция – это вспомогательный алгоритм, который возвращает некоторое значение – результат;
- в Паскале функция располагается выше основной программы и оформляется следующим образом (вместо многоточия могут быть любые операторы):

```
function F(x: integer):integer;
begin
  ...
  F:= <результат функции>
end;
```

- в заголовке функции записывают имя функции, в скобках – список параметров, далее через двоеточие – тип возвращаемого значения; в приведенном примере функция **F** принимает один целый параметр, к которому внутри функции нужно обращаться по имени **x**, и возвращает целое число;
- результат функции записывается в специальную переменную, имя которой совпадает с именем функции; объявлять эту переменную не нужно – тип описывается в заголовке функции;
- в основной программе обращение к функции выполняется через её имя и перечисление значений для передачи параметрам;
- при вызове функции расчет значения происходит по алгоритму, описанному отдельно в разделе function, а результат возвращается в основную программу для дальнейших вычислений по основному алгоритму.

Частыми ошибками экзаменуемых являются:

- неумение работать с циклом, неправильное определение конечного значения параметра цикла;
- неправильная работа с параметрами функции, неверные значения для параметра;
- не учитываются отрицательные числа, свойство четности функции.

Основная сложность – проанализировать функцию, найти её минимумы (максимумы).

При подготовке учащихся важно не только добиться знания операторов, но и уметь читать программу и выполнять её по шагам, а так же проводить анализ функции.

**В15. Основы логики: высказывания, логические операции, кванторы, истинность высказывания.** Процент выполнения очень низкий – **6,94%** (2013 г. процент выполнения – 40,19%). Очень большое расхождение с 2013 годом: процент выполнения задания понизился по сравнению с 2013 годом более, чем

на 33.

Это задание высокого уровня сложности, традиционно вызывающее трудности у экзаменуемых.

При его выполнении требуется уверенное владение алгеброй логики, знание комбинаторики. Необходим опыт анализа логических выражений и математического доказательства существования решений логического уравнения или системы уравнений.

Но казуально то, что на его выполнение рекомендуется затрачивать около 10 минут и дается всего 1 первичный балл. Поэтому очень высокий процент выпускников, не бравшихся за выполнение этого задания (26,2%).

Рассмотрим способы и примеры выполнения задания В15.

Задание:

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

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

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

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

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

Решение 1. (Метод логических рассуждений и комбинаторика)

Рассмотрим первое уравнение  $(x_1 \equiv x_2) \wedge (x_3 \equiv x_4) = 1$ , оно имеет решение когда одновременно истинны  $(x_1 \equiv x_2)$  и  $(x_3 \equiv x_4)$ .  $x_1 \equiv x_2$  истинно при одинаковых значениях  $x_1$  и  $x_2$ : всего таких наборов два – 11 и 00. То же самое верно и для выражения  $x_3 \equiv x_4$ . Таким образом, первому уравнению удовлетворяет всего  $2 \times 2 = 4$  набора: 1111, 1100, 0011 и 0000.

Аналогичные рассуждения верны при изучении второго уравнения  $(x_3 \equiv x_4) \wedge (x_5 \equiv x_6) = 1$ . Только в полученных выше 4-х наборах решений, значения для  $x_3$  и  $x_4$  уже зафиксированы и добавляются только наборы допустимых значений для  $x_5$  и  $x_6$ , а их только 2: 11 и 00. Тогда всего возможных решений для системы из первых двух уравнений получается  $4 \times 2 = 8$ .

Рассуждая аналогично, получим  $8 \times 2 = 16$  различных допустимых наборов значений переменных  $x_1, x_2, \dots, x_8$ . Соответственно получаем  $16 \times 2 = 32$  различных набора значений переменных  $x_1, x_2, \dots, x_{10}$  при которых выполнены все четыре равенства.

Ответ: 32.

Минус такого приема, нет единого способа рассуждений. Каждая система логических уравнений уникальна, и закономерности выявить бывает затруднительно. Здесь и встречаются основные ошибки – неверно установленные закономерности, пропущенные решения, неполное или неверное рассуждение. Часто уравнения сложны и объемны, стоит предварительно их упростить. На этом этапе необходимо хорошо владеть законами логики.

Найти наборы значений переменных, удовлетворяющих первому уравнению можно и просто, построив таблицу истинности.

Полезно знать дополнительно к основным законам логики законы отрицания и устранения импликации и такие тождества:

$$A \equiv B = \bar{A} \wedge \bar{B} \vee A \wedge B$$

$$A \equiv B = A \rightarrow B \wedge B \rightarrow A$$

Решение 2. (Воспользуемся деревом).

Рассуждения при построении дерева проводим аналогично приведенным в первом решении.

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$		
1	1	1	1	1	1	1	1	1	1		
						0	0	0	0		
				0	0	1	1	1	1		
						0	0	0	0		
		0	0	0	0	1	1	1	1	1	1
								0	0	0	0
						0	0	1	0	1	1
								0	0	0	0
	0		0	0	0	1	1	1	1	1	1
								0	0	0	0
						0	0	1	1	1	1
								0	0	0	0
		0	0	0	0	1	1	1	1	1	1
								0	0	0	0
						0	0	1	1	1	1
								0	0	0	0
2 решения		4 решения		8 решений		16 решений		32 решения			

Ответ: 32.

Решение 3. (Метод отображений).

На начальном этапе, при разборе первого уравнения проводим рассуждения, аналогичные решению 1. Можно построить таблицу истинности.

Далее замечаем, что пара  $x_1 x_2 = 11$  «порождает» для  $x_3 x_4$  две пары: 11 и 00 или будем говорить «отображается». Пара  $x_1 x_2 = 00$  «порождает» для  $x_3 x_4$  так же две пары: 11 и 00. А вот пары 01 и 10 значений переменных  $x_1$  и  $x_2$  «порождают» для  $x_3 x_4$  нуль решений. Запишем это схематично так:

$x_1 x_2$		$x_3 x_4$
11		11
10		10
01		01
00		00

или

	$x_1 x_2$	$x_3 x_4$
11	1	2
10	0	0
01	0	0
00	1	2

Заметим, что во втором и далее уравнениях вторая пара переменных из предыдущего уравнения по той же схеме «порождает» наборы для новой пары переменных. Составим полную схему отображений:

	$x_1 x_2$	$x_3 x_4$	$x_5 x_6$	$x_7 x_8$	$x_9 x_{10}$
11	1	2	4	8	16
10	0	0	0	0	0
01	0	0	0	0	0
00	1	2	4	8	16
Итого:	2	4	8	16	32

Ответ: 32.

Метод отображений наиболее универсальный и демонстрирует компактный способ изображения решений. Однако, сложность опять же остается в том, чтобы упростить логические выражения и увидеть закономерности отображения. Кроме того, «отображений» может быть больше. Могут «отображаться» не только пары, но и тройки и четверки и т.д. переменных. К тому же в зависимости от системы логических уравнений отображений могут повторяться не на каждую пару (тройку и пр.) значений, а чередоваться через шаг и т.п.

### Выводы по выполнению заданий части В

1. В результате анализа выполнения заданий второй части ЕГЭ по информатике и ИКТ в 2014 году можно констатировать слабое владение даже на базовом уровне абитуриентами темой «Измерение и кодирование информации, информационные процессы». Что подтверждает выводы по первой части ЕГЭ.

2. Низкий уровень владения демонстрируют экзаменуемые 2014 года по теме «Системы счисления» при выполнении заданий повышенной сложности. Соответственно для изучающих предмет «Информатика и ИКТ» на повышенном уровне стоит больше уделять внимания таким заданиям по этой теме.

3. В подтверждение выводам по части А по теме «Основы логики» с увеличением уровня сложности резко падает уровень владения знаниями и методами решения задач. Особое внимание следует обратить на овладение законами логики, приемами упрощения логического выражения, решению систем логических уравнений, способам логических доказательств, комбинаторике. В первую очередь это необходимо учащимся, стремящимся получить максимальные баллы по ЕГЭ.

4. Как и в части А, в части В выпускники 2014 года продемонстрировали ровные и уверенные знания на базовом уровне сложности по разделу «информационно-коммуникационные технологии» в части технологии обработки числовой информации. А вот в части архитектуры компьютерных сетей и технологии поиска информации в сети демонстрируется низкий уровень знаний.

5. По теме «Алгоритмы и исполнители» есть пробелы в знаниях по рекуррентным вычислениям. Следовательно, при изучении информатики в школе на повышенном уровне стоит усилить внимание работе с рядами данных, итерационными процессами, рекуррентными формулами, рекурсией.

6. В теме «Программирование» проблемы экзаменуемые испытывают при работе с программами, содержащими циклы и подпрограммы (функции). Соответственно, в углубленном курсе «Информатика и ИКТ» необходимо усилить роль задач, требующих выполнения и отладки таких программ.

7. Твердые знания, как на базовом, так и на повышенном уровне показали экзаменуемые по теме «моделирование».

### 6.3. Анализ выполнения части С

Третья часть экзаменационной работы ЕГЭ по информатике и ИКТ содержит четыре задания, требующих подробного ответа в развернутой форме. Задания направлены на проверку умений анализировать текст программы, находить и исправлять ошибки, разрабатывать алгоритмы решения задач с доказательством правильности решения и, в конечном итоге, писать программы на языке программирования.

В сравнении с прошлым 2013 годом необходимо отметить некоторое повышение качества подготовки учащихся к разработке алгоритмов и программ. Не будем забывать, что это учащиеся, освоившие углубленный курс «Информатика и ИКТ». Сравнительный анализ выполнения учениками заданий части С на максимальный балл представлен в таблице 11.

Таблица 11

#### Выполнение заданий с развёрнутым ответом

№ задания	Спецификация задания	% выполнивших задание полностью, 2014 г.	% выполнивших задание полностью, 2013 г.	Прирост
C1	Программирование <i>Проверяемые элементы содержания:</i> основные конструкции языка программирования <i>Проверяемые умения:</i> читать и отлаживать программы на языке программирования	<b>24,28</b>	33,98	-9,7
C2	Программирование <i>Проверяемые элементы содержания:</i> построение алгоритмов и практические вычисления <i>Проверяемые умения:</i> создавать программы на языке программирования по их описанию	<b>31,21</b>	23,88	7,33
C3	Алгоритмы и исполнители <i>Проверяемые элементы содержания:</i>	<b>38,54</b>	32,82	5,72



№ задания	Спецификация задания	% выпол- нивших задание полностью, 2014 г.	% выпол- нивших задание полностью, 2013 г.	Прирост
	разработка выигрышной стратегии, цепочки (конечные последовательности), деревья, списки, графы, матрицы (массивы), псевдослучайные последовательности <i>Проверяемые умения:</i> строить информационные модели объектов, систем и процессов в виде алгоритмов			
С4	Программирование <i>Проверяемые элементы содержания:</i> основные этапы разработки программ, разбиение задачи на подзадачи <i>Проверяемые умения:</i> создавать программы на языке программирования по их описанию	1,35	5,83	-4,48

Как видим, с заданиями С2 и С3 в 2014 году учащиеся справились лучше, по сравнению с 2013 годом.

На наш взгляд, снижение баллов в задании С1 связано с новыми требованиями к оцениванию, введёнными в 2014 году, по которым снижается балл при указании ложных ошибок в программе. В 2013 году при оценивании данного задания ложно названные ошибки не учитывались, баллы выставлялись за каждое верно выполненное действие.

Задание С4 является самым трудным и за его выполнение брались лишь те экзаменуемые, кто претендовал на максимальный балл ЕГЭ (всеми приступили к выполнению чуть более 20%). Понижение процента выполнения на максимальный балл может быть причиной того, что в 2013 году была утечка информации о содержании КИМ ЕГЭ, и ответах к заданиям в СМИ. Стоит отметить, что в прошлом году большой процент заданий С4 были выполнены как «под копирку» и полностью совпадали с примерами решений в рекомендациях к оцениванию для экспертов. В 2014 году ответы экзаменуемых были уникальными, что, порой, усложняло проверку (в случаях представления сложных, неверных, запутанных алгоритмов, которые, все же, могли соответствовать критериям на 1 балл).

Третья часть экзаменационной работы (часть С) является важной и весомой, ведь за неё выпускник получает максимально 30% баллов от возможных максимальных 40 первичных тестовых баллов. Поэтому выполнение данного вида работы особенно важно для получения высокой отметки.

## 7. Типичные ошибки, допущенные участниками экзамена при выполнении заданий третьей части (С)

### 7.1. Задание С1

В задании С1 проверяются умения выполнять программу по шагам, находить и исправлять ошибки. Рекомендуемое время на выполнение – 30 минут.

Рассмотрим пример задания С1 ЕГЭ по информатике и ИКТ 2014 года с решением. В вариантах задания примеры программ приводятся на Бейсике, Паскале, Си и Алгоритмическом языке. Мы будем рассматривать примеры программ на Паскале и Си как наиболее распространенных при выполнении работ экзаменующимися.

**Пример формулировки задания С1.** На обработку поступает последовательность из четырёх неотрицательных целых чисел (некоторые числа могут быть одинаковыми). Нужно написать программу, которая выводит на экран сумму всех чисел исходной последовательности, которые делятся на 4, и количество таких чисел. Если в последовательности нет чисел, которые делятся на 4, то на экран нужно вывести «NO». Известно, что вводимые числа не превышают 1000.

Программист написал программу неправильно.

Си	Паскаль
<pre>#include &lt;stdio.h&gt; #define n 4  void main(void) {     int i, x, sum, count;     count = 0;     sum = 0;     for (i = 1; i &lt;= n; i++)     {         scanf("%d", &amp;x);         if (x % 4 == 0)         {             count++;             sum = sum + i;         }     }     if (sum &gt; 0)     {         printf("%d\n", count);         printf("%d\n", sum);     }     else         printf("NO\n"); }</pre>	<pre>const n=4; var i, x, sum, count: integer; begin     count := 0;     sum := 0;     for i := 1 to n do     begin         read(x);         if x mod 4 = 0 then         begin             count := count + 1;             sum := sum + i         end     end;     if sum &gt; 0 then     begin         writeln(count);         writeln(sum)     end     else         writeln('NO')     end.</pre>

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе последовательности: 20, 93, 40, 39

2. Приведите пример последовательности, в которой есть хотя бы одно число, делящееся на 4, и при которой, несмотря на ошибки, программа печатает правильный ответ.

3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и

может быть исправлена без изменения других строк. Для каждой ошибки:

- 1) выпишите строку, в которой сделана ошибка;
- 2) укажите, как исправить ошибку, т.е. приведите правильный вариант строки.

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в имеющейся программе, а не написать свою, возможно, использующую другой алгоритм решения.

Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

Примечание: 0 делится на любое число.

**Пример содержания верного ответа.** Решение использует запись программы на Паскале. Допускается использование программы на трёх других языках.

1. Программа выведет два числа: 2 и 4.

2. Пример последовательности, содержащей числа, делящиеся на 4, для которой программа работает правильно: 1 2 3 4.

Замечание для проверяющего. В конце работы программы значение переменной `sum` равно сумме номеров позиций, на которых стоят числа, которые делятся на 4, или 0, если в последовательности нет таких чисел.

Соответственно, программа будет работать верно, если в последовательности сумма чисел, которые делятся на 4, равна сумме номеров позиций, на которых эти числа стоят. Значение переменной `count` будет правильным в любом случае.

3. В программе есть две ошибки.

Первая ошибка: неверное присваивание при вычислении текущей суммы.

Строка с ошибкой:

```
sum := sum + i
```

Вторая ошибка: неверная проверка наличия чисел, которые делятся на 4.

Строка с ошибкой:

```
if sum > 0 then
```

Верное исправление:

```
sum := sum + x
```

Верное исправление:

```
if count > 0 then
```

## Основные ошибки в решениях задачи C1

**1. Частая ошибка – неправильная проверка на кратность (нахождение искомого числа), пропуск или неверное исправление ошибки проверки условия нахождения искомого числа.**

*Пример 1*

*C1*

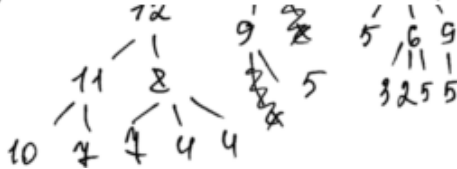
*1 При вводе последовательности: 20; 93; 40; 39.  
программа выведет числа 2 и 4*

*2. 7; 0; 22; 4. - при вводе этой последовательности  
программа выведет правильный ответ 2 и 4.*

### 3. Ошибка в программе в строке:

$sum := sum + i$  - программа прибавляет к сумме не число, а его номер (т.е. 1, 2, 3 или 4).

правильный вариант строки:  $sum := sum + x$



### Пример 2.

С1. 1) При входе последовательности 20, 93, 40, 39 программа выведет числа 2 и 4 (2-кол-во чисел кратных 4, 4- сумма порядкового номера этих чисел).

2) Несмотря на ошибки программа выведет правильный ответ при последовательности: 1, 3, 5, 4. Ответ будет: 1; 4.

3) Ошибки допущены в строках:

$sum := sum + i$ , заменить на  $sum := sum + x$ ;

$if\ sum > 0\ then$ , заменить на  $if\ sum > -1\ then$

т.к. при последовательности 0, 0, 0 - все числа кратны 4 и их сумма равна 0.

**Комментарий:** В примере 1 второе действие выполнено неправильно, так как сумма индексов делящихся на 4 чисел из приведенной последовательности 7, 0, 22, 4 равна не 4, а 6. Ноль делится на любое число, об этом есть напоминание в условии. Так, что выведется программой 2 и 6, что не является правильным исполнением.

Правильно в примере выполнено первое действие и найдена и исправлена первая ошибка. Вторая ошибка не найдена ( $if\ sum > 0$ ), а здесь опять же может быть найдено делящееся на 4 число 0, а программа выведет при такой проверке «NO». Работа оценена в соответствии с критериями в 1 балл (по критериям прошлого года было бы 2 балла).

Во втором примере неправильно исправлена вторая ошибка. Оцениваются 3 правильно выполненных действия из 4-х при отсутствии неверно названных ошибок в 2 балла.

### 2. Типичная ошибка – ошибка инициализации максимума (минимума, счетчика и др.).

#### Пример 1

С1 1. 3 9 (count = 3; minimum = 9)

2. 3333

3.  $if\ x\ mod\ 2 \leq 0\ then$  →  $if\ x\ mod\ 2 = 1\ then$   
 $if\ x > minimum\ then$  →  $if\ x < minimum\ then$

### Пример 2

1)  $minimum = 2943$   
 $count = 2 \& 9$  —

2) ~~10, 9, 129~~ —

3) -  $minimum := 1$  *надо  $minimum := 9$*   
-  $if\ x > minimum\ then\ minimum := x$  *if  $x\ mod.\ 10 < minimum$   
 $minimum := x\ mod.\ 10$*   
-  $head(x)$  - *написано* *надо for n:=1 to n do*  
-  $minimum := x$   
*надо  $x := x\ div\ 10$*

### Комментарий:

В примере 1 автор работы не увидел ошибки неверной инициализации минимума, зато неверно назвал строку с проверкой числа на нечетность ошибочной, кроме этого в данной работе неправильно выполнена трассировка программы (действие 1). Итог: только 2 правильно выполненных действия из четырех – это 1 балл.

В примере 2 в третьем действии неверно предложено исправление инициализации минимума. Автор работы не понимает, что первоначально в качестве минимального берем заведомо большее из возможных чисел. В данном случае  $minimum$  может быть любое целое число, большее или равное 999, либо  $MAXINT$ . К слову в этой работе вообще нет ни одного правильно выполненного действия. Оценка – 0 баллов.

### 3. Неправильное выполнение и отладка программы.

#### Пример 1

(1) при вводе последовательности: 20, 93, 40, 39  
программа выдает:

2

6

2) Пример последовательности 2, 53, 4, 71

3)  $sum := sum + i$  |  $if\ sum > 0\ then$   
 $\downarrow$  |  $\downarrow$   
 $sum := sum + x$  |  $if\ sum >= 0\ then$

C2

~~$S := 0$~~   
~~for i:=1 to N do~~

$S := 0;$   
for i:=1 to N do  
if  $a[i] \bmod 2 <> 0$  then  
 $S := a[i] + S;$   
writeln(S)

Комментарий: В примере 1 неправильно выполнена трассировка программы (действие 1), пример последовательности чисел, для которой программа работает верно, тоже приведен неправильно, следовательно, автор

работы не может отследить, что присваивается идентификатору sum и/или параметру i. Кроме того, в работе неверно исправлена вторая ошибка. Итог: только 1 правильно выполненное действие из четырех – это 0 баллов.

#### 4. Указание вместо ошибочной верной строки программы.

Пример 1

С1. ① Три вводе последовательности 2 9 4 3 программа выведет на следующие:

2  
3

② 1 2 2 2

③ ~~maximum := 0;~~ maximum := 0

~~if x > maximum then~~

maximum := 999; (исправить: maximum := 0;)

if x < maximum then (исправить: if (x > maximum)

then)   
 if x mod 2 <> 0 then (исправить: if (x mod 2 <> 0) then) then)

Комментарий: В примере 1 все действия выполнены правильно, но указана третья строка как ошибочная, таковой не являющаяся. Здесь автор демонстрирует незнание приоритета операций. Итог: 2 балла вместо 3-х.

Таким образом, на основе анализа типичных ошибок в решениях задачи С1 участников ЕГЭ по информатике и ИКТ в 2014 году можно выделить предполагаемые причины их появления:

- незнание основных арифметических операций в языке программирования;
- незнание приоритета операций;
- неумение работать с критическими значениями, пограничными в рассматриваемом множестве (наибольшее, наименьшее), в связи с этим выполняется неверная инициализация переменных или использование строгого неравенства вместо нестрогого (и наоборот);
- неумение выполнять программу по шагам.

Для предупреждения этих ошибок необходимо больше внимания уделять отработке навыков трассировки программы, работе с критическими значениями исходных и промежуточных данных, разработке тестов для программ, выполнению тестирования и отладки программ.

Кроме того, при изучении раздела «Программирование» большое внимание следует уделять операциям, с помощью которых можно определять свойства чисел (кратность, положительность, максимальность, минимальность и пр.) – это логические операции, операции целочисленного деления и др. Необходимо подбирать разнообразные задачи для отработки навыков применения и составления сложных логических и арифметических операций, использования приоритета операций.

## 7.2. Задание С2

В задании С1 проверяются умения составлять программу обработки элементов массива. Рекомендуемое время выполнения – 30 минут.

**Пример формулировки задания С2.** Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от 0 до 10000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, произведение которых нечётно, а сумма кратна 3. Под парой подразумевается два подряд идущих элемента массива.

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка.

Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

Сп	Паскаль
<pre>#include &lt;stdio.h&gt; #define N 20 void main() {     long a[N];     long i, j, k;     for (i = 0; i &lt; N; i++)         scanf("%ld", &amp;a[i]);     ... }</pre>	<pre>const N = 20; var a: array [1..N] of longint; i, j, k: longint; begin     for i := 1 to N do         readln(a[i]);     ... end.</pre>

### Пример содержания верного ответа.

На языке Паскаль
<pre>k := 0; for i := 1 to N - 1 do     if (a[i]*a[i+1] mod 2 &lt;&gt; 0) and ((a[i]+a[i+1]) mod 3 = 0) then         inc(k); writeln(k);</pre>
На языке Сп
<pre>k = 0; for (i = 0; i &lt; N - 1; i++)     if (a[i]*a[i+1]%2!=0 &amp;&amp; (a[i]+a[i+1])%3==0)         k++; printf("%ld", k);</pre>

### Основные ошибки при выполнении задания С2

**1. Наиболее часто встречающаяся ошибка – в цикле происходит выход за границу массива.**

Пример 1

```
С2
const
N = 20;
var
a: array [1..N] of longint;
i, j, k: longint;
begin
    for i = 1 to N do
        read (a[i]);
    end;
```

```

for i:=1 to N do
begin
if a[i]+a[i+1] <= 0 then if a[i]*a[i+1] mod 2 <= 0 then
if a[i]+a[i+1] mod 3 = 0 then
j:=j+1;
end;
writeln ('Количество пар равно =', j)
end.

```

Пример 2

```

C2: const N=20;
var
a: array [1..N] of longint;
i, j, k: longint;
begin
for i:=1 to N do
readln (a[i]);
for i:=1 to N do begin
if (a[i]*a[i+1] mod 2 <> 0) and (a[i]+a[i+1] mod 3 = 0)
then k:=k+1;
end;
writeln (k);
end.

```

Пример 3

```

C2 k:=0;
for i:=01 to N do
if (a[i]*a[i+1] mod 2 = 0) and (a[i]*a[i+1] > 0) then
k:=k+1;
writeln(k);

```

Пример 4

C2 Фрагмент программы на языке Паскаль:

```

k:=0;
for i:=1 to N do begin
if (a[i]*a[i+1] mod 2 <> 0) and (a[i]*a[i+1] > 0) then
k:=k+1;
end;
write(k);

```

Комментарий: В примерах 1 и 2 обращение к  $a[i+1]$  элементу неправомерно при  $i=N$ . Здесь есть и другие ошибки: неверно расставленные операторные скобки, скобки приоритета операций. Работы на 0 баллов.

В примере 3 явно допущена ошибка выхода за границы массива. Если считать отсутствие одной скобки опiskой, то работа на 1 балл.

В работе 4 только ошибка выхода за границы массива – 1 балл.



## 2. Распространенная ошибка – не инициализируется или неверно инициализируется счётчик.

Пример 1

```
C2. for i:=1 to N-1 do if (a[i]*a[i+1] mod 2 = 0) AND (a[i]*a[i+1] > 0) then
    inc(k);
    writeln(k);
```

Пример 1

C2 Часть префакши, которую необходимо вставить на место много точек:

```
for i:=1 to n-1 do
    if ((a[i]*a[i+1] mod 2 = 1) and ((a[i]*a[i+1] mod 3 = 0)
        then k:=k+1;
    writeln(k);
```

Комментарий: В примере 1 нет инициализации переменной k. Отметка – 1 балл.

В работе 2 нет инициализации переменной k, кроме этого в сложном логическом выражении пропущена скобка, если считать это за опisku, то работа оценивается в 1 балл из двух. Если подходить формально – 0 баллов.

## 3. Неверно организована проверка данных.

Пример 1

```
C2 Program C2;
const
    N=20;
var
    a: array [1..N] of longint;
    i, j, k: longint;
begin
    k:=0;
    for i:=1 to N do
        for j:=2 to N do
            a[i]*a[j]>0 and (a[i]*a[j] mod 2 > 0) then
                k:=k+1;
    end;
    begin
        if k > 0 then
            writeln(k);
        else
            writeln('не найдено');
        end.
end.
```

Пример 2

```
C2 sum:=0;
    if a[i] mod 2 <> 0 then {находим нечетные}
        begin
            s:=s+a[i] {суммируем нечетные}
        end
    end.
writeln(s); {выводим сумму}
end.
```

Пример 3

```
C2 j:=0;
  for i:=2 to N do
    if (a[i]*a[i-1])>0 and
    if (a[i]*a[i-1]) mod 2 > 0 then
      begin
        j:=j+1;
      end
  writeln(j);
end.
```

Пример 4

```
k:=0;
for i:=1 to N-1 do
  if ((a[i]*a[i+1]) mod 2 <> 0) and ((a[i]*a[i+1]) <> 0) then
    k:=k+1;
writeln(k);
```

*Комментарий:* В примере 1 проверяются условия не для рядом стоящих пар, а для всех возможных пар значений в массиве. То есть, выполняющий усложнил себе задачу. Правда здесь не сформирован сам массив элементов, хотя программа записана полностью, с раздела описания данных. Есть и другие ошибки: неверно расставленные операторные скобки, скобки приоритета операций. Работа на 0 баллов.

В примере 2 представлена работа с множеством ошибок, хотя главная – автор забыл использовать оператор цикла. Если организовывать проверку одновременно с их вводом, то инициализация суммы должна быть до цикла, и тогда, видимо планировалось использовать операторные скобки, закрытие которых мы наблюдаем, а открытия – нет. Кроме того есть еще ошибки: 1) используется переменная Sum, не объявленная в разделе описания переменных; 2) не инициализирована переменная S. С другой стороны, имеет место явная описка, так как далее используется переменная S, а не Sum и тогда остается только неправильное использование операторных скобок и инициализация должна быть до цикла. Итог – 0 баллов.

Работа 3 могла быть выполнена на 2 максимальных балла, если бы при использовании вложенных условных операторов автор вместо and написал then и правильно оформил проверку на нечетность. В данном случае – 0 баллов.

В примере 4 нужно было проверять произведение пары на нечетность и положительность ( $>0$ ), условие же  $a[i]*a[i+1]<>0$  допускает отрицательные значения, то есть является неверным. Оценка – 1 балл.

#### 4. Не учитывается приоритет операций.

Пример 1

```
C2  const
    N = 20;
    var
    a: array [1..N] of longint;
    i, j, k: longint;
    begin
        for i:=1 to N do
            read(a[i]);
        end;
        for i:=1 to N do
            begin
if a[i]+a[i+1] <= 0 then if a[i]*a[i+1] mod 2 <= 0 then
            if a[i]+a[i+1] mod 3 = 0 then
                j:=j+1;
            end;
        end;
        writeln('Команда не правна =', j)
    end.
```

Пример 2

```
C2:  const  N=20;
    var
    a: array [1..N] of longint;
    i, j, k: longint;
    begin
        for i:=1 to N do
            readln(a[i]);
        end;
        for i:=1 to N do begin
            if (a[i]*a[i+1] mod 2 <> 0) and (a[i]+a[i+1] mod 3 = 0)
            then k:=k+1;
        end;
        writeln(k);
    end.
```

Комментарий: В примерах 1 и 2 операция целочисленного деления mod имеет больший приоритет, чем сложение, следовательно, необходимо расставить скобки. Здесь есть и другие ошибки: неверно расставленные операторные скобки, выход за пределы массива, не инициализирован счетчик. Работа на 0 баллов.

#### 5. Ошибки в сложном логическом условии

Пример 1

```
C2.  k:=0
    if A[i] >= -10000 and A[i] <= 10000 then
j:=A[i]*A[i] end (j mod 3=0) and (j>0);
    k:=j+1;
    end;
    writeln(k);
    end.
```

*Комментарий:* В примере 1 неверно составлено сложное логическое выражение, причем используется оно не в качестве условия проверки данных, а присваивается целочисленной переменной. Здесь есть и другие ошибки: неверно расставленные операторные скобки. Работа на 0 баллов.

### 6. Неверно расставлены операторные скобки.

#### Пример 1

```

C2  const
    N = 20;
    var
        a: array [1..N] of longint;
        i, j, k: longint;
    begin
        for i = 1 to N do
            read (a[i]);
        end;
        for i := 1 to N do
            begin
                if a[i] + a[i+1] <= 0 then if a[i] * a[i+1] mod 2 <= 0 then
                if a[i] + a[i+1] mod 3 = 0 then
                    j := j + 1;
                end;
            end;
        writeln ('Количество пар равно =', j)
    end.

```

#### Пример 2

```

C2
const
    N = 20,
var
    a: array [1..N] of longint; i, k, j: longint,
begin
    for i := 1 to N do
        readln (a[i]);
    k := 0;
    for i := 1 to (N-1) do
        j := a[i] * a[i+1];
        if (j mod 2 <> 0) and (j > 0) then
            k := k + 1;
        end.

```

*Комментарий:* В примере 1 используется непарное слово end. Здесь есть и другие ошибки: неверно расставленные скобки в логическом выражении, выход за пределы массива, не инициализирован счетчик. Работа на 0 баллов.

В примере 2 неверно организована проверка условий – нет операторных скобок. Кроме того отсутствует вывод результата. Работа заслуживает 0 баллов.

## 7. Неправильная организация работы цикла.

Пример 1

```

C2. for i:=1 to N-1 do      k:=0;
    begin                  for i:=1 to N-1 do
                                begin
                                if ((a[i]+a[i+1]) mod 3 = 0) and ((a[i]*a[i+1]) mod 3 <> 0)
                                then k:=k+1;
                                end;
                                writeln(k);
    
```

Пример 2

```

C2 const N=20
var
  a: array[1..N] of longint;
  i, k: longint;
begin
  for i:=1 to N do
    readln(a[i]);
    k:=0;
    if ((a[i]*a[i+1]) mod 2 <> 0) and (a[i]+a[i+1]) mod 3 = 0 then
      begin
        k:=k+1;
        write(k);
      end
    else write('not found');
  end.

```

*Комментарий:* В примере 1 индексной переменной неправильно присваивается начальное значение (отсутствует знак равно). Однако это синтаксическая ошибка и работа оценивается в 2 максимальных балла.

В примере 2 вместо цикла записан условный оператор, выполняющий проверку только для одной пары чисел, причем при проверке условия  $i=N$  и обращение к  $a[i+1]$  элементу массива ошибочно. Если считать это неправильной расстановкой операторных скобок, предполагая, что автор пытался выполнить проверку в имеющейся конструкции цикла одновременно с вводом данных, то нужно выводить из цикла инициализацию счетчика, вывод ответа, да и обращение к  $a[i+1]$  элементу массива приводит к выходу за границы массива. Таким образом – 0 баллов.

## 8. Неверно описывается алгоритм решения на естественном языке.

Следует отметить, что при том, что в формулировке задания С2 традиционно значится «запишите на естественном языке или языке программирования», большинство экзаменуемых записывает ответ на известном им языке программирования. Это неудивительно, так как задание формально записать алгоритм на естественном языке является для школьников относительно новым. В современной российской школе уделяется недостаточно внимания (не только на информатике) обучению школьников умению грамотно и четко формулировать целостные логические (алгоритмические) вы-

сказывания на естественном языке, в то время как упражнения на составление простых программ обработки массивов предполагаются большинством школьных программ по информатике.

### Пример 1

С2. Если остаток от деления  $\neq 0$ , тогда это число прибавим к сумме, иначе не прибавляем.  
Выводим сумму всех чисел, остаток от деления которых  $\neq 0$ ; если существует только одно такое число, то выводим его.

### Пример 2

С2. Объявляем массив  $A$  из 20 элементов.  
Объявляем целочисленные переменные  $i, j, k$ .  
В цикле от 1 до 20 вводим элементы массива  $A$  с 1-го по 20-й. Присваиваем « $k$ » и « $j$ » значение 0.  
С помощью цикла присваиваем переменной « $k$ » значения сумм значений  $k$  элементов рядом стоящих друг от друга:  $A[1] + A[2], A[2] + A[3], \dots$ .  
Добавляем условие: если остаток от деления на 2 переменной « $k$ » не равен 0, тогда проверяем через второе условие: если остаток от деления на переменной « $k$ » равен 0, тогда переменной « $j$ » присваиваем значение  $j + 1$ .  
Если условия не выполняются, то переменной « $j$ » присваиваем значение  $j$ . Выводим на экран значение переменной « $j$ ».

*Комментарий:* В примере 1 приведенное описание алгоритмом не является, его невозможно однозначно формализовать. Итог – 0 баллов.

В примере 2 видно понимание автором как организовывать работу программы, но все же, есть ошибки, упущения, не позволяющие четко выделить шаги и формально их описать. Например, непонятно остаток от деления на что проверяем во втором условии. Или, «если условия не выполняются, то  $j$  присваиваем значение  $j$ » в каждом из условных операторов по отдельности или они вложенные? Кроме того, явная ошибка в том, что не все рядом стоящие элементы массива проверяются попарно. Итог – 0 баллов.

## 9. Отсутствует вывод результата.

Пример 1

c2.

```
const
  N = 20;
var a: array [1..N] of longint;
    i, j, k: longint;
begin
  for i := 1 to N do
    readln(a[i]);
  -----
  k := 0;
  for i := 1 to n-1 do begin
j := a[i] j := a[i] * a[i+1];
    if (j > 0) and (j mod 2 <> 0) then
      k := k + 1;
  -----
  end;
End.
```

*Комментарий:* Пример работы, в которой не хочется снимать 1 балл: автор аккуратно переписал начало программы, правильно выполнил задание, выделив свою часть работы, но при этом упустил вывод данных. Не стоило переписывать начало программы, а сконцентрироваться нужно было на получении результата. Итог – 1 балл из двух.

**10. Представление не фрагмента программы, как требуется, а всей программы с начала.**

Это ошибкой не является, но, во-первых тратится лишнее время на экзамене, во-вторых меньше концентрации над выполнением самого задания, в итоге – допускаются ошибки...

Пример 1

```
c2
const
  N = 20;
var
  a: array [1..N] of longint;
  i, j, k: longint;
begin
  for i := 1 to N do
    read(a[i]);
  end;
  for i := 1 to N do
    begin
if a[i] + a[i+1] <= 0 then if a[i] * a[i+1] mod 2 <> 0 then
      if a[i] + a[i+1] mod 3 = 0 then
        j := j + 1;
    end;
  writeln('Количество пар равно =', j)
end.
```

### Пример 2

```
C2: const N=20;
var
  a: array [1..N] of longint;
  i, j, k: longint;
begin
  for i:=1 to N do
    readln (a[i]);
  for i:=1 to N do begin
    if (a[i]*a[i+1] mod 2 <> 0) and (a[i]+a[i+1] mod 3 = 0)
    then k:=k+1;
  end;
  writeln (k);
end.
```

Таким образом, на основе анализа типичных ошибок при выполнении задания С2 участников ЕГЭ по информатике и ИКТ в 2014 году можно выделить предполагаемые причины их появления:

- незнание принципов работы оперативной памяти ЭВМ и возможностей автоматической инициализации переменных («по умолчанию»);
- незнание приоритета операций;
- неумение правильно организовывать работу цикла: устанавливать корректные начальные и конечные значения для индексной переменной; выделять тело цикла;
  - неумение работать с вложенными конструкциями;
  - невнимательность при прочтении условия задачи (упущение возможности получения отрицательных значений; использование необъявленной переменной; не учет того, что нужно написать фрагмент программы, а не всю с нуля);

Для предупреждения этих ошибок необходимо добиться усвоения учащимися основных арифметических и логических операций, алгоритмических конструкций и операторов языка программирования.

Больше времени уделять разработке программ с использованием конструкций цикла и вложенных структур, решению задач на всевозможные переборы и комбинации элементов массива. Обращать внимание учащихся на поведение программы в критических точках, составлять и использовать тест.

Следует развивать умения учащихся грамотно и четко формулировать целостные логические (алгоритмические) высказывания на естественном языке, описывать алгоритмы так, чтобы это описание можно было формализовать.

### 7.3. Задание С3

**Пример формулировки задания С3.** Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по



очереди, первый ход делает Петя. За один ход игрок может добавить в кучу один или три камня или увеличить количество камней в куче в два раза. Например, имея кучу из 15 камней, за один ход можно получить кучу из 16, 18 или 30 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится не менее 51.

Победителем считается игрок, сделавший последний ход, т.е. первым получивший кучу, в которой будет 51 или больше камней.

В начальный момент в куче было  $S$  камней;  $1 \leq S \leq 50$ .

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.

Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

Задание 1

а) Укажите все такие значения числа  $S$ , при которых Петя может выиграть в один ход. Обоснуйте, что найдены все нужные значения  $S$ , и укажите выигрышающие ходы.

б) Укажите такое значение  $S$ , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом. Опишите выигрышную стратегию Вани.

Задание 2

Укажите два таких значения  $S$ , при которых у Пети есть выигрышная стратегия, причём одновременно выполняются два условия:

– Петя не может выиграть за один ход;

– Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня.

Для каждого указанного значения  $S$  опишите выигрышную стратегию Пети.

Задание 3

Укажите значение  $S$ , при котором одновременно выполняются два условия:

– у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети;

– у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Для указанного значения  $S$  опишите выигрышную стратегию Вани.

Постройте дерево всех партий, возможных при этой выигрышной стратегии Вани (в виде рисунка или таблицы). На рисунке на рёбрах дерева указывайте, кто делает ход; в узлах – количество камней в позиции.

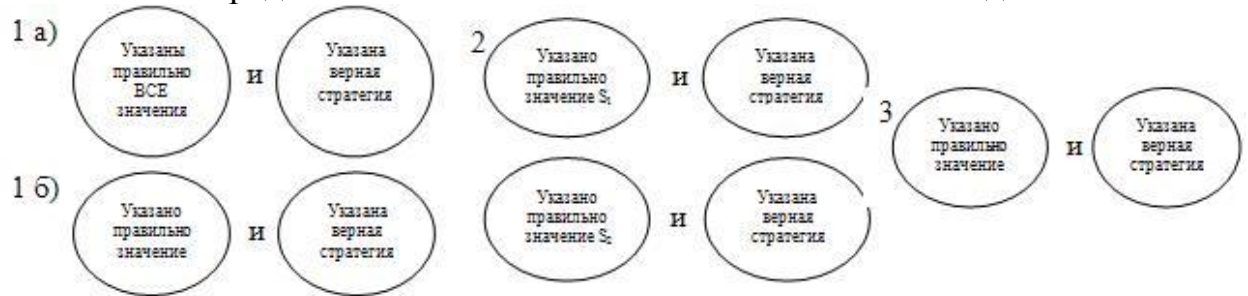
*Комментарий автора:* как видим, в задании С3 необходимо выполнить 3 подзадачи, при этом каждая содержит несколько действий. Задание объемное и по формулировке и по выполняемым действиям, однако это зада-

ние во всех вариантах отличается только числовыми данными, а алгоритм выполнения и приемы решения абсолютно идентичные.

В задании проверяются умения построить дерево игры по заданному алгоритму и обосновать выигрышную стратегию.

В полном ответе на это задание должно быть обязательно указание выигрывающего игрока, его первого выигрышного хода и доказательство существования выигрышной стратегии при любом ответном ходе противника. То есть дерево игры не обязательно должно быть полным, но **выигрышные стратегии** должны быть **отчетливо указаны и описаны до конца**.

Ниже представлена схема полностью выполненного задания С3:



### Пример содержания верного ответа.

#### Задание 1

а) Петя может выиграть, удвоив количество камней в куче, если  $S = 26, \dots, 50$ . При меньших значениях  $S$  за один ход нельзя получить кучу, в которой не менее 51 камня.

б) Ваня может выиграть первым ходом (как бы ни играл Петя), если исходно в куче будет  $S = 25$  камней. Тогда после первого хода Пети в куче будет 26, 28 или 50 камней. Во всех случаях Ваня удваивает количество камней и выигрывает в один ход.

*Замечание для проверяющего:* В случае 50 камней игрок может выиграть и иначе – добавив один или три камня. В задаче не требуется указать все выигрышные стратегии. Если в работе ученика, как в приведённом примере, просто сказано, что игрок всегда удваивает количество камней, – это не ошибка.

#### Задание 2

Возможные значения  $S$ : 22, 24. В этих случаях Петя, очевидно, не может выиграть первым ходом. Однако он может получить кучу из 25 камней. Эта позиция разобрана в п. 1б. В ней игрок, который будет ходить (теперь это Ваня), выиграть не может, а его противник (т.е. Петя) следующим ходом выигрывает.

#### Задание 3

Возможные значения  $S$ : 21, 23.

Например, для  $S = 21$  после первого хода Пети в куче будет 22, 24 или 42 камня. Если в куче станет 42 камня, Ваня удвоит количество камней и выиграет первым ходом. Ситуация, когда в куче 22 или 24 камня, разобрана в п. 2. В этой ситуации игрок, который будет ходить (теперь это Ваня), выигрывает своим вторым ходом.

В таблице изображено дерево возможных партий при описанной стратегии Вани для первого возможного значения. Для второго возможного значения дерево строится аналогично. Заключительные позиции (в них выигрывает Ваня) подчёркнуты. На рисунке это же дерево изображено в графическом виде (оба способа изображения дерева допустимы).

н.п.	Положения после очередных ходов				
	1-й ход Пети (разобраны все ходы)	1-й ход Вани (только ход по стратегии)	2-й ход Пети (разобраны все ходы)	2-й ход Вани (только ход по стратегии)	
21	21 + 1 = 22	22 + 3 = 25	25 + 1 = 26	<u>26 * 2 = 52</u>	
			25 + 3 = 28	<u>28 * 2 = 56</u>	
			25 * 2 = 50	50 + 1 = 51	<u>50 + 3 = 53</u>
				50 + 3 = 53	<u>50 * 2 = 100</u>
				50 * 2 = 100	
			21 + 3 = 24	24 + 1 = 25	25 + 1 = 26
	25 + 3 = 28	<u>28 * 2 = 56</u>			
	25 * 2 = 50	50 + 1 = 51			<u>50 + 3 = 53</u>
		50 * 2 = 100			
	21 * 2 = 42	<u>42 * 2 = 84</u>			

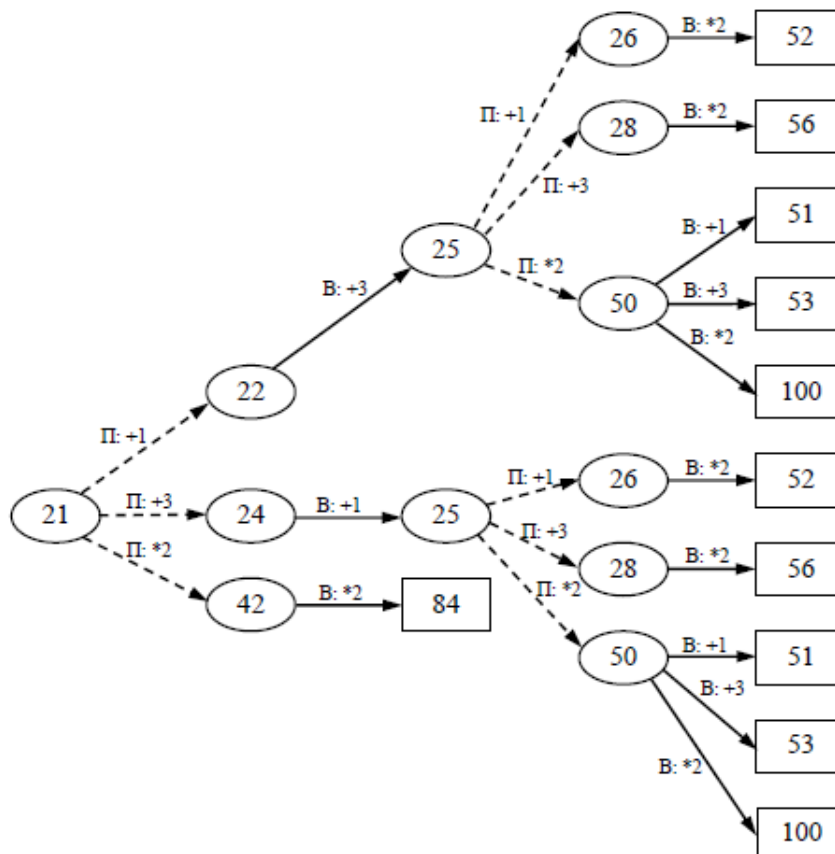
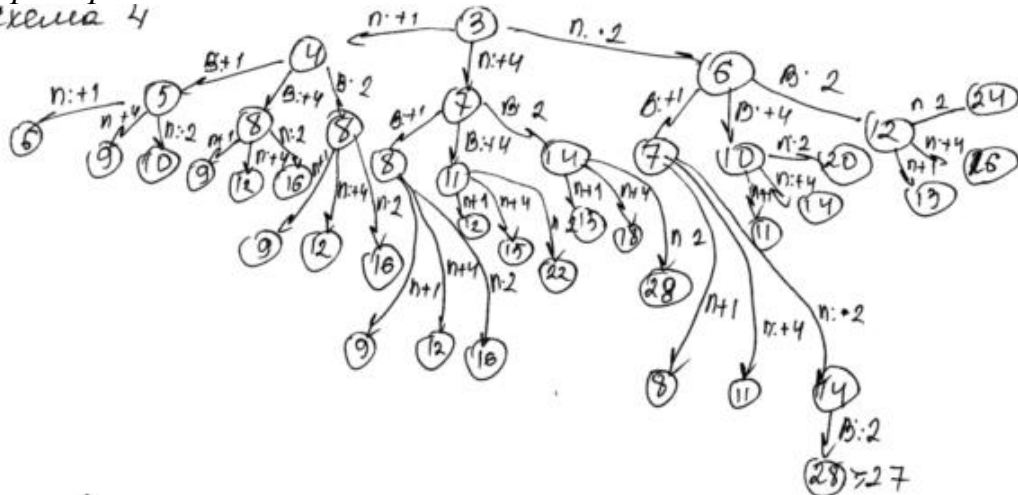


Рис. 7. Дерево всех партий, возможных при Ваниной стратегии. Прямоугольником обозначены позиции, в которых партия заканчивается

## Основные ошибки при выполнении задания С3

1. Не обоснована выигрышная стратегия одного игрока при ЛЮБОМ ходе партнера по игре.

Пример 1  
схема 4



Пример 2

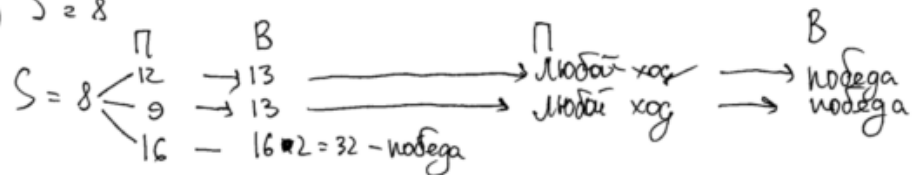
а) Это числа  $S \in [14; 26]$ . Петья поведет за 1 ход умножив их на 2. Если  $S \leq 13$  то за 1 ход он никак не сможет победить.

б)  $S = 13$ . Петья не может выбрать, делая любой ход он увеличивает количество камней как минимум в 2 раза. Вана остается только умножить  $S$  на 2 и победит.

2)  $S = 12$  и  $S = 9$ . при  $S = 12$  Петья добавляет 1, Вана делает любой ход и Петья побеждает на 2 ходе.

При  $S = 9$  Петья добавляет 4, Вана делает любой ход и Петья побеждает.

3)  $S = 8$



Чтобы Вана победил в 1 ходе Петья должен умножить  $S$  на 2, а чтобы во втором ходе Вана должен сделать любой другой ход.

Пример 3  
С3.

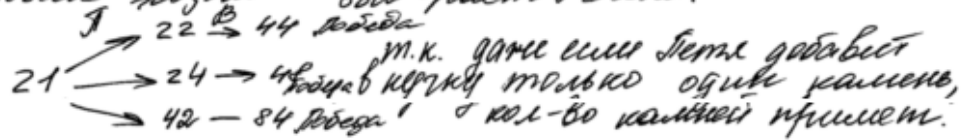
1. а) При  $S \in [22 \dots 43]$  Петья выигрывает своим первым ходом т.к. если  $22 \cdot 2 = 44$  то получается,  $S \geq 44$ , значит первым ходом Петья выигрываетเสมอ.

Все числа из этого промежутка при умножении на 2 будут давать более или равно 44 камням.

$$\begin{aligned}
 22 \cdot 2 &= 44 \geq 44 \\
 23 \cdot 2 &= 46 \geq 44 \\
 24 \cdot 2 &= 48 \geq 44 \\
 43 \cdot 2 &= 86 \geq 44
 \end{aligned}$$

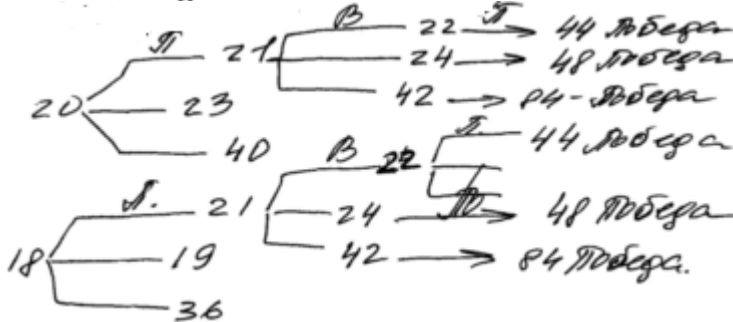
Если же  $21 \cdot 2 = 42$  это меньше и не равно 44, Петя не сможет выиграть первым ходом как и с другими числами  $< 22$

б) Если количество камней в куче будет 21, то при любом ходе Вани Петя выиграет своим первым ходом выиграв 44 камня.

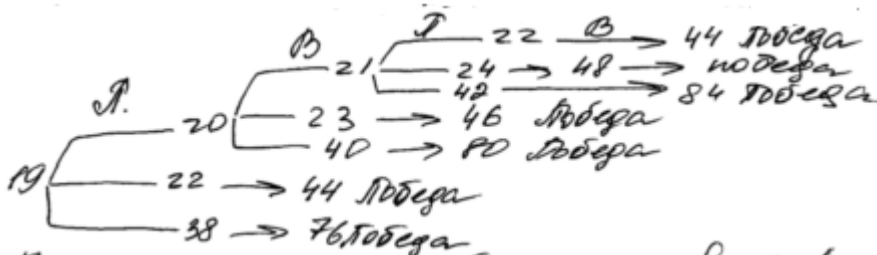


значение 22, при таком кол-ве камней Вани сможет выиграть игру (см. 1а).

2. Это значение 20 и 18.



3. Это значение 19.



Комментарий: В примере 1 автор работы не обосновывает выигрышную стратегию одного из игроков, а рассматривает все возможные варианты развития событий, не выделяя выигрышных ходов.

В примере 2 при выполнении третьей части задания автор не указывает выигрышной стратегии Вани при его втором ходе, «любой ход» там не подходит. Итог – 2 балла.

В работе 3 автор не указывает ни в части 2 (для Пети), ни в части 3 (для Вани), что игрок для выигрыша должен сделать ход в позицию 21 и не делать других ходов. В третьей части указаны выигрышные ходы и для Пети и для Вани. Обычно эксперты в таких случаях склонны оценивать в пользу ученика (3 балла), но если подходить формально – это только 2 балла.

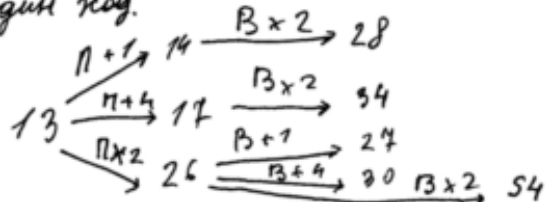
**2. Указаны не все выигрышные позиции для игрока.**

Пример 1

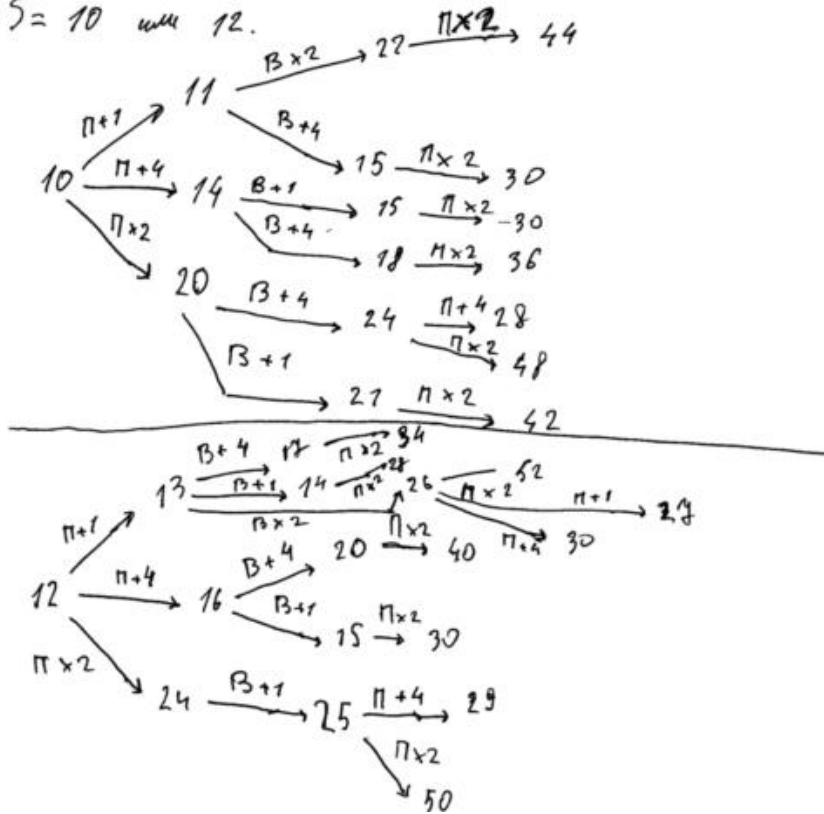
С3. Если в куче будет 26 камней, то Петя может выиграть в один ход

$$26 + 1 = 27; \quad 26 + 4 = 30; \quad 26 \times 2 = 52;$$

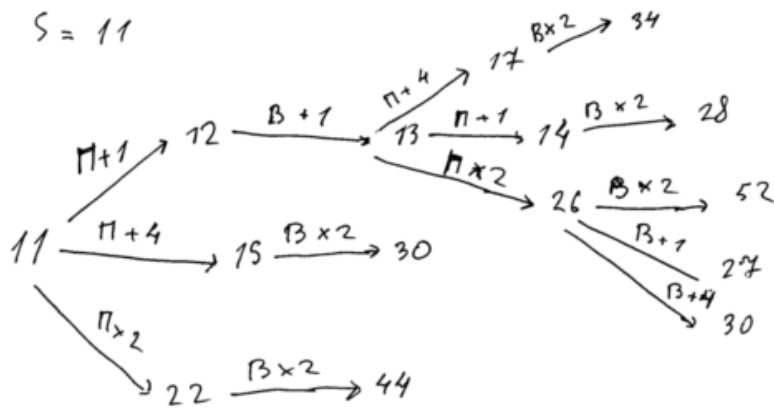
1б. Если в куче будет 13 камней то Ваня сможет победить в один ход.



2)  $S = 10$  или  $12$ .



3)  $S = 11$



Пример 2

СЗ. 1р: 50, 48, 26. Значения кол-ва камней в куче при условии выигрыша первого игрока  $\leq 3$ , так как всего 3 разновидности камней. Выигрышные карты:  $50 + 1 = 51$  ( $\geq 51$ ),  $48 + 3 = 51$  ( $\geq 51$ ),  $26 \times 2 = 52$  ( $\geq 51$ ). Сделка 500 руб.

д)  $S = 25$  Рассмотрим таблицу:

старт	1 ход	2 ход
25	26 28	52 56 100

1 ход - все варианты ходов Тети.  
2 ход - выигрышная стратег. Вани.

В таблице следует, что при любом ходе Тети, Ваня выигрывает (выигр. стратег.)

2)  $S = 24, 22$ . Рассмотрим исход игры ~~с~~ с учетом выигрышной стратегии Тети:

старт	1 ход	2 ход	3 ход
24, 22	25, 25	50, 50 26, 26 28, 28	100, 100 52, 52 56, 56

~~1 ход - все варианты~~  
1 ход - выигр. стратег. Тети  
2 ход - все варианты Вани  
3 ход аналог 1 ходу.

Выигр. стратег. Тети:

1. При  $S = 24$ : первое действие +1
2. При  $S = 22$ : первое действие +3

В обоих случаях вывести количество к 25 (ч. 15)

3.  $S = 12$

старт	1 ход	2 ход	3 ход	4 ход
	24	25	27/50/26	$\times 2$ - выигр
12	13	16/14	32/17/18 28/15/16	$\times 2$ - выигр
	15	18/16	20/19/20	

1 ход - все варианты  
2 ход - выигр  
3 ход - выигр

Пример 3

СЗ) 1) а) При  $S$  от 14 до 26 Тети можно выиграть за один ход.

если  $S$  от 14 до 26 - увеличит количество в 2 раза ~~...~~

если  $S = 26$  - прибавить 1 или 4 камня. } возможные хода.

если  $S$  от ~~23~~ до 26 - прибавить 4 камня

~~если  $S = 23$  прибавить 4~~

б) При  $S = 13$  учитывая на 2 попутки наибольшее - 26. Это недостаточное для победы.

Если хода Тети будет  $13 + 1 = 14$  увеличив в 2 раза  
 $13 + 4 = 17$  количество камней  
 $13 \cdot 2 = 26$  Ваня выигрывает 1 ходом.

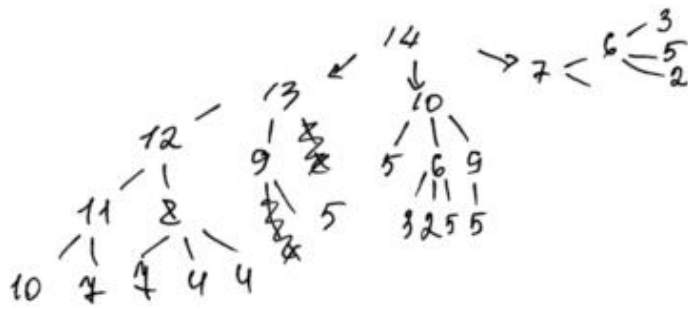
2)  $S = 13$  если Ваня прибавит 1 то Тети, увеличив в 2 раза количество до 28 и победит. (1 в худшем случае, но независимо от хода Вани Ваня выигрывает).

3)  $S = 12$

Прибавив 1 ходом 1 в куче будет 13 камней из этой позиции выиграть невозможно за 1 ход (см. 1б). Если любого хода Вани Тети выигрывает своим вторым ходом.

$S = 10$  прибавив 1 ходом 4 в куче будет 13 камней. Аналогично после любого хода Вани Тети выигрывает своим вторым ходом.

3) Необходимо получить число 14



Комментарий: В примере 1 из выигрышных позиций от 14 до 26, указано только число 26, хотя в части 1б) автор рассматривает ходы первого игрока приводящие именно к 14 или более, для выигрыша первым ходом второго игрока. Во второй части ответа из двух выигрышных позиций правильно указана только одна. Полностью выполнена в данном примере только третья часть задания. Итог – 2 балла.

В примере 2 в части 1а) указан не весь диапазон значений от 26 до 50, а только три из них. В третьей части рассмотрена неверная позиция. правильно выполнена только вторая часть задания. Итог – 1 балл.

В примере 3 во второй части неверно указана вторая позиция и в третьей части неверно указана выигрышная позиция. Полностью выполнена только первая часть задания С3. Итог – 1 балл.

### 3. Ошибки в вычислениях, неверная интерпретация результатов.

Пример 1

С3.

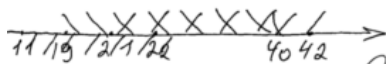
$$1) a) S \xrightarrow{\pi_1} \geq 44$$

$$\begin{cases} S+1 \geq 44 \\ S+3 \geq 44 \\ S \cdot 2 \geq 44 \end{cases} \quad \begin{cases} S \geq 43 \\ S \geq 41 \\ S \geq 22 \end{cases}$$

Ответ: 22.

$$b) S \xrightarrow{\pi_1} [22 \dots 43] \xrightarrow{\pi_2} \geq 44$$

$$\begin{cases} S+1 \in [22 \dots 43] \\ S+3 \in [22 \dots 43] \\ S \cdot 2 \in [22 \dots 43] \end{cases} \quad \begin{cases} S \in [21 \dots 42] \\ S \in [19 \dots 40] \\ S \in [11 \dots 22] \end{cases}$$



Ответ: 19, 21.

$$2. S \xrightarrow{\pi_1} [19, 21] \xrightarrow{\pi_2} [22 \dots 43] \xrightarrow{\pi_2} \geq 44$$

$$\begin{cases} S+1 \in [19, 21] \\ S+3 \in [19, 21] \\ S \cdot 2 \in [19, 21] \end{cases} \quad \begin{cases} S \in [18, 20] \\ S \in [16, 18] \\ S \in \emptyset \end{cases}$$

Ответ: 18.



$$\begin{cases} S \geq \frac{22 \dots 43}{n_1} \geq \frac{244}{n_1} \\ S \geq \frac{18}{n_1} \geq \frac{19, 21}{n_1} \geq \frac{22 \dots 43}{n_2} \geq \frac{244}{n_2} \end{cases}$$

$$\begin{cases} S+1 \in 22 \dots 43 \\ S+1 \in 18 \end{cases} \quad \begin{cases} S \in 21 \dots 42 \\ S \in 17 \end{cases}$$

$$S \in 17, 21 \dots 42.$$

$$\begin{cases} S+3 \in 22 \dots 43 \\ S+3 \in 18 \end{cases} \quad \begin{cases} S \in 19 \dots 40 \\ S \in 15 \end{cases}$$

$$S \in 15, 19 \dots 40.$$

$$\begin{cases} S \cdot 2 \in 22 \dots 43 \\ S \cdot 2 \in 18 \end{cases} \quad \begin{cases} S \in 11 \dots 22 \\ S \in 9 \end{cases}$$

$$S \in 9, 11 \dots 22.$$

Ответ: 21.

Ч.П	наименьше ходов			
	1 ход Ферр.	1 ход Валли	2 ход Ферр.	2 ход Валли
21	$21+1=22$	$22+1=23$	$23+1=24$ $27+3=30$	$24 \cdot 2=48$ $30 \cdot 2=60$
	$21+3=24$	$24+3=27$	$84 \cdot 2=168$	$168 \cdot 2=336$
	$21 \cdot 2=42$	$42 \cdot 2=84$		

**Комментарий:** Выбран прием решения, позволяющий математически точно обосновать ответ, но допущены ошибки в вычислениях. В части 1а) неверный вывод – ответ должен быть объединением множеств, а не пересечением, то есть  $S \geq 22$ , а не 22. В части 1б) при  $2 \cdot S \in [22 \dots 43]$ ,  $S$  принадлежит отрезку  $[11 \dots 21]$ , а не  $[11 \dots 22]$ . Здесь ошибка вычисления, которая дает лишний ответ. Но автор ошибся и в решении системы: 19 не принадлежит пересечению множеств!

В части 2 снова неверно интерпретируется результат – решением должно быть объединение значений.

В части 3 ошибки, как в построении системы неравенств, так и в её решении.

Итак, первая часть задания С3 выполнена неверно, во второй части правильно названо и обосновано только одно из двух решений. Часть 3 выполнена неверно. В соответствии с критериями эксперт оценит работу в 1 балл.

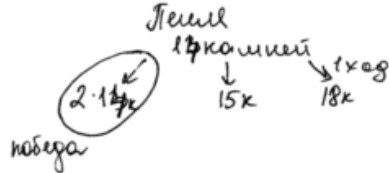
**4. Вместо описания выигрышной стратегии одного игрока при любом ходе другого, рассматривается частный случай выигрыша за указанное число ходов.**

Пример 1

$C_3$

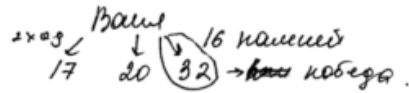
Задача 1:

а) Пети может выиграть за 1 ход тогда, когда количество камней в его куче будет превышать 13.

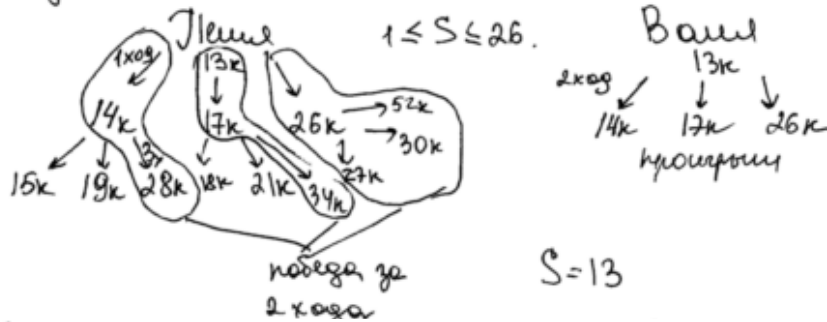


б) Пети не сможет выиграть за один ход тогда, когда количество камней в его куче будет меньше или равно 13.

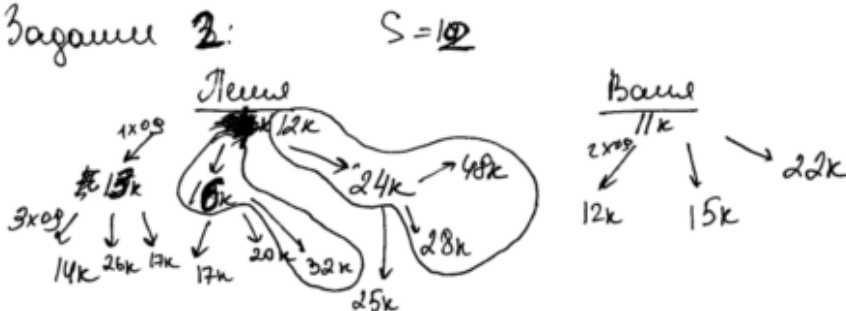
Тогда у Вани в куче будет 16 камней, и с вероятностью  $\frac{1}{3}$  он может выиграть.



Задача 2:



Задача 3:



Комментарий: В примере 1 при выполнении всех частей задания автор работы указывает не выигрышные позиции, а некоторое количество камней и рассматривает частный случай выигрыша нужного игрока за нужное количество ходов в указанной позиции. При этом нет даже попытки обоснования выигрышной стратегии. Задание выполнено на 0 баллов.

5. Ошибка в интерпретации задания части 3.

Часто экзаменующиеся в задании «Укажите значение  $S$ , при котором одновременно выполняются два условия: – у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети; – у Вани нет стратегии, которая позволит ему гарантированно выиг-

рать первым ходом» не видят требования одновременности выполнения условий и не понимают, что первый ход негарантированно дает выигрыш, а второй – обязательно выигрышный, в случае невыигрышного первого хода.

Иногда рассматривают две разных позиции для каждого условия, иногда рассматривают частные случаи и др.

Пример 1

С3.

Задание 1:

а)  $S = \{22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40,$   
 $41, 42, 43\}$ . Число 43 при прибавлении 1 дает результат

44. Числа 41, 42, 43 при прибавлении 3 дают результат 44 и более.

<sup>Все</sup> ~~Составные~~ числа при умножении на 2 дают результат более 44 (кроме числа 22, которое при умножении на 2 даёт ровно 44).

б)  $S = 21$ . Тётя может сделать 1 из 3 возможных ходов: прибавить 1, прибавить 3, умножить на 2. В любом случае Ваня умножив полученное число на 2 получит результат более 44 или ровно 44.

Задание 2:

$S = 18$ . Тётя прибавив 3 получает 21. После ~~х~~ любого хода Вани, получается одно из 3 чисел (22, 24, 42), при которых Тётя может выиграть в один ход.

$S = 20$ . Тётя прибавив 1 получает 21. Дальнейшие события такие же как с числом 18.

Задание 3:

Противоречие в условиях:

- Есть стратегия выиграть 1 ходом
- Нет стратегии выиграть 1 ходом.

Комментарий: В примере 1 работа могла быть оценена в 3 балла, если бы ученик был внимателен при прочтении текста и не принял бы два условия за противоречащие. Итог – 2 балла.

### 6. Не все пункты задания выполнены.

Собственно ошибкой это не является, просто задание считается не выполненным полностью и не может быть оценено на максимальные 3 балла. Однако выполнение отдельных его пунктов может заслуживать 1 или 2 баллов.

Итак, на основе анализа типичных ошибок при выполнении задания С3 участников ЕГЭ по информатике и ИКТ в 2014 году можно выделить предполагаемые причины их появления:

- непонимание сути разработки выигрышной стратегии для игрока 1: при любом ходе игрока 2 из заданной позиции, указать только ходы, приводящие к выигрышу игрока 1;
- неумение строго обосновывать возможность выигрыша;
- неумение составлять и решать системы и совокупности неравенств, незнание, что является решением системы, а что – совокупности;
- невнимательность при прочтении условия задачи;
- неумение логически рассуждать.

Для предупреждения этих ошибок важно при изучении углубленного курса информатики и ИКТ уделять внимание заданиям на разработку стратегии игры. Такие задания позволяют сформировать алгоритмическое мышление, в целом развивать качества стратега и тактика.

Следует развивать умения учащихся грамотно и четко формулировать логические (алгоритмические) высказывания на естественном языке и делать выводы.

Традиционно сложилось, что особое внимание построению доказательной базы отводится на уроках математики, тогда как при изучении информатики и ИКТ это так же является немаловажной частью умений, а точнее – одной из основных, при выборе моделей, построении правильных алгоритмов вычислений и правильно работающих программ.

#### **7.4. Задание С4**

Задание С4 направлено на проверку умений выполнять все этапы решения задачи с помощью ЭВМ: вычленение исходных данных и результата (постановка задачи); формализация, моделирование, составление теста, разбиение задачи на подзадачи, разработка алгоритма, написание программы. Это самое трудоемкое задание в ЕГЭ по информатике и ИКТ. На его выполнение рекомендуется отводить около 55 минут.

Сложность задания состоит не только в необходимости самостоятельного выполнения всего цикла разработки программы, но и в наложении условий эффективности алгоритма, как по времени выполнения, так и по использованию памяти. Задачи имеют прикладное значение, и, часто подходят для олимпиад, хотя, стоит отметить, что формулировки заданий С4 в 2014 году более просты для восприятия по сравнению с 2013 годом.

##### **Пример формулировки задания С4.**

Сейсмограф автоматической геофизической станции «Токанава» передает показания каждую минуту. Показания сейсмографа – положительные вещественные числа, не превышающие 1000.

Необходимо найти в заданной серии показаний прибора минимальную сумму двух показаний, между моментами передачи которых прошло не менее 4 минут. Общее количество показаний прибора в серии не превышает 10 000. Временем передачи сигнала можно пренебречь.

Напишите на любом языке программирования программу для решения поставленной задачи. Ваша оценка будет зависеть не только от правиль-

ности программы, но и от того, насколько она эффективна.

Программа считается эффективной по времени, если время работы программы пропорционально количеству полученных показаний прибора  $N$ , т.е. при увеличении  $N$  в  $k$  раз время работы программы должно увеличиваться не более чем в  $k$  раз.

Программа считается эффективной по памяти, если размер памяти, использованной в программе для хранения данных, не зависит от числа  $N$  и не превышает 1 килобайта.

Максимальная оценка за правильную программу, эффективную по времени и по памяти, – 4 балла.

Максимальная оценка за правильную программу, эффективную по времени, но неэффективную по памяти, – 3 балла.

Максимальная оценка за правильную программу, неэффективную ни по времени, ни по памяти, – 2 балла.

Перед программой укажите версию языка и кратко опишите использованный алгоритм.

В первой строке задаётся число  $N$  – общее количество показаний прибора. Гарантируется, что  $N > 4$ . В каждой из следующих  $N$  строк задаётся одно положительное вещественное число – очередное показание прибора.

Пример входных данных:

9  
12  
45  
5  
4  
21  
20  
10  
12  
26

Программа должна вывести одно число – описанную в условии сумму.

Пример выходных данных для приведённого выше примера входных данных:

15

### **Пример программы, эффективной по времени и по памяти.**

Для построения программы, эффективной по времени, можно определить для каждого элемента входных данных минимальное значение от начала данных до этого элемента включительно. Затем нужно прибавлять к каждому элементу, начиная с пятого, значение этого минимума, взятого на четыре элемента раньше, и выбрать наименьшую из этих сумм.

```
program c4;  
const s = 4;           {требуемое расстояние}  
var N: integer;  
    a: array[0..s - 1] of real; {хранение показаний прибора}  
                                {k-е число записываем в a[k mod 4]}
```

```

a_: real;           {ввод очередного показания}
mn: real;          {минимальное введенное число}
                  {не считая 4 последних}
m: real;           {значение минимальной суммы}
i: integer;
begin
readln(N);
{ Пролог. Ввод первых четырех чисел}
for i := 1 to s do
begin
readln(a_);
a[i mod s] := a_
end;
{ Ввод остальных значений, поиск минимальной суммы}
mn := 1001; m := 2001;
for i := s + 1 to N do
begin
readln(a_);
if a[i mod s] < mn then mn := a[i mod s];
if a_ + mn < m then m := a_ + mn;
a[i mod s] := a_
end;
writeln(m)
end.

```

#### **Основные ошибки при выполнении задания С4**

Можно было бы предположить, что ошибки, указанные при анализе заданий частей А, В по теме «программирование» и заданий С1 и С2 будут распространены и в задании С4. Но это предположение неверное. Синтаксические ошибки, ошибки в применении операторов, ошибки при использовании арифметических и логических операций при выполнении данного задания допускаются крайне редко. Это объясняется тем, что брались за выполнение данного задания только «сильные» учащиеся, умеющие программировать.

Основные типы ошибок, допущенных при выполнении задания С4, это:

- неэффективное использование памяти и времени работы программы (хотя это хороший прием без риска заработать 2 балла из 4-х);
- алгоритмические (содержательные) ошибки;
- неправильное выделение этапов работы программы (подзадач), пропуск некоторых этапов;
- отдельные синтаксические ошибки.

# 1. Написание неэффективной программы.

## Пример 1

```
С4                                     Паскаль
var
a: array [1..10000] of integer;
minimal, i, j: integer;
begin
  readln(n);
  minimal := 2000;

  for i := 1 to n do
    readln(a[i]);
    for j := i+4 to n do begin
      if ((a[i] + a[j]) < minimal) then
        minimal := a[i] + a[j];
    end;
  end;
  writeln(minimal);
end.
```

## Пример 2

```
С4. PascalABC.net
var
  a: array [1..N] of real integer;
  sum, i, j: integer;
  N: longint;
begin
  sum := 10000;
  readln(N);
  for i := 1 to N do readln(a[i]);
  for j := 1 to N do
    for i := 4 to N do
      if ((a[j] + a[i]) < sum) then sum := a[j] + a[i];
  writeln(sum);
end.
```

## Пример 3

```
С4) var                                     { PascalABC.net }
  i, N, max, s1, s2: integer;
  a: array [1..N] of integer;

begin
  readln(N);
  for i := 1 to N do readln(a[i]);
  max := -10;
  for i := 1 to N-5 do
    begin
      k := 5;
      while (i+k) ≤ N do
        {1) считываем число показаний }
        {2) считываем показания }
        {3) делаем переход всех показаний,
            начинаем с первого и заканчиваем 5-ым
            с конца }
        {4) считываем сумму квадр-
```

```

begin
  s1 := sqr(a[i]);
  s2 := sqr(a[i+k]);
  if (s1+s2) > max then max := s1+s2;
  k := k+1;
end;
end;
writeln(max);
end.

```

паров пар. одной показываю  
связь с тем, между  
объектами передать ко-  
торых можно не менее  
5 штук, и находим  
максимальное значение  $\{$

$\{5\}$  выводим максимальную  
сумму квадратов пар  $\}$

*Комментарий:* В примере 1 программа работает правильно, но неэффективна по памяти, так как в оперативной памяти создается массив для хранения всех элементов и объем занятой памяти зависит от  $N$ . Программа неэффективна и по времени, так как используются вложенные циклы и производится полный перебор всех пар (временная емкость порядка  $N^2$ ).

Кроме этого в программе неправильно описаны типы данных (по условию показания – вещественные числа) и значение индексной переменной внешнего цикла должно быть  $N-4$ , иначе бессмысленно 4 раза проверяется условие выхода из цикла. Однако, эти ошибки следует приравнивать к синтаксическим, они в целом не влияют на алгоритм решения.

В соответствии с критериями такое решение оценивается в 2 балла, в этом случае допускается неэффективность, наличие до 7 синтаксических ошибок и до двух «содержательных».

В примере 2 аналогичный случай, ошибка:  $i:=4$ , вместо  $i:=j+4$ . Оценка – 2 балла.

В примере 3 работающая программа, неэффективная по времени и памяти. Замечание одно – не вещественный тип данных! Итог – 2 балла.

## 2. Алгоритмические (содержательные) ошибки.

### Пример 1

```

Free pascal 1.6.4
var a1, a2, a3, a4, z;
    min: real;
begin
  cin readln (a1; a2; a3; a4; z);
  min := z + a1;
  while not eof do
    begin
      readln z;
      a1 := a2;
      a2 := a3;
      a3 := a4;
      a4 := z;
      readln(z);
      if (min > a1 + z) then min := a1 + z;
    end;
  writeln (min);
end.

```



## Пример 2

С4 PascalABC.net. - ~~задача~~ проф.з

Краткий алгоритм:  
Присваиваем ~~минимум~~ значение минимуму  
максимальной из возможных, считываем кол-во  
показаний.

Вводим 4 переименованные, которые будут хранить  
показания за последние 4 минуты.

Затем проверяем, является ли сумма показаний  
полученных с разницей в 4 секунды наименьшей  
или максимумом, если является, то присваиваем  
значение минимуму значение максимум  
суммы.

Выводим значение минимальной суммы  
на экран

```
var t, k, n, min, i2, i1, i3, i4: longint;  
i: integer;
```

```
begin
```

```
  min := 2000;
```

```
  readln(n);
```

```
  readln(i1);
```

```
  readln(i2);
```

```
  readln(i3);
```

```
  readln(i4);
```

```
  for j := 5 to n do
```

```
    begin
```

```
      readln(t)
```

```
      k := i mod 4;
```

```
      if k = 0 then
```

```
        begin
```

```
          if (i4 + t) < min then min := i4 + t;
```

```
          i4 := t
```

```
        end
```

```
      else
```

```
        if k = 1 then
```

```
          begin
```

```
            if (i1 + t) < min then min := i1 + t;
```

```
            i1 := t;
```

```
          end
```

```
        else
```

```
          if k = 2 then
```

```
            begin
```

```
              if (i2 + t) < min then min := i2 + t;
```

```
              i2 := t;
```

```
            end
```

```
          else if k = 3 then
```

```
            begin
```

```
              if (i3 + t) < min then min := i3 + t;
```

```
              i3 := t;
```

```
            end;
```

```
          end;
```

```
  writeln(min);
```

```
end.
```

### Пример 3

C4. Die Free Pascal 2.4.2.

При рассмотрении  $i$ -го элемента ( $x_i$ ) при  $i > 5$  будем думать в памяти  $x_{i-5}, \dots, x_{i-1}$ . Таким образом для  $i > 5$  можно легко вычислить максимальное значение  $x_i, \dots, x_{i+5}$  при последовательном рассмотрении. Для каждого  $i > 5$  найдем сумму  $x_i$  и макс квадрата ближайших соседа  $x_{i-5}, \dots, x_{i+5}$ . Найдем и выведем максимальную сумму.

```
VAR
  i, n: integer;
  x, maxa, maxs: real;
  a: array [1..5] of real;
begin
  readln (n); maxs:=0; maxa:=0;
  for i:=1 to n do begin
    readln(x); if i=1 then maxa
    if i > 5 then begin
      if maxa < a[5] then maxa:=a[5];
      if maxa*maxa+x*x > maxs then maxs:=maxa*maxa+x*x;
    end;
    a[5]:=a[4]; a[4]:=a[3]; a[3]:=a[2]; a[2]:=a[1]; a[1]:=x;
  end;
  writeln (maxs);
end.
```

Комментарий: В примере 1 приведено интересное решение, эффективное по времени и памяти, но задача не решена правильно. Здесь находится минимальная сумма двух элементов с шагом равным 4, а не больше либо равным 4! Но один балл решение заслуживает вполне.

В примере 2 одна принципиальная ошибка при разработке алгоритма – находится минимальная сумма элементов отстоящих на кратное 4-м число, а не на  $\geq 4$ ! Итог только 1 балл.

В примере 3 приведена работа на 3 балла. Здесь неверная обработка начальных элементов данных (не обработаны первые пять элементов).

### 3. Ошибки разбиения задачи на подзадачи.

#### Пример 1

C4. Pas Паскаль

```
const N = 10000
var i, x, min: real;
    a: array [1..N] of real
begin
  min := 2000;
  for i:=1 to N do begin
    readln(a[i]);
    if i > 5 then
      x := a[i] + a[i-5];
    if x < min then
      min := x;
  end;
  writeln (min)
end.
```

### Пример 2

```
Ch. Turbo Pascal 7.0.
program a1;
const
  N=10000;
var a,b,c,t1,t2 : integer;
d : array [1..N] of integer;
begin
  for i:=1 to N do
  begin
    read a[i]; a:=1000; b:=1000
    if a[i]<a then begin
      a:=a[i];
      t1:=i
    end;
    if (a[i]<b) then and (a[i]>a) then begin
      b:=a[i];
      t2:=i
    end;
    if abs(t2-t1) <= 4 then
      c:=a abs(a-b)
    writeln(c)
  end.
```

### Пример 3

```
Ch Pascal
uses crt;
var mas : array [1..N] of integer;
N, i, min, j, Smin, Smin 1, Smin 2 : integer;
begin
  clrscr;
readln(N); readln(N); {уману ве отпону}
  for i:=1 to N do {уману элемент массива}
    readln(mas[i]);
  for i=1 to N do
    Smin := 2000;
  for i=1 to N do
  begin
    if ((a[i] + a[j]) < Smin) and (j >= i+4) then Smin 1 := a[i] + a[j];
    if ((a[i] + a[j]) < Smin) and (j <= i+4) then Smin 2 := a[i] + a[j];
  end;
  if Smin 1 < Smin 2 then Smin := Smin 1. {исца мин сумма}
  else Smin := Smin 2;
  writeln(Smin);
  readkey;
end
```

Комментарий: В примере 1 представлен неправильный алгоритм: расстояние между данными рассматривается не «>=4» как требует условие, а «=5», к тому же в массиве каждый пятый элемент это сумма всех предыду-

щих на расстоянии 5. Таким образом, искомая сумма пары данных не ищется. Итог – 0 баллов.

В примере 2 реализована попытка разработки эффективной по времени, но не по памяти программы. Реализован поиск двух наименьших показаний и зафиксировано время их регистрации. Только в конце не обязательно окажется, что разница во времени регистрации показаний будет больше или равна 4 (правда автор ошибочно проверяет разницу неравную четырем). К тому же не закрыта одна из операторных скобок. Если её закрыть перед проверкой разности  $t_1$  и  $t_2$ , то к алгоритму останется лишь одно замечание, указанное выше. В общем, работу можно оценить в 1 балл, есть понимание в целом о порядке решения задачи.

В примере 3 тоже можно увидеть понимание того, что должно выполняться в программе, но неверно расставлены операторные скобки и организована работа циклов, переменная `Smin` инициализирована не в нужном месте, не та индексная переменная использована во втором цикле. Если исправить эти ошибки, то алгоритм бы работал неэффективно, но правильно. Такое выполнение задания допустимо оценить в 1 балл.

Перед текстом программы можно кратко описать используемый алгоритм решения задачи. Хорошее описание может быть основанием для выставления 1 балла в случае плохой реализации. В критериях оценивания на 1 балл указано: «Программа, возможно, неверно работает при некоторых входных данных, но по приведенному тексту решения ясно, что экзаменуемый понимает, из каких этапов должно состоять решение задачи». Как правило, такое описание отсутствует, а если и есть, то является плохим примером изложения алгоритма на естественном языке, с нечетко выделенными этапами, невозможным для формализации.

#### **4. Синтаксические ошибки.**

Как выше было отмечено, синтаксических ошибок при выполнении задания С4 встречалось немного. К тому же, технические программистские ошибки (например, отведение недостаточного количества памяти, необнуление значения переменной или выход за границу массива, некорректное использование стандартных функций соответствующего языка программирования) в данной работе следует приравнивать к синтаксическим, если они не порождены ошибками в алгоритме решения.

Чаще всего встречались следующие ошибки: описание неверного типа данных; пропуск разделителей «;»; выход за границу массива при обращении к элементу  $a[i+5]$ . Примеры этого встречаются в приведенных выше работах.

Итак, на основе анализа типичных ошибок при выполнении задания С4 участников ЕГЭ по информатике и ИКТ в 2014 году можно выделить предполагаемые причины их появления:

- неумение строить наиболее эффективные алгоритмы решения;
- неуверенное владение приемами формализации данных и моделирования;
- невнимательность при прочтении условия задачи;
- выполнение разработки алгоритма сразу на языке программиро-

вания без продумывания и четкого формулирования на естественном языке каждого этапа решения задачи.

Для предупреждения этих ошибок важно при изучении углубленного курса информатики и ИКТ в разделе «Программирование» уделять внимание отработке каждого этапа разработки программы для ЭВМ, разработке тестов и применению их при отладке программы.

Учащиеся должны быть приучены к тому, что в критериях оценивания программы предусмотрено прохождение теста на время выполнения и объем используемой памяти.

## **8. Влияние на результаты экзамена в форме ЕГЭ используемого учебника**

Школьный учитель имеет возможность выбора той или иной программы и соответствующего ей учебного комплекса, рекомендованных или утверждённых Министерством образования Российской Федерации к введению в учебный процесс. Перечень таких учебников ежегодно утверждается и публикуется на информационных сайтах ([www.mon.gov.ru](http://www.mon.gov.ru) – сайт Министерства образования; [www.edu.ru](http://www.edu.ru) – портал Российского образования).

На сегодняшний день нет ни одного учебника по информатике, по которому можно подготовиться к ЕГЭ. Для качественной подготовки стоит использовать комбинацию допущенных и рекомендованных учебников в сочетании с теми, в которых та или иная тема изложена методически более привлекательно. Можно говорить о необходимости компилировать содержание разных пособий для успешной подготовки к ЕГЭ.

Профильный характер экзамена не позволяет подготовиться к нему при наличии лишь базового курса информатики, предполагающего 1 час занятий в неделю. В ряде школ учителя выбирают форму дополнительной, послеурочной подготовки выпускников, выбравших данный предмет. Другим вариантом подготовки является выбор профиля информационной направленности с последующим набором ряда элективных курсов, позволяющих подготовиться к ЕГЭ «в сетке часов».

Можно использовать для самоподготовки учащихся цикл видеороликов центра онлайн-обучения [100EGE.ru](http://100EGE.ru) на YouTube, разработанный преподавателями ФИПИ.

Статистически влияние выбора основного учебника на результаты ЕГЭ в этом году не отслеживалось РЦОИ ЕГЭ в Алтайском крае.

## 9. Сведения о работе конфликтной комиссии

Количество поданных и удовлетворенных апелляций по результатам ЕГЭ по информатике и ИКТ в 2014 году:

Количество поданных апелляций всего.....	10
из них: по процедуре.....	0
по результатам.....	10
Удовлетворено апелляций всего.....	6
из них: с понижением балла.....	0
с повышением балла.....	5
без изменения балла.....	1
Отклонено апелляций.....	4

В одной из удовлетворенных апелляций изменения коснулись только части В (техническая ошибка при верификации).

Основными причинами удовлетворения апелляций по части С: несоответствие выставленного за задание балла его критериям (возможно, что экспертами была проверена структура решения без его подробностей); изменение в подходах оценивания задачи С1; нечеткость границ в критериях к задаче С4; недостаточное исследование экспертами особенностей сложных (неэффективных) решений заданий С2 и С4; недостаточная компетентность экспертов во владении некоторыми языками программирования.

Работа конфликтной комиссии проходила в спокойной и доброжелательной обстановке, была хорошо организована. В рассмотрении работ, поданных на апелляцию, участвовали опытные эксперты, председатель предметной комиссии и его заместитель.

Поданные апелляции рассматривались в присутствии выпускников или их доверенных лиц.

## **10. Общие выводы и рекомендации**

**(уровень подготовки участников экзамена по предмету в целом; умения, которые показали выпускники; недостатки в подготовке участников экзамена; рекомендации)**

**Общий уровень** подготовки участников ЕГЭ по информатике и ИКТ в Алтайском крае можно признать удовлетворительным с учетом специфики преподавания этого предмета в общеобразовательных учреждениях нашего края. Наблюдается незначительное понижение результатов экзамена по информатике и ИКТ по сравнению с прошлым годом.

Значительная часть выпускников показала освоение на базовом уровне умений использовать приобретенные знания из курса информатики и ИКТ в практической деятельности.

Средний балл по региону составил 58,96 и оказался несколько ниже, чем в прошлом году.

Количество работ с результатами 91-100 баллов по сравнению с 2013 годом понизилось в процентном отношении более чем в 4 раза. По сравнению с прошлым 2013 годом уменьшилось более чем в 1,5 раз количество выпускников, чьи работы получили максимальную оценку в 100 баллов.

Процент неудовлетворительных результатов сдачи экзамена выше, чем в прошлом году в 1,7 раз.

В целом показатели по Алтайскому краю остаются удовлетворительными и стабильными, но следует обратить внимание на рост числа выпускников, не преодолевших порогового значения тестового балла. Основная причина этого видится в отсутствии отлаженной системы подготовки к ЕГЭ по информатике и ИКТ в виду ряда факторов: необязательность выбора ЕГЭ по информатике и ИКТ в качестве итогового экзамена; отсутствие преподавания предмета на предыдущих этапах обучения и условий для самостоятельного освоения информационных и компьютерных технологий; использование ЕГЭ по информатике и ИКТ в качестве вступительного испытания только в одном вузе края.

Факт того, что значительная часть преподавательского состава (особенно в районах Алтайского) не имеет профессиональной подготовки по предмету и анализ результатов единого государственного экзамена позволяют сделать вывод о недостаточности подготовки учащихся для сдачи единого государственного экзамена при изучении предмета на базовом уровне для поступления на профильное обучение. При изучении предмета на базовом уровне стоит рекомендовать учащимся посещение занятий в центрах дополнительного образования и на курсах подготовки к ЕГЭ в течение двухгодичного курса (10–11 класс).

### **Умения, которые показали выпускники:**

- оценивать числовые параметры информационных объектов и процессов в двоичной системе счисления;
- описывать реальный объект и процесс в виде таблицы и графа и работать с построенной моделью;



- строить модели объектов, систем и процессов в виде таблицы истинности для логического высказывания;
- работать с файловой системой организации данных;
- использовать базы данных для поиска и отбора информации;
- представлять и анализировать табличную информацию в виде графиков и диаграмм;
- строить алгоритмы в заданной системе команд исполнителя и выполнять практические вычисления;
- читать и выполнять по шагам программы на языке программирования, содержащие операторы присваивания и ветвления.

**Недостатки в подготовке участников экзамена:**

- слабое владение даже на базовом уровне абитуриентами темой «Измерение и кодирование информации, информационные процессы»;
- низкий уровень владения темой «Системы счисления» при выполнении заданий повышенной сложности;
- низкие знания и неуверенное владение методами решения показали учащиеся по теме «основы логики»: законами логики, приемами упрощения логического выражения, решения систем логических уравнений, способам логических доказательств, комбинаторики;
- низкий уровень владения принципами устройства архитектуры компьютерных сетей и технологиями поиска информации;
- неумение работать с рядами данных, итерационными процессами, рекуррентными формулами, рекурсией;
- незнание основных арифметических операций в языке программирования;
- незнание приоритета операций;
- неумение работать с критическими значениями, пограничными в рассматриваемом множестве (наибольшее, наименьшее), в связи с этим выполняется неверная инициализация переменных или использование строгого неравенства вместо нестрогого (и наоборот);
- неумение выполнять по шагам программу, содержащую циклы, функции;
- неумение правильно организовывать работу цикла: устанавливать корректные начальные и конечные значения для индексной переменной; выделять тело цикла;
- неумение работать с вложенными конструкциями;
- непонимание сути разработки выигрышной стратегии для игрока 1: при любом ходе игрока 2 из заданной позиции, указать только ходы, приводящие к выигрышу игрока 1;
- неумение строго обосновывать возможность выигрыша;
- неумение составлять и решать системы и совокупности неравенств;
- неумение строить наиболее эффективные алгоритмы решения;
- неуверенное владение приемами формализации данных и моделирования при разработке алгоритмов решения задач.

## Методические рекомендации для учителей информатики и ИКТ

В случае выбора информатики для поступления в вуз следует при подготовке к экзамену тщательно выбирать стратегию подготовки в соответствии с будущей направленностью профессиональной деятельности, рекомендовать занятия в центрах дополнительного образования, участие в олимпиадах и конкурсах, при самостоятельной подготовке учащихся предложить список учебных пособий и интернет-ресурсов.

При подготовке учащихся к ЕГЭ следует сосредоточить усилия прежде всего на темах, включенных в программы для поступающих в вузы: алгоритмизация, программирование и изучение базовых принципов организации и функционирования ПК, как наиболее сложных для изучения и требующих продолжительного времени на отработку умений и навыков. Задания, связанные с проверкой знаний по этим темам, традиционно имеют самый низкий процент выполнения. 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 компьютерной программы и ее последующей отладки. 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. Провести разбор и пропедевтику указанных в аналитическом отчете типичных ошибок при выполнении заданий ЕГЭ.

С учащимися, имеющими слабую подготовку по информатике и ИКТ, стоит сконцентрироваться на формировании их базовых компетенций, определить наиболее успешно решаемые данными учащимися типы задач и доводить, в первую очередь, их решение «до совершенства». Другими словами, для учащихся с разным уровнем подготовки должны быть выстроены принципиально разные стратегии подготовки к экзамену, необходима дифференциация обучения, разработка стратегии обучения и подготовки к выпускному экзамену с учетом уже имеющегося у выпускника уровня образовательной подготовки.

Подготовиться к выполнению заданий части С наскоком нельзя. Нужна планомерная работа по развитию соответствующих качеств ума, творческих и аналитических способностей в течение всего обучения и их систематизации при подготовке к ЕГЭ. Учителю информатики и ИКТ рекомендуется проявлять **готовность к самообучению**.

1. Прежде всего, учителю необходимо познакомиться со структурой и содержанием КИМов, сравнить их с содержанием программного материала и того учебника, по которому учатся школьники. Следует отметить, что в настоящее время учащиеся в разных школах и у разных учителей изучают информатику и ИКТ по разным учебникам, которые сопровождаются своими дидактическими и контрольными материалами. Стили представления заданий, само содержание заданий у разных авторов разнятся между собой. С другой стороны, у разработчиков КИМов могут быть свои стилевые предпо-

чтения в формулировках, характере заданий и пр. В связи с этим учитель должен понять, что ученик изучает предмет, а не учебник. А потому при планировании изучения каждой ключевой темы школьного курса информатики и ИКТ учитель должен ориентироваться в действующих учебниках и достаточно обширной методической литературы и самому проектировать учебные модули, учитывая все достоинства и недостатки имеющегося учебника. Кроме того, использовать наряду с традиционной формой контроля знаний – тестовую, причем использовать в своей работе различные тесты разных авторов, чтобы ученики могли понимать разные стили и типы заданий, а не привыкли бы к какому-то одному стилю (типу).

2. Учителю необходимо продумать специальную работу по освоению учащимися техники тестирования. Тренаж именно по тестированию имеет большое значение, ведь эта форма отличается от привычных школьнику письменных и устных экзаменов. Центром тестирования РФ издаются сборники тестов для учащихся с 5 по 11 класс. Кроме этого существуют различные ИНТЕРНЕТ-ресурсы (<http://reshuege.ru/> и др.); открытый банк заданий по информатике и ИКТ на сайте <http://fipi.ru> и др. С их помощью можно оценить уровень усвоения материала, отработать технику выполнения теста, что способствует возможности повысить результат. Зная типовые конструкции тестовых заданий, ученик практически не будет тратить время на понимание инструкций. Во время таких тренировок формируются соответствующие психотехнические навыки саморегуляции и самоконтроля. Это может быть в форме домашнего задания, самопроверки, взаимопроверки и др. Ученые считают, что психотехнические навыки сдачи экзаменов не только повышают эффективность подготовки к экзаменам, но и позволяют более успешно вести себя во время экзамена, мобилизовать себя в решающей ситуации. Опыт работы с определенными типами задач позволяет сразу выбирать и применять наиболее эффективные методы решения (при условии владения ими), а не тратить время на поиски таких методов решения.

3. Учителю необходимо реализовать сбалансированное сочетание традиционных и новых методов контроля знаний. При подготовке старшеклассников к ЕГЭ, необходимо учесть наличие в КИМах части С, требующей развернутого решения задания высокой сложности, с необходимыми пояснениями всех логических ходов. Поэтому для подготовки к экзамену следует уделять внимание грамотному решению и оформлению решения сложных заданий исследовательского характера части С.

4. Учителю необходимо планировать обобщающее повторение курса информатики и ИКТ, традиционно проводимое в конце 11 класса, с учетом основных содержательных линий курса. Ориентиром в планировании могут послужить:

- кодификатор требований к уровню подготовки выпускников общеобразовательных учреждений для проведения единого государственного экзамена по информатике и ИКТ;
- спецификация контрольных измерительных материалов для прове-

дения в текущем году единого государственного экзамена по информатике и ИКТ;

– демонстрационный вариант контрольных измерительных материалов единого государственного экзамена текущего года по информатике и ИКТ.

При новой форме диагностики качества образования учителю необходимо непрерывно повышать свой профессиональный академический уровень. Если раньше (до ЕГЭ) учитель считал, что подготовка выпускников к поступлению в вуз не является его задачей и задачей школы и учитель не несет ответственности за поступление или не поступление в вуз, то сейчас каждый учитель (как основной так и старшей школы) заинтересован в получении высоких результатов ЕГЭ, так как по ним могут судить о его профессионально - академическом уровне. Здесь очень важно понимать, что учителя основной школы не менее ответственны за подготовку учащихся к новой форме аттестации, как в области процедурной организации деятельности школьников, так и в подготовке учащихся по предмету. Учитель обязан дать возможность каждому ученику освоить высокий уровень знаний.

Наличие в интернете открытого банка заданий части КИМ ЕГЭ по информатике и ИКТ (<http://fipi.ru>) дает возможность постепенно включать эти задания в учебный процесс, а на завершающем этапе подготовки к экзамену эффективно проводить диагностику недостатков и их устранение в усвоении отдельных тем путем решения серии задач конкретного типа. Доступ к заданиям открытого банка свободный и для школьника, и для учителя, и для родителя.

#### **Рекомендации по численности и составу предметной комиссии**

Количественный состав экспертов в 2014-2015 учебном году оставить без изменений, однако продолжить практику качественной ротации кадров: активнее привлекать к проверке развёрнутых ответов ЕГЭ учителей, работающих в старших классах; учителей высшей категории.

Продолжить в практике подготовки к экзамену проведение обучающих семинаров для экспертов, расширить практику работы экспертов с прототипом дистанционного обучения. Целью такой деятельности является выработка единых подходов к оцениванию работ, позволяющая, во-первых, повысить качество проверяемых работ, во-вторых снизить процент третьих проверок.

Обеспечить участие председателя предметной комиссии и его заместителя в работе семинаров по согласованию подходов к оцениванию заданий ЕГЭ с развернутой формой ответов, проводимых ФГБНУ «ФИПИ».

#### **Рекомендации по обеспечению нормативной и методической информацией по организации деятельности предметной комиссии**

Необходимо осуществлять ознакомление учителей и преподавателей информатики образовательных учреждений (в том числе, учреждений начального и среднего профессионального образования) с нормативными документами и методическими материалами ЕГЭ по информатике и ИКТ. Такую деятельность можно осуществлять в следующих формах:

– обсуждение демонстрационных вариантов, кодификаторов и спецификаций ЕГЭ по информатике и ИКТ нового учебного года на методических объединениях учителей в образовательных учреждениях, муниципалитетах, в районных методических объединениях с привлечением учителей информатики и ИКТ, работающих в 10-11 классах в текущем учебном году;

– проведение курсов повышения квалификации, организуемые *с непосредственным участием* представителей предметной комиссии по информатике и ИКТ;

– выездные обучающие семинары в образовательных округах Алтайского края (где массово выбирается ЕГЭ по информатике и ИКТ) для учителей, работающих в 11 классах в текущем учебном году, включающие в программу вопросы нормативного и методического обеспечения ЕГЭ *с непосредственным участием* представителей краевой предметной комиссии по информатике и ИКТ;

– своевременной размещение нормативной и методической документации ЕГЭ текущего года на сайтах РЦОИ ЕГЭ в Алтайском крае, Главного управления образования и молодёжной политики Алтайского края;

– подготовка и распространение в образовательных учреждениях Алтайского края для обсуждения на методических комиссиях по предмету «Анализа результатов Единого государственного экзамена в текущем году», сформированного председателями предметных комиссий с участием РЦОИ.

Методическую помощь учителю и учащимся при подготовке к ЕГЭ могут оказать материалы с сайта ФИПИ ([www.fipi.ru](http://www.fipi.ru)):

– документы, определяющие структуру и содержание КИМ (кодификаторы элементов содержания и требований, спецификация и демонстрационный вариант КИМ);

– открытый сегмент Федерального банка тестовых заданий;

– учебно-методические материалы для председателей и членов региональных предметных комиссий по проверке выполнения заданий с развёрнутым ответом экзаменационных работ ЕГЭ;

– аналитические отчеты о результатах экзамена и методические письма прошлых лет;

– перечень учебных изданий, разработанных специалистами ФИПИ или рекомендуемых ФИПИ для подготовки к ЕГЭ.

## Литература

1. Евич, Л.Н., Лисица С.Ю. Информатика и ИКТ. Подготовка к ЕГЭ-2013 / Под ред. Ф.Ф. Лысенко, Л.Н. Евич. – Ростов-на-Дону: Легион, 2012. – 432 с.
2. ЕГЭ-2013 : Информатика : самое полное издание типовых вариантов заданий / авт.-сост. Д.М. Ушаков, А.П. Якушкин. – Москва: АСТ-Астрель, 2013. – 316 с.
3. Златопольский, Д.М. ЕГЭ по информатике. Решение задач по программированию / Д.М. Златопольский. – СПб.: БХВ-Петербург, 2013. – 304 с.
4. Лещинер, В.Р. ЕГЭ-2014. Информатика. Типовые тестовые задания / В.Р. Лещинер. – Москва: Изд. «Экзамен», 2014. – 167 с.
5. Лещинер, В.Р. Оптимальный банк заданий для подготовки учащихся. Единый государственный экзамен 2014. Информатика. Учебное пособие / В.Р. Лещинер, С.С. Крылов, А.П. Якушкин. – Москва: Интеллект-Центр, 2014. – 176 с.
6. Мациевский, С.В. Информатика как решение задач ЕГЭ: учебное пособие / С.В. Мациевский. – Калининград: Изд-во РГУ им. И. Канта, 2009. – 419 с.
7. Островская, Е.М., Самылкина, Н.Н. Подготовка к ЕГЭ. Высший уровень качества ЕГЭ 2012. Информатика. Сдаем без проблем / Е.М. Островская, Н.Н. Самылкина.– Москва: ЭКСМО, 2011.
8. Якушкин, А.П., Лещинер, В.Р., Кириенко, Д.П. Информатика ЕГЭ 2011. Типовые тестовые задания. 10 вариантов заданий. Разбор решений. Ответы. – Москва: Изд. «Экзамен», 2011.

## Информационные ресурсы

- В помощь учителю. Федерация интернет-образования <http://som.fio.ru/>;
- Интернет-ресурсы по обучающим программам Дистанционное обучение – проект «Открытый колледж» <http://www.college.ru/indexGraph.php3>;
- Образовательный портал «РЕШУ ЕГЭ» <http://inf.reshuege.ru/>
- Ресурс <http://ege.yandex.ru/> (Раздел <http://ege.yandex.ru/informatics/>);
- Ресурс <http://www.ctege.info/>;
- Российский образовательный портал. Каталог справочно-информационных источников [http://www.school.edu.ru/catalog.asp?cat\\_ob\\_no=1165](http://www.school.edu.ru/catalog.asp?cat_ob_no=1165);
- Российское образование. Федеральный портал <http://www.edu.ru/>;
- Сайт К.Ю. Полякова <http://kpolyakov.narod.ru/> (Разделы <http://kpolyakov.narod.ru/school/ege.htm> и <http://kpolyakov.narod.ru/school/kumir.htm>);
- Специализированный ресурс по информатике и математике

<http://ege-go.ru>;

• Учитель.ру – Федерация интернет-образования  
<http://teacher.fio.ru>;

• Федеральная служба по надзору в сфере образования и науки.  
Федеральный институт педагогических измерений <http://www.fipi.ru> .