Министерство образования Иркутской области
Государственное автономное учреждение
дополнительного профессионального образования Иркутской области
«Институт развития образования Иркутской области»

Результаты государственной итоговой аттестации в форме единого государственного экзамена по информатике и ИКТ в Иркутской области в 2019 году

Методические рекомендации

ГАУ ДПО ИРО, РЦОИ

Иркутск, 2019

УДК 371.29
ББК 74.202.83

Рецензент: Зинченко Анна Сергеевна, кандидат физико-математических наук, доцент кафедры алгебраических и информационных систем ИМЭИ ФГБОУ "Иркутский Государственный университет"

Семичева Н. Л., Лебедева С. Ю.
Результаты государственной итоговой аттестации в форме единого государственного экзамена по информатике и ИКТ в Иркутской области в 2019 году. Методические рекомендации / Семичева Н. Л., канд. физ.-мат. наук, доцент; Лебедева С. Ю. Иркутск: ГАУ ДПО ИРО, 2019. 50 с.

ГАУ ДПО ИРО, РЦОИ

В методических рекомендациях представлены статистические данные о результатах ЕГЭ в Иркутской области. Проведен методический анализ результатов ЕГЭ по учебному предмету и анализ типичных затруднений выпускников региона при выполнении заданий ЕГЭ. Даны рекомендации по повышению качества образования по предмету.

Методические рекомендации предназначены для работников системы образования: специалистов органов управления образованием, специалистов организаций дополнительного профессионального образования, руководителей образовательных организаций и организаций среднего профессионального образования, учителей-предметников. Могут быть интересны обучающимся, их родителям, представителям широкой общественности.

Статистические данные представлены региональным центром обработки информации (комплекс программ РИС ГИА-11).

УДК 371.29
ББК 74.202.83

© Н. Л. Семичева
© С. Ю. Лебедева
© ГАУ ДПО ИРО, 2019

СОДЕРЖАНИЕ

I. ХАРАКТЕРИСТИКА УЧАСТНИКОВ ЕГЭ ПО УЧЕБНОМУ ПРЕДМЕТУ 4

1.1. Количество участников ЕГЭ по учебному предмету (за последние 3 года) 4

1.2. Процентное соотношение юношей и девушек, участвующих в ЕГЭ 4

1.3. Количество участников ЕГЭ в регионе по категориям 4

1.4. Количество участников ЕГЭ по типам ОО 4

1.5. Количество участников ЕГЭ по предмету по АТЕ 5

II. ОСНОВНЫЕ РЕЗУЛЬТАТЫ ЕГЭ ПО ПРЕДМЕТУ 7

2.1. Диаграмма распределения тестовых баллов по предмету в 2019 г. 7

2.2. Динамика результатов ЕГЭ по предмету за последние 3 года 7

2.3. Результаты по группам участников экзамена с различным уровнем подготовки 7

2.4. Перечень ОО, продемонстрировавших высокие результаты ЕГЭ по предмету 9

2.5. Перечень ОО, продемонстрировавших низкие результаты ЕГЭ по предмету 10

2.6. Вывод о характере изменения результатов ЕГЭ по предмету 11

III. АНАЛИЗ РЕЗУЛЬТАТОВ ВЫПОЛНЕНИЯ ОТДЕЛЬНЫХ ЗАДАНИЙ ИЛИ ГРУПП ЗАДАНИЙ 13

3.1. Краткая характеристика КИМ по учебному предмету 13

3.2. Анализ КИМ 14

3.3. Характеристики выявленных сложных для участников ЕГЭ заданий с указанием типичных ошибок и выводов о вероятных причинах затруднений при выполнении указанных заданий 19

3.4. Выводы по разделу 48

IV. РЕКОМЕНДАЦИИ ПО ПОДГОТОВКЕ К ЕГЭ 50

ГАУ ДПО ИРО, РЦОИ

I. ХАРАКТЕРИСТИКА УЧАСТНИКОВ ЕГЭ ПО УЧЕБНОМУ ПРЕДМЕТУ

1.1. Количество участников ЕГЭ по учебному предмету (за последние 3 года)

Таблица 1

2017 г.		2018 г.		2019 г.	
чел.	% от общего числа участников	чел.	% от общего числа участников	чел.	% от общего числа участников
1 314	9,8	1 681	11,9	1 862	12,8

1.2. Процентное соотношение юношей и девушек, участвующих в ЕГЭ

Таблица 2

Пол	**2017 г.**		**2018 г.**		**2019 г.**	
	чел.	% от общего числа участников	чел.	% от общего числа участников	чел.	% от общего числа участников
Девушки	407	31	446	26,5	517	27,8
Юноши	907	69	1 235	73,5	1 345	72,2

1.3. Количество участников ЕГЭ в регионе по категориям

Таблица 3

	Количество	**Доля от общего числа участников**
Всего участников ЕГЭ по предмету	1 862	100
Из них: выпускников текущего года, обучающихся по программам СОО	1 762	94,6
выпускников текущего года, обучающихся по программам СПО	19	1
выпускников прошлых лет	80	4,3
участников с ограниченными возможностями здоровья	16	0,9

1.4. Количество участников ЕГЭ по типам ОО

Таблица 4

	Количество	**Процент**
Всего участников экзамена (без учета ВПЛ)	1 782	100
Из них: – выпускники лицеев и гимназий	635	35,6
– выпускники СОШ	1 096	61,5
– выпускники вечерних СОШ	16	0,9
– выпускники СПО	16	0,9
– другие	19	1,1

ГАУ ДПО ИРО РЦОИ

1.5. Количество участников ЕГЭ по предмету по АТЕ

Таблица 5

№	АТЕ	Количество участников ЕГЭ по учебному предмету	% от общего числа участников в регионе
1	Ангарский городской округ	282	1,9
2	Зиминское городское МО	6	0,04
3	Зиминское районное МО	1	0,01
4	г. Иркутск	571	3,9
5	Иркутское районное МО	30	0,2
6	МО Аларский район	9	0,1
7	МО Балаганский район	6	0,04
8	МО Баяндаевский район	4	0,03
9	МО Боханский район	9	0,06
10	МО Братский район	26	0,2
11	МО город Саянск	33	0,2
12	МО город Свирск	6	0,04
13	МО город Тулун	16	0,1
14	МО город Усолье-Сибирское	45	0,3
15	МО город Усть-Илимск	73	0,5
16	МО город Черемхово	19	0,1
17	МО города Бодайбо и района	22	0,2
18	МО города Братска	251	1,7
19	МО Жигаловский район	4	0,03
20	МО Заларинский район	7	0,05
21	МО Иркутской области Казачинско-Ленский район	11	0,08
22	МО Катангский район	2	0,01
23	МО Качугский район	3	0,02
24	МО Киренский район	6	0,04
25	МО Куйтунский район	8	0,05
26	МО Мамско-Чуйский район	3	0,02
27	МО Нижнеилимский район	34	0,2
28	МО Нижнеудинский район	52	0,4
29	МО Нукутский район	2	0,01
30	МО Осинский муниципальный район	11	0,1
31	МО Слюдянский район	40	0,3
32	МО Тайшетский район	40	0,3
33	МО Тулунский район	2	0,01
34	МО Усть-Илимский район	6	0,04
35	МО Эхирит-Булагатский район	13	0,1
36	Ольхонское районное МО	6	0,04
37	Усольское районное МО	18	0,1
38	Усть-Кутское МО	39	0,3
39	Черемховское районное МО	10	0,1
40	Чунское районное МО	20	0,1
41	МО Шелеховский район	69	0,5

ГАУ ДПО ИРО РЦОИ

Ежегодно увеличивается показатель участников ЕГЭ, сдающих информатику и ИКТ, причем не только в количественном выражении, но и в

процентном. Если посмотреть динамику увеличения процента сдающих информатику и ИКТ по отношению к общему числу участников ЕГЭ с 2017 года (2017 г. – 9,8, 2018 г. – 11,9, 2019 г. – 12,8), можно сделать вывод о постоянном возрастающем интересе к этому предмету. Заметим также, что юношей, сдающих информатику и ИКТ стабильно больше, чем девушек, причем в 2018 и 2019 годах более чем в два с половиной раза (2017 г. – в 2,2 раза, в 2018 г. – в 2,8, в 2019 г. – в 2,6), причем средний бал у юношей и девушек не сильно отличается и последние три года составляет от 52 до 59 баллов.

В основном, экзамен сдают выпускники СОО текущего года (94 % – 95 %), участников ЕГЭ по информатике и ИКТ, окончивших СПО – единицы (2017 г. – 14 человек, 2018–2019 гг. – 19 человек). В 2019 году экзамен также сдавал один выпускник иностранной ОО.

По-прежнему, больше трети участников экзамена являются выпускниками лицеев и гимназий, хотя их значительно меньше, чем СОШ, и они в основном сосредоточены в больших населенных пунктах.

Также можно заметить, что около трети участников экзамена, являются выпускниками школ, лицеев и гимназий областного центра. Большое количество участников приходится на такие большие АТЕ, как Ангарское МО и МО г. Братска (соответственно, 15 % и 13 %). К тому же количество участников экзамена в этих АТЕ постоянно увеличивается. Если в 2017 году в Иркутске Информатику и ИКТ сдавали только 393 человека, в 2018 году – уже 545, а в 2019 году – 571. В Ангарском МО и МО г. Братска схожая динамика: 2017 год – 193 и 156 человек, соответственно, 2018 год – 217 и 214, 2019 год – 282 и 251. Еще можно отметить стабильное увеличение количества участников экзамена в МО г. Усть-Илимска, МО Слюдянского района, МО Черемховского района, МО Балаганского района. Замечено резкое увеличение количества участников экзамена по Информатике и ИКТ в 2019 году в МО г. Усолье-Сибирское (с 28 до 45 человек), в МО г. Бодайбо и района (с 6 до 22). Постоянной отрицательной динамики по количеству участников экзамена по информатике и ИКТ нет ни в одном АТЕ Иркутской области, но есть уменьшение количества участников более чем на 30 % по сравнению с 2018 годом в следующих АТЕ: МО Боханский район – 36 %, МО Заларинский район – 44 %, Зиминское городское МО – 77 % (АТЕ, где стабильно информатику и ИКТ сдает меньше 10 человек, не рассматриваем).

ГАУ АПО ИРО, РЦОИ

II. ОСНОВНЫЕ РЕЗУЛЬТАТЫ ЕГЭ ПО ПРЕДМЕТУ

2.1. Диаграмма распределения тестовых баллов по предмету в 2019 г.

2.2. Динамика результатов ЕГЭ по предмету за последние 3 года

Таблица 6

	Иркутская область					
	2017 г.		2018 г.		2019 г.	
	Кол-во	%	Кол-во	%	Кол-во	%
Не преодолели минимального балла	201	15,3	322	19,2	257	13,8
Средний тестовый балл	54,7		52,4		57	
Получили от 81 до 99 баллов	135	10,3	110	6,5	246	13,2
Получили 100 баллов	6	0,5	4	0,2	6	0,3

2.3. Результаты по группам участников экзамена с различным уровнем подготовки

А) с учетом категории участников ЕГЭ

Таблица 7

	Выпускники текущего года, обучающиеся по программам СОО	**Выпускники текущего года, обучающиеся по программам СПО**	**Выпускники прошлых лет**	**Участники ЕГЭ с ОВЗ**
Доля участников, набравших балл ниже минимального	13,1	52,6	20	18,8
Доля участников, получивших тестовый балл от минимального балла до 60 баллов	41	26,3	46,3	56,3

ГАУ ДПО ИРО РД

	Выпускники текущего года, обучающиеся по программам СОО	Выпускники текущего года, обучающиеся по программам СПО	Выпускники прошлых лет	Участники ЕГЭ с ОВЗ
Доля участников, получивших от 61 до 80 баллов	32,1	21,1	22,5	18,8
Доля участников, получивших от 81 до 99 баллов	13,5	0	11,3	6,3
Количество участников, получивших 100 баллов	6	0	0	0

Б) с учетом типа ОО

Таблица 8

	Доля участников, получивших тестовый балл				Количество участников, получивших 100 баллов
	ниже минимального	от минимального до 60 баллов	от 61 до 80 баллов	от 81 до 99 баллов	
СОШ	18,4	49,1	26,1	6,4	0
Лицеи, гимназии, СОШ с УИП	2,8	26,9	43,5	25,8	6
Другие дневные ОО	11,1	55,6	16,7	16,7	0
Вечерние СОШ	62,5	25	12,5	0	0
СПО	56,3	31,3	12,5	0	0

В) основные результаты ЕГЭ по предмету в сравнении по АТЕ

Таблица 9

№	Наименование АТЕ	Количество участников экзамена	Доля участников, получивших тестовый балл				Количество участников, получивших 100 баллов
			ниже минимального	от минимального балла до 60 баллов	от 61 до 80 баллов	от 81 до 99 баллов	
1.	Ангарский городской округ	282	11	39	35,1	13,8	3
2.	Зиминское городское МО	6	50	16,7	33,3	0	0
3.	Зиминское районное МО	1	0	100	0	0	0
4.	г. Иркутск	571	10,5	33,5	35,6	20,3	1
5.	Иркутское районное МО	30	33,3	33,3	13,3	20	0
6.	МО Аларский район	9	0	44,4	33,3	22,2	0
7.	МО Балаганский район	6	50	50	0	0	0
8.	МО Баяндаевский район	4	25	50	25	0	0
9.	МО Боханский район	9	11,1	77,8	0	11,1	0
10.	МО Братский район	26	34,6	42,3	19,2	3,9	0

ГАУ ДПО ИРО РЦОИ

№	Наименование АТЕ	Количество участников экзамена	Доля участников, получивших тестовый балл				Количество участников, получивших 100 баллов
			ниже минимального	от минимального балла до 60 баллов	от 61 до 80 баллов	от 81 до 99 баллов	
11.	МО город Саянск	33	18,2	39,4	30,3	12,1	0
12.	МО город Свирск	6	0	50	16,7	33,3	0
13.	МО город Тулун	16	31,3	37,5	25	6,3	0
14.	МО город Усолье-Сибирское	45	6,7	37,8	40	15,6	0
15.	МО город Усть-Илимск	73	8,2	41,1	30,1	20,6	0
16.	МО город Черемхово	19	10,5	47,4	36,8	5,3	0
17.	МО города Бодайбо и района	22	9,1	63,6	18,2	9,1	0
18.	МО города Братска	251	16,3	47	27,5	8,4	2
19.	МО Жигаловский район	4	0	50	50	0	0
20.	МО Заларинский район	7	0	71,4	28,6	0	0
21.	МО Иркутской области Казачинско-Ленский район	11	36,4	54,6	9,1	0	0
22.	МО Катангский район	2	0	100	0	0	0
23.	МО Качугский район	3	0	66,7	33,3	0	0
24.	МО Киренский район	6	33,3	50	16,7	0	0
25.	МО Куйтунский район	8	12,5	75	0	12,5	0
26.	МО Мамско-Чуйский район	3	33,3	66,7	0	0	0
27.	МО Нижнеилимский район	34	5,9	52,9	38,2	2,9	0
28.	МО Нижнеудинский район	52	25	55,8	19,2	0	0
29.	МО Нукутский район	2	0	50	50	0	0
30.	МО Осинский район	11	0	63,6	27,3	9,1	0
31.	МО Слюдянский район	40	5	52,5	35	7,5	0
32.	МО Тайшетский район	40	12,5	55	32,5	0	0
33.	МО Тулунский район	2	0	100	0	0	0
34.	МО Усть-Илимский район	6	33,3	50	16,7	0	0
35.	МО Эхирит-Булагатский район	13	15,4	46,2	23,1	15,4	0
36.	Ольхонское районное МО	6	16,7	33,3	50	0	0
37.	Усольское районное МО	18	33,3	27,8	33,3	5,6	0
38.	Усть-Кутское МО	39	28,2	48,7	20,5	2,6	0
39.	Черемховское районное МО	10	50	20	30	0	0
40.	Чунское районное МО	20	15	55	30	0	0
41.	МО Шелеховский район	69	5,8	34,8	44,9	14,5	0

ГАУ ДПО ИРО, РЦОИ

2.4. Перечень ОО, продемонстрировавших высокие результаты ЕГЭ по предмету

В экзамене по информатике и ИКТ участвовало 332 ОО Иркутской области. Из них только в 44 ОО количество участников ЕГЭ составило не менее 10 человек.

Выборка ОО, продемонстрировавших высокие результаты, проводилась с учетом количества участников экзамена от 10 и выше; доля участников ЕГЭ, **получивших от 61 до 100 баллов,** имеет ***значение не ниже 60;*** доля участников ЕГЭ, **не достигших минимального балла**, имеет ***значение ноль*** (по сравнению с другими ОО Иркутской области).

Таблица 10

№	Наименование ОО	Количество участников экзамена	Доля участников, получивших от 81 до 100 баллов	Доля участников, получивших от 61 до 80 баллов	Доля участников, не достигших минимального балла
1.	МАОУ «Экспериментальный лицей "Научно-образовательный комплекс"», г. Усть-Илимск	13	69,2	15,4	0
2.	МАОУ «Гимназия № 8», г. Ангарск	17	64,7	35,3	0
3.	МБОУ г. Иркутска «Лицей-интернат № 1»	14	64,3	21,4	0
4.	МАОУ «Лицей ИГУ» г. Иркутска	42	57,1	38,1	0
5.	МБОУ «Гимназия № 44» г. Иркутска	14	57,1	21,4	0
6.	МБОУ г. Иркутска «Лицей № 3»	41	39	53,7	0
7.	МБОУ «СОШ № 10», г. Ангарск	26	30,8	42,3	0
8.	НОУ «Лицей № 36» ОАО «РЖД», г. Иркутск	26	30,8	38,5	0
9.	МБОУ «Лицей № 1», г. Усолье- Сибирское	14	28,6	50	0
10.	МБОУ г. Братска «Лицей № 2»	23	26,1	56,5	0
11.	МБОУ г. Иркутска «Гимназия № 3»	20	25	40	0
12.	МБОУ г. Братска «Лицей № 1»	20	25	20	0
13.	МБОУ г. Иркутска «Лицей № 2»	37	24,3	62,2	0
14.	МАОУ «Ангарский лицей № 2 им. М. К. Янгеля»	59	20,3	59,3	0
15.	МБОУ ШР «Гимназия», г. Шелехов	11	18,2	45,5	0

ГАУДПО ИРО РЦОИ

2.5. Перечень ОО, продемонстрировавших <u>низкие результаты ЕГЭ</u> по предмету

Выборка ОО, продемонстрировавших низкие результаты, проводилась с учетом количества участников экзамена от 10 и выше; доля участников ЕГЭ, **получивших от 61 до 100 баллов,** имеет ***значение 40 и ниже;*** доля участников ЕГЭ, **не достигших минимального балла**, имеет ***значение 15 и выше*** (по сравнению с другими ОО Иркутской области).

Таблица 11

№	Наименование ОО	Количество участников экзамена	Доля участников, получивших от 81 до 100 баллов	Доля участников, получивших от 61 до 80 баллов	Доля участников, не достигших минимального балла
1.	МБОУ г. Иркутска «СОШ № 11»	20	5	35	15

№	Наименование ОО	Количество участников экзамена	Доля участников, получивших от 81 до 100 баллов	Доля участников, получивших от 61 до 80 баллов	Доля участников, не достигших минимального балла
2.	МБОУ г. Братска «СОШ № 3»	22	4,55	31,82	36,36
3.	МБОУ «СОШ № 5», г. Ангарск	14	0	14,29	21,43
4.	МБОУ г. Братска «Лицей № 3»	10	0	30	20
5.	МБОУ г. Братска «СОШ № 46»	17	0	5,88	23,53

2.6. Вывод о характере изменения результатов ЕГЭ по предмету

В дополнение к Диаграмме распределения тестовых баллов в 2019 году приведем диаграмму распределения тестовых баллов в 2017–2019 годах. Из диаграммы видно, что в целом ЕГЭ по информатике и ИКТ сдали лучше, чем в предыдущие два года. Как и в 2017, 2018 годах, в 2019 году максимальный процент составляют участники ЕГЭ, набравшие 41 до 50 баллов, но сам показатель в этом диапазоне и в двух последующих ниже, чем в предыдущие годы. Положительным моментом является то, что уменьшение процента участников, набравших от 41 до 70 баллов, произошло за счет того, что значительно увеличился процент участников, набравших 71–100 баллов. Также уменьшился процент участников, не набравших минимальный балл с 19,6 до 13,2.

ЛО ИРО РЦОИ

Выпускники текущего года, обучающиеся по программам СОО, показывают существенно лучшие результаты по сравнению с обучающимися по программам СПО и выпускниками прошлых лет.

Как и в прошлые годы, лучшие результаты показали выпускники классов с углубленным изучением информатики и ИКТ, а худшие – выпускники универсальных классов. Такой результат является закономерным, поскольку контрольные измерительные материалы содержат как задания базового уровня сложности, проверяющие знания и умения, предусмотренные стандартами базового уровня подготовки по предмету, так и задания повышенного и высокого уровней, проверяющие знания и умения, предусмотренные профильным стандартом.

Среди АТЕ традиционно лучшие результаты достигнуты в МО Шелеховский район и МО город Иркутск. К ним присоединились МО город Усолье-Сибирское, МО город Усть-Илимск. В перечисленных АТЕ больше половины участников набрали выше 60 баллов. Максимальная доля участников, набравших балл ниже минимального, отмечается в МО Балаганского района, Черемховском районном МО, Зиминском городском МО (50 %).

Среди ОО, демонстрирующих высокие результаты ЕГЭ по предмету, как и раньше, на первых позициях стоит МАОУ «Лицей ИГУ» г. Иркутска (по соотношению количества и качества). В числе лидеров остались МАОУ «Гимназия № 8» и МБОУ «СОШ № 10» Ангарского городского округа, МБОУ г. Иркутска «Лицей № 3» и «Гимназия № 44» г. Иркутска. Отличные результаты показали такие ОО, как МАОУ «Экспериментальный лицей "Научно-образовательный комплекс"» МО города Усть-Илимск, МБОУ г. Иркутска «Лицей-интернат № 1» г. Иркутска, НОУ «Лицей № 36» ОАО «РЖД» г. Иркутска, МБОУ «Лицей № 1» МО город Усолье-Сибирское, МБОУ г. Братска «Лицей № 2», МО города Братска, МБОУ г. Иркутска «Лицей № 2» г. Иркутска. Во всех перечисленных ОО нет ни одного ученика, не набравшего минимальный балл. Из таблицы 13 видно, что в 5 ОО из перечисленных больше половины участников ЕГЭ набрали балл от 81 до 100. И во всех перечисленных ОО больше 80 % участников ЕГЭ по информатике и ИКТ набрали больше 60 баллов.

В перечне ОО, продемонстрировавших низкие результаты ЕГЭ по предмету, вошли 5 школ: МБОУ г. Иркутска СОШ № 11, МБОУ «СОШ № 5», г. Ангарск, МБОУ г. Братска «СОШ № 3», МБОУ г. Братска «Лицей № 3», МБОУ г. Братска «СОШ № 46».

Стоит отметить, что в регионе достаточно в большом количестве ОО, в которых в ЕГЭ принимали участие 1 или несколько человек, ни один участник не справился с экзаменом. Причем в подавляющем большинстве случаев в этих ОО ученики не сдавали данный экзамен, либо сдавали с таким же показателем, что говорит об отсутствии практики преподавания данной дисциплины в этих ОО на достаточном уровне.

ГАУ АПО ИРО, РЦОИ

III. АНАЛИЗ РЕЗУЛЬТАТОВ ВЫПОЛНЕНИЯ ОТДЕЛЬНЫХ ЗАДАНИЙ ИЛИ ГРУПП ЗАДАНИЙ

3.1. Краткая характеристика КИМ по учебному предмету

Содержание заданий разработано по основным темам курса информатики и ИКТ, объединенных в следующие тематические блоки: «Информация и ее кодирование», «Моделирование и компьютерный эксперимент», «Системы счисления», «Логика и алгоритмы», «Элементы теории алгоритмов», «Программирование», «Архитектура компьютеров и компьютерных сетей», «Обработка числовой информации», «Технологии поиска и хранения информации».

Содержанием экзаменационной работы охватывается основное содержание курса информатики и ИКТ, важнейшие его темы, наиболее значимый в них материал, однозначно трактуемый в большинстве преподаваемых в школе вариантов курса информатики и ИКТ.

Работа содержит как задания базового уровня сложности, проверяющие знания и умения, предусмотренные стандартом базового уровня, так и задания повышенного и высокого уровней сложности, проверяющие знания и умения, предусмотренные стандартом профильного уровня.

В КИМ используются задания двух типов: с кратким ответом и развернутым ответом. Структура экзаменационной работы обеспечивает оптимальный баланс заданий разных типов и разновидностей, трех уровней сложности, проверяющих знания и умения на трех различных уровнях: воспроизведения, применения в стандартной ситуации, применения в новой ситуации.

Каждый вариант экзаменационной работы состоит из двух частей и включает в себя 27 заданий, различающихся формой и уровнем сложности.

Часть 1 содержит практически поровну задания базового и повышенного уровней (12 и 10, соответственно) и одно задание высокого уровня сложности. В этой части собраны задания с кратким ответом, подразумевающие самостоятельное формулирование и запись ответа в виде числа или последовательности символов. Задания проверяют материал всех тематических блоков.

Часть 2 содержит 4 задания, первое из которых повышенного уровня сложности, остальные 3 задания высокого уровня сложности. Задания этой части подразумевают запись развернутого ответа в произвольной форме с выполнением строго определенных требований. Задания части 2 направлены на проверку сформированности важнейших умений записи и анализа алгоритмов. Эти умения проверяются на повышенном и высоком уровнях сложности. Также на высоком уровне сложности проверяются умения по теме «Технология программирования».

По открытому варианту КИМ можно сделать вывод, что предложенные задания полностью соответствуют спецификации КИМ. Можно отметить, что приведенное задание 19 полностью соответствует спецификации, в которой в обобщенном варианте КИМ ЕГЭ для данного задания предусмотрена «Работа с

ГАУ ДПО ИРО, РЦОИ

массивами (заполнение, считывание, поиск, сортировка, массовые операции и др.)». Но сам представленный алгоритм требует от учеников большое количество мелких вычислений, которые могут приводить к арифметическим ошибкам. В результате, ошибки при выполнении данного задания показывают не только неумение работать с массивами данных, но и неспособность в стрессовой ситуации в условиях ограниченного времени верно выполнить большое количество мелких вычислений.

Тоже можно сказать про задание 18, которое должно проверять «Знание основных понятий и законов математической логики». Но предложенное задание может вызвать у учеников трудности с интерпретацией математической части данного задания и с логическим мышлением в целом.

Стоит отметить задание 27, которое было представлено в открытом варианте КИМ в 2019 году. Само задание и критерии оценки составлены таким образом, чтобы максимально качественно ранжировать учеников, выполняющих данное задание, по уровню подготовки: если ученик не понимает предложенную формулировку эффективной по времени программы, он скорее всего предложит алгоритм, эффективный по памяти, но неэффективный по времени, вследствие чего не сможет получить максимальный балл. То есть в данном задании проверяются не только умение ученика писать код, но и понимание, сколько места выделяется под используемые им переменные, сколько операций выполняется в ходе выполнения алгоритма, если он использует стандартные алгоритмы, то также должен знать или уметь посчитать сложность этих алгоритмов. Спорным моментом в плане проверки «Умения создавать собственные программы (30–50 строк) для решения задач средней сложности» являются следующие ситуации: 1) ученик пишет эффективную верную программу, но допускает при ее записи ошибку, которая может трактоваться и как описка, и как содержательная, не перечисленная в критериях; в результате решение оценивается в 1 балл; 2) неэффективное переборное решение оценивается в 2 балла.

ГАУ АПО ИРО, РЦОИ

3.2. Анализ КИМ

Анализ проводится в соответствии с методическими традициями предмета и особенностями экзаменационной модели по предмету.

В качестве приложения используется план КИМ по предмету с указанием средних процентов выполнения по каждой линии заданий открытого варианта КИМ в регионе.

Заметим также, что общая картина выполнения заданий в регионе по всем вариантам схожа с представленной по открытому варианту КИМ, но в большинстве заданий процент выполнения чуть выше. При анализе такие случаи будут оговорены.

Таблица 12

Обозначение задания в работе	Проверяемые элементы содержания / умения	Уровень сложности задания	Процент выполнения задания в Иркутской области[1]			
			средний	в группе не преодолевших минимальный балл	в группе 61–80 т. б.	в группе 81–100 т. б.
1.	Знания о системах счисления и двоичном представлении информации в памяти компьютера	Б	83,8	50	91,2	100
2.	Умения строить таблицы истинности и логические схемы	Б	58,1	0	91,2	100
3.	Умение представлять и считывать данные в разных типах информационных моделей (схемы, карты, таблицы, графики и формулы)	Б	88,1	61,5	98,5	92,9
4.	Знания о файловой системе организации данных или о технологии хранения, поиска и сортировки информации в базах данных	Б	82,4	53,9	88,2	100
5.	Умение кодировать и декодировать информацию	Б	81	46,2	92,7	100
6.	Формальное исполнение алгоритма, записанного на естественном языке или умение создавать линейный алгоритм для формального исполнителя с ограниченным набором команд	Б	49,5	3,9	75	85,7
7.	Знание технологии обработки информации в электронных таблицах и методов визуализации данных с помощью диаграмм и графиков	Б	80,5	26,9	95,6	100
8.	Знание основных конструкций языка программирования, понятия переменной, оператора присваивания	Б	82,4	46,2	94,1	100
9.	Умение определять скорость передачи информации при заданной пропускной способности канала, объем памяти, необходимый для хранения звуковой и графической информации	Б	51,4	7,7	82,4	100
10.	Знания о методах измерения количества информации	Б	55,2	7,7	88,2	100
11.	Умение исполнить рекурсивный алгоритм	Б	49,1	0	79,4	92,9
12.	Знание базовых принципов организации и функционирования компьютерных сетей, адресации в сети	Б	45,7	7,7	70,6	100

ГАУ ДПО ИРО, РЦОИ

[1]Сумма первичных баллов, полученных всеми участниками группы за выполнение задания, отнесенная к количеству участников группы на максимальный первичный балл, который можно получить за выполнение задания.

Обозначение задания в работе	Проверяемые элементы содержания / умения	Уровень сложности задания	Процент выполнения задания в Иркутской области[1]			
			средний	в группе не преодолевших минимальный балл	в группе 61–80 т. б.	в группе 81–100 т. б.
13.	Умение подсчитывать информационный объем сообщения	П	52,4	0	86,8	100
14.	Умение исполнить алгоритм для конкретного исполнителя с фиксированным набором команд	П	26,7	3,9	36,8	71,4
15.	Умение представлять и считывать данные в разных типах информационных моделей (схемы, карты, таблицы, графики и формулы)	П	71,9	23,1	80,9	100
16.	Знание позиционных систем счисления	П	45,7	3,85	70,6	100
17.	Умение осуществлять поиск информации в сети Интернет	П	62,4	11,5	82,4	100
18.	Знание основных понятий и законов математической логики	П	19,1	7,7	32,4	57,1
19.	Работа с массивами (заполнение, считывание, поиск, сортировка, массовые операции и др.)	П	12,4	0	20,6	64,3
20.	Анализ алгоритма, содержащего цикл и ветвление	П	41	7,7	66,2	92,9
21.	Умение анализировать программу, использующую процедуры и функции	П	25,2	0	48,5	100
22.	Умение анализировать результат исполнения алгоритма	П	44,3	0	75	92,9
23.	Умение строить и преобразовывать логические выражения	В	11	3,9	19,1	57,1
24.	Умение прочесть фрагмент программы на языке программирования и исправить допущенные ошибки	П	39,1	0	76,5	97,6
25.	Умения написать короткую (10—15 строк) простую программу на языке программирования	В	20	0	40,4	96,4
26.	Умение построить дерево игры по заданному алгоритму и обосновать выигрышную стратегию	В	26,7	1,3	47,6	90,5
27.	Умения создавать собственные программы (30—50 строк) для решения задач средней сложности	В	5	0	7,7	37,5

ГАУ ДПО ИРО РДОИ

В дальнейшем анализе для компактности выделенные группы обучающихся будут пронумерованы: 1 – обучающиеся, не преодолевшие минимальный порог, 2 – обучающиеся, набравшие от 61 до 80 баллов, 3 – обучающиеся, набравшие от 81 до 100 баллов.

В основном, тенденция выполнения определенных заданий схожа во всех группах. То есть, если с каким-то заданием лучше всего справились в одной

группе, то и в других группах с ним справились сопоставимо, только с другими значениями процентов.

На базовом уровне сложности наиболее успешно (средний процент выполнения которых выше 80 %) были выполнены приведенные ниже задания (в скобках указан проверяемый элемент содержания и средний процент выполнения открытого варианта и по всем вариантам). Заметим, что с перечисленными заданиями, кроме задания 3, справилось 100 % участников третьей группы, выполнявших открытый вариант КИМ. Номера заданий приведены в порядке убывания среднего процента выполнения:

– 3 (умение представлять и считывать данные в разных типах информационных моделей, 88,1 % и 90,3 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 61,5 %, 98,5 %, 92,9 %. Результат выполнения этого задания по группам участников в целом по региону: 58,8 %, 97,9 %, 98 %.

– 1 (знания о системах счисления и двоичном представлении информации в памяти компьютера, 83,8 % и 81,4 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 50 %, 91,2 %, 100 %. Результат выполнения этого задания по группам участников в целом по региону: 40,8 %, 93,3 %, 99,6 %.

– 4 (знания о файловой системе организации данных или о технологии хранения, поиска и сортировки информации в базах данных, 82,4 % и 79,4 %). Это задание оказалось наиболее легким для первой группы участников. Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 53,9 %, 88,2 %, 100 %, Результат выполнения этого задания по группам участников в целом по региону: 48,6 %, 87,8 %, 96,4 %.

– 8 (знание основных конструкций языка программирования, понятия переменной, оператора присваивания, 82,4 % и 83,4 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 46,2 %, 94,1 %, 100 %, Результат выполнения этого задания по группам участников в целом по региону: 41,2 %, 95,2 %, 97,6 %. Заметим, что в целом по области ученики второй группы справились с этим заданием лучше всего.

– 5 (умение кодировать и декодировать информацию, 80,9 % и 76,3 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 46,2 %, 92,7 %, 100 %. Результат выполнения этого задания по группам участников в целом по региону: 21,2 %, 92,1 %, 97,2 %.

– 7 (знание технологии обработки информации в электронных таблицах и методов визуализации данных с помощью диаграмм и графиков). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 26,9 %, 95,6 %, 100 %, Результат выполнения этого задания по группам участников в целом по региону: 25,1 %, 94,7 %, 98,4 %. Заметим, что ученики второй группы, выполнявшие открытый вариант КИМ, справились с этим заданием лучше всего.

ГАУ ДПО ИРО. РЦОИ

Заметим, что процент выполнения заданий 8 и 7 участниками, выполнявшими открытый вариант ЕГЭ и в целом по региону, отличается практически в 2 раза, что может говорить о том, что задания в разных вариантах были разной сложности.

С остальными заданиями базовой части в среднем справились значительно хуже: если средний процент выполнения перечисленных выше заданий больше 80, то для всех остальных заданий базового уровня – меньше 60. Меньше половины участников, выполнявших открытый вариант КИМ, справились с тремя заданиями базового уровня:

– 6 (формальное исполнение алгоритма, записанного на естественном языке или умение создавать линейный алгоритм для формального исполнителя с ограниченным набором команд, 49,5 % и 50,3 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 3,9 %, 75 %, 85,7 %, Результат выполнения этого задания по группам участников в целом по региону: 5,9 %, 70,5 %, 92,7 %.

– 11 (умение исполнить рекурсивный алгоритм, 49,1 % и 51 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 0 %, 79,4 %, 92,9 %, Результат выполнения этого задания по группам участников в целом по региону: 0,8 %, 79,7 %, 96,8 %.

– 12 (знание базовых принципов организации и функционирования компьютерных сетей, адресации в сети, 45,7 % и 53,6 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 7,7 %, 70,6 %, 100 %, Результат выполнения этого задания по группам участников в целом по региону: 3,5 %, 80,9 %, 96 %.

На повышенном уровне сложности наиболее успешно (средний процент выполнения которых больше 50 %) были выполнены следующие задания:

– 15 (умение представлять и считывать данные в разных типах информационных моделей, 71,9 % и 68,3 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 23,1 %, 80,9 %, 100 %, Результат выполнения этого задания по группам участников в целом по региону: 27,5 %, 82,2 %, 92,3 %.

– 17 (умение осуществлять поиск информации в сети Интернет, 62,4 % и 63,9 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 11,5 %, 82,4 %, 100 %, Результат выполнения этого задания по группам участников в целом по региону: 7,8 %, 85,7 %, 99,2 %.

– 13 (умение подсчитывать информационный объем сообщения, 52,4 % и 52,6 %). Результат выполнения этого задания по группам участников, выполнявших открытый вариант КИМ: 0 %, 86,8 %, 100 %. Результат выполнения этого задания по группам участников в целом по региону: 1,2 %, 83,5 %, 96,8 %.

На повышенном уровне сложности участники ЕГЭ первой группы, выполнявшие открытый вариант КИМ, совсем не справились с рядом заданий. Ученики, выполнявшие остальные, относящиеся к первой группе, также либо

ГАУ ДПО ИРО РЦОИ

не справились с этими заданиями, либо справилось совсем небольшое количество. Перечислим эти задания.

– 13 (умение подсчитывать информационный объем сообщения).

– 22 (умение анализировать результат исполнения алгоритма).

– 24 (умение прочесть фрагмент программы на языке программирования и исправить допущенные ошибки).

– 21 (умение анализировать программу, использующую процедуры и функции).

– 19 (Работа с массивами).

Для второй и третьей группы на повышенном уровне также сложным оказалось задание 19, а также задания 14 и 18. Результат выполнения этих заданий во второй группе участников, выполнявших открытый вариант КИМ: 20,6 %, 36,8 %, 32,4 %, в целом по региону: 36,7 %, 43,6 %, 37,2 %. Результат выполнения этих заданий в третьей группе участников, выполнявших открытый вариант КИМ: 64,3 %, 71,4 %, 57,1 %, в целом по региону: 76,2 %, 69,4 %, 74,2 %. Для участников второй группы, чуть менее сложным, чем перечисленные, но с которым справилось меньше половины участников этой группы, оказалось еще задание 21. Заметим, что участники третьей группы с остальными заданиями повышенного уровня справились более чем в 90 % случаев, а участники второй группы — более чем в 65 % случаев.

Интересно, что на высоком уровне сложности, среди участников первой группы, больше всего участников справилось с заданием 23, и чуть меньше — с заданием 26, с остальными заданиями не справился никто.

Для второй и третьей группы на высоком уровне более простыми оказались задания 26 и 25. Результат выполнения этих заданий во второй группе участниками, выполнявшими открытый вариант КИМ: 47,6 %, 40,4 %, в целом по региону: 55,8 %, 49,9 %. Результат выполнения этих заданий в третьей группе участниками, выполнявшими открытый вариант КИМ: 90,5 %, 96,4 %, в целом по региону: 92,7 %, 92,5 %.

Для второй и третьей группы на высоком уровне сложными оказались задания 23 и 27. Результат выполнения этих заданий во второй группе участниками, выполнявшими открытый вариант КИМ: 19,1 %, 7,7 %, в целом по региону: 14,7 %, 8,2 %. Результат выполнения этих заданий в третьей группе участниками, выполнявшими открытый вариант КИМ: 57,1 %, 37,5%, в целом по региону: 65,7 %, 41,8 %.

Из приведенных данных получается, что на всех уровнях сложности и во всех группах основные проблемы у участников экзамена возникают с заданиями из двух разделов: «Логика и алгоритмы» и «Программирование».

3.3. Характеристики выявленных сложных для участников ЕГЭ заданий с указанием типичных ошибок и выводов о вероятных причинах затруднений при выполнении указанных заданий

Как уже было сказано, что на всех уровнях сложности и во всех группах основные проблемы у участников экзамена возникают с заданиями из двух разделов: «Логика и алгоритмы» и «Программирование».

ГАУ АПО ИРО, ЦОИ

На базовом уровне у учеников во всех группах возникают сложности с заданиями 6 и 12. По кодификатору задание 6 проверяет: «Формализация понятия алгоритма» и «Построение алгоритмов и практические вычисления», а задание 12 – «Программная и аппаратная организация компьютеров и компьютерных систем. Виды программного обеспечения». Но на самом деле эти задания комплексные и требуют знаний и умений в разных разделах, за счет чего и возникают дополнительные ошибки.

Наиболее распространенные ошибки при выполнении задания 6:

1. Вычислительные ошибки при переводе из одной системы счисления в другую.
2. Невнимательность при интерпретации вывода ответа.
3. Сложности с пониманием результата работы алгоритма.
4. Невнимательность при выполнении отдельных шагов алгоритма.

Наиболее распространенные ошибки при выполнении задания 12:

1. Вычислительные ошибки при переводе из одной системы счисления в другую.
2. Недостаточно развитое логическое мышление (ученики не всегда понимают, почему в одном случае в ответ записывается 0, а в другом 1, хотя базовые логические операции выполняются ими на хорошем уровне).
3. Непонимание принципов адресации в компьютерных сетях.

Сложности при выполнении задания 11, проверяющего «Умение исполнить рекурсивный алгоритм», возникают в основном в связи с недостаточной алгоритмической подготовкой учеников. Как правило, тема «Рекурсивные алгоритмы» либо не изучается на уроках в связи с нехваткой времени, либо прорабатывается на недостаточном уровне.

Также на базовом уровне возникли проблемы с применением знаний по темам «Количество информации» и «Элементы комбинаторики». Например, если посмотреть на ответы, данные участниками экзамена при решении открытого варианта КИМ, в задании 9 можно сделать вывод, что многие не понимают разницы между количеством использованных цветов и количеством бит, выделяемых для хранения одного пикселя изображения. То есть ошибка в понимании термина «Количество информации». Ответы, данные при решении задания 10, показывают, что многие не понимают, как в принципе необходимо решать данную задачу и большинство действий делают наугад, не используя принципы комбинаторики.

Четыре из пяти заданий, с которыми не справились участники ЕГЭ первой группы, проверяют знания по темам «Алгоритмы» и «Программирования». Проблема возникает в связи со сложностью восприятия данной темы учениками и слабым уровнем развития практических навыков по данным темам в классах с базовой подготовкой. Такой вывод можно сделать, посмотрев, что с аналогичными заданиями участники второй и третьей групп справились гораздо более успешно, а в них в основном входят ученики профильных классов по Информатике и ИКТ.

ГАУ ДПО ИРО, РЦОИ

Сложности с выполнением задания 14 во второй и третьей группах возникают не столько с неумением применять алгоритм, сколько с необходимостью обобщать информацию и делать выводы.

При выполнении задания 18 у учеников второй и третьей групп трудности возникают с интерпретацией математической части данного задания и с логическим мышлением в целом.

Все задания высокого уровня, очевидно, для большинства являются слишком сложными, многие к ним даже не приступали, а справилось с ними не больше трети участников.

Самым «легким» из этих заданий традиционно является задание 26. Как обычно, наиболее распространенными ошибками являются ошибки описания выигрышной стратегии:

1. В 2019 году в задании каждый раз у игроков было четыре варианта хода вместо четырех, но ученики по привычке при описании стратегии указывали только два хода.
2. Ученики не понимают, зачем за проигрывающего игрока указывать «очевидно» проигрышные ходы.
3. При описании дерева игры при выполнении задания 3, ссылаются на деревья, построенные в заданиях 1 и 2.

Следующим по сложности было задание 25. Большинство учеников, приступающих к выполнению этого задания, справляются с ним. Наиболее частые ошибки, которые все же допускаются:

1. Ученики выводят нужные значения, но не изменяют исходный массив.
2. Ошибки инициализации.
3. Ошибки в расстановке операторных скобок.

К выполнению задания 23 многие также не приступали. Сложности в данном задании для учеников составляют:

1) реализация преобразования сложных логических выражений;
2) анализ и систематизация полученной информации;
3) непонимание, что является решением системы логических уравнений.

Задание 27 является самым сложным, и с ним справился даже меньший процент участников экзамена, чем в предыдущие 2 года. Это связано с тем, что формулировка задания в 2019 году немного изменилась, и многие не сориентировались, как создать эффективный алгоритм для ее решения. Также многие участники не понимают, как оценить разработанный алгоритм на эффективность, упускают из вида исключения, для которых необходимо в решение вводить дополнительные условия, данные или действия.

Далее рассмотрим примеры заданий, вызвавшие наибольшие затруднения у участников экзамена. Для лучшего понимания сути заданий дальнейшее изложение будем проводить в следующем порядке:

1. Пример задания из открытого варианта КИМ.

ГАУ АПО ИРО, РЦОИ

2. Пример решения приведенного задания с указанием сложных для учеников моментов.

3. Перечисление наиболее распространенных проблем, с которыми сталкиваются ученики при решении конкретной задачи и возможные пути решения этих проблем.

Задание 14.

Исполнитель Редактор получает на вход строку цифр и преобразовывает её. Редактор может выполнять две команды, в обеих командах v и w обозначают цепочки символов.

```
заменить (v, w)
нашлось (v)
```

Первая команда заменяет в строке первое слева вхождение цепочки v на цепочку w. Если цепочки v в строке нет, эта команда не изменяет строку. Вторая команда проверяет, встречается ли цепочка v в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение "истина", в противном случае возвращает значение "ложь".

На вход программы подали строку, состоящую из 57 идущих подряд цифр 9. Чему равна сумма цифр в строке, которая получится в результате выполнения следующей программы:

```
НАЧАЛО
ПОКА нашлось (222) ИЛИ нашлось (999)
ЕСЛИ нашлось (222)
ТО заменить (222, 19)
ИНАЧЕ заменить (999,2)
КОНЕЦ ЕСЛИ
КОНЕЦ ПОКА
КОНЕЦ
```

Решение:

1. Из программы видим, что Редактор выполняет действие только тогда, когда в строке есть цепочка 222 или цепочка 999; то есть, если ни одной из этих цепочек нет, программа останавливается.

2. Если в строке есть 222, то, в первую очередь, именно эта цепочка меняется (на 19).

3. Если в строке нет цепочки 222, но есть 999, то цепочка 999 меняется на 2.

4. Попробуем формально выполнить первые шаги алгоритма для цепочки из 57 цифр 9.

5. Сначала первые 999 меняются на 2, получается

2 [54 *цифры 9***]**

6. Дальше так же меняем следующие две тройки из цифр 9:

222 [*48 цифр 9***]**

7. Теперь (внимание!) у нас появилась цепочка 222, поэтому в соответствии с алгоритмом она сразу будет заменена на 19, получаем

1[*49 цифр 9***]**

ГАУ ДПО ИРО, РЦОИ

8. Таким образом, за первые четыре шага работы цикла мы удалили из исходной строки восемь цифр 9 и дописали слева одну цифру 1.

9. Очевидно, что следующие четыре шага удалят ещё восемь девяток и добавят еще одну цифру 1 слева и т. д. При этом надо помнить, что для удаления восьми девяток и добавления 1 необходимо наличие не восьми, а девяти девяток. Это условие можно сформулировать следующим образом:

```
ПОКА нашлось (999999999)
заменить (999999999, 19)
КОНЕЦ ПОКА
```

А когда останется меньше 9 девяток закончить алгоритм следующим образом:

```
ПОКА нашлось (999)
заменить (999,2)
КОНЕЦ ПОКА
```

10. Вычислим количество подобных преобразований для исходной строки: 57 : 8 = 7 и 1 в остатке. То есть, после завершения алгоритма исходная строка будет преобразована к следующему виду: семь единиц и одна девятка. Так как в строке больше не содержатся цепочки 222 или 999, то цикл завершает свою работу.

11. Итак, преобразованная цепочка имеет вид: 11111119. По условию задачи необходимо найти сумму цифр полученной цепочки. Эта сумма равна 16.

Ответ: 16.

Какие сложности возникают при решении этой задачи и на что необходимо обратить внимание?

Проблема 1. При выполнении пункта 7. Зачастую ученики увлекаются заменой первого типа и выполняют ее до конца, несмотря на условие алгоритма, то есть заменяют (999) на (2), пока не закончатся девятки, а потом начинают заменять (222) на (19).

Решение. Акцентировать внимание учеников на том, что нельзя торопиться с формализацией алгоритма, необходимо внимательно пошагово выполнить алгоритм до тех пор, пока не обозначатся четко действия цикла, которые необходимо формализовать.

Проблема 2. У учеников зачастую возникают сложности с выполнением пунктов 8 и 9: с обобщением полученной информации. То есть они могут выполнять действия пошагово, но не могут преобразовать эти действия в новый цикл.

Проблема 3. Также возникают сложности с выполнением пункта 10: с формализацией задачи. Для многих непонятно, что полученный цикл можно заменить делением и как интерпретировать полученные результаты деления.

Решение. Последние две проблемы можно решить, только увеличив объем задач, в которых необходимо анализировать действия алгоритма, делать выводы о результатах его выполнения. Причем исполнители и алгоритмы должны быть самыми разными, чтобы ученики учились именно обобщать и

ГАУ АПО ИРО, РЦОИ

формализовывать полученную из алгоритма информацию, а не тренировались подставлять цифры в задачи определенного типа.

Задание 18.

Для какого наименьшего целого неотрицательного числа *A* выражение

$$(x<A)\vee(y<A)\vee(x+2y>40)$$

тождественно истинно, то есть принимает значение 1 при любых целых неотрицательных *x* и *y*?

Решение:

Общие замечания к решению: при решении задач такого типа необходимо определить, когда предложенное сложное логическое высказывание тождественно истинно при любом *A*, а когда *A* должно принимать конкретные значения, чтобы выражение оставалось тождественно истинным. Для решения задачи интересен только второй случай.

Поэтому, как правило, первые шаги решения не зависят от сути простых высказываний, из которых состоит исходное. Можно, например, высказывание с *A* обозначить *A*, а высказывание без *A* обозначить *B* и пронумеровать эти высказывания, если высказываний первого или второго типа несколько. Тогда предложенное в примере высказывание примет вид:

$$A_1 \vee A_2 \vee B$$

Теперь можно приступать к решению задачи.

1. Если *B* истинно, то все выражение также будет истинно, независимо от значения A_1 и A_2 (так как все три условия связаны знаками дизъюнкции). Этот случай для решения задачи нам бесполезен.

2. Если *B* ложно, то, чтобы все выражение было истинно, необходимо, чтобы A_1 или A_2 были истинны. Выявили ситуацию, когда А должно принимать конкретные значения, чтобы выражение оставалось тождественно истинным. Далее будем работать с этим случаем.

3. Составим условие из предыдущего пункта, исключив логические операции:

Если *B* ложно, тогда A_1 — истинно или A_2 — истинно.

4. Произведем обратную замену:

Если $(x+2y>40)$ — ложно, тогда $(x<A)$ — истинно или $(y<A)$ — истинно.

5. Заметим, что $(x+2y>40)$ — ложно тогда, когда $(x+2y\le 40)$ —истинно.

6. Теперь немного уберем в получившемся выражении слово истинно, чтобы получилось понятное предложение не на языке логики, а на простом русском языке:

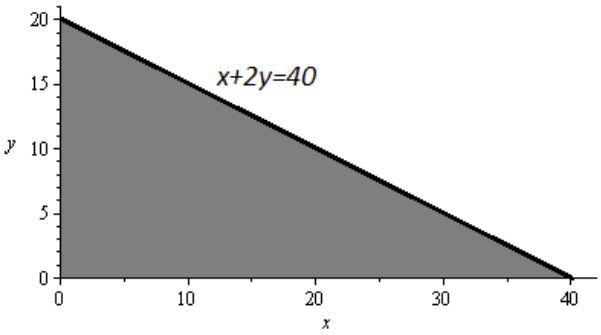
Если $(x+2y\le 40)$, тогда $(x<A)$ или $(y<A)$.

Что значит это выражение?

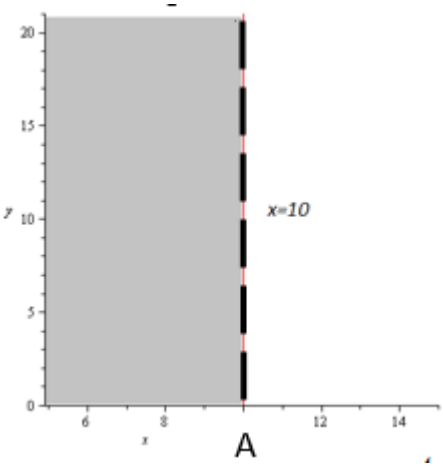
а. Начнем с первой части «если $(x+2y\le 40)$»: $(x+2y=40)$ — это уравнение прямой. Получаем, что выражение «если $(x+2y\le 40)$», обозначает, что точки с координатами *(x;y)* должны быть ниже (левее) прямой $(x+2y=40)$, включая точки прямой.

ГАУ ДПО ИРО, РЦОИ

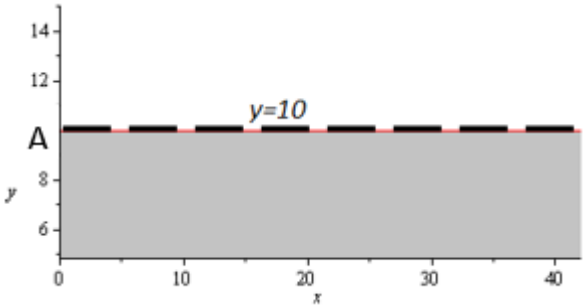
b. Построим график прямой $(x+2y=40)$ и обозначим область точек, удовлетворяющих условию $(x+2y\le 40)$ (точки с координатами $(x;y)$ ниже (левее) прямой $x+2y=40$ и на прямой). Не забудем, что по условию задачи x и y – целые неотрицательные (при обозначении области опустим пока условие, что x и y – целые).



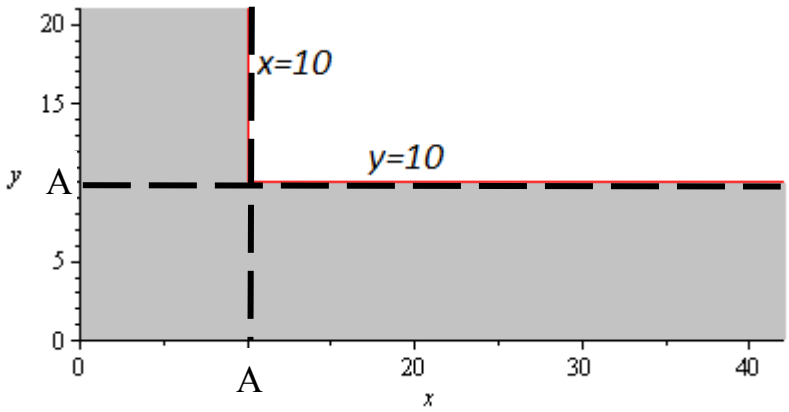
c. Вторая часть: «тогда $(x<A)$ или $(y<A)$». Посмотрим, что значит $x<A$ для конкретного $A=10$, то есть получим $x<10$. Так как $x=10$ — это уравнение прямой, тогда $x<10$ — это область, все точки которой находятся левее прямой $x=10$.



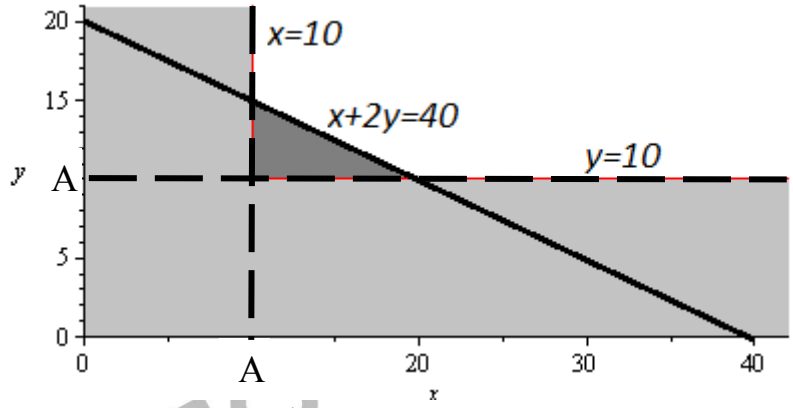
Теперь посмотрим, что значит $y<A$ для $A=10$, то есть получим $y<10$. Так как $y=10$ — это уравнение прямой, тогда $y<10$ — это область, все точки которой находятся ниже прямой $y=10$.



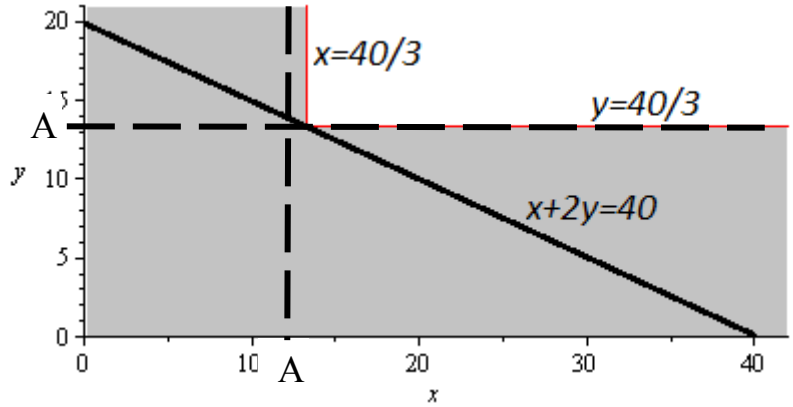
Так как высказывание звучит «тогда $(x<A)$ или $(y<A)$», нам подходят точки, представленные как на первом, так и на втором графике. Совместим их.



d. Теперь объединим обе части выражения: «Если $(x+2y\leq 40)$, тогда $(x<A)$ или $(y<A)$». Получаем, что **все точки с координатами (x, y) из области 1 (темно-серый цвет), должны находиться в области 2 (светло-серый цвет)**. Наложим область 2 на область 1.



7. Из построенного графика видно, то при $A=10$ полученное условие не выполняется: не все точки с координатами (x, y) из области 1 находятся под областью 2. То есть теперь надо подобрать A, чтобы выполнялось условие и A было минимальным. Очевидно, что условие выполняется, если A будет любое целое большее тому, которое изображено на графике ниже (так как область 2 целиком накрыла область 1):



8. Найдем для начала A для графика, изображенного на последнем рисунке. Для этого подставим в уравнение прямой $x+2y=40$ координаты точки пересечения всех трех прямых (A, A) и решим полученное уравнение:

$$A+2A=40; \qquad A=\frac{40}{3}.$$

9. Так как выражение $(x<A)\vee(y<A)\vee(x+2y>40)$ тождественно истинно для любого A большее найденного, выбираем минимальное целое, большего найденного значения $A=40/3$. Получаем $A=14$.

Ответ: 14.

Теперь рассмотрим, какие сложности возникают при решении этой задачи и на что необходимо обратить внимание.

Проблема 1. При выполнении первых пунктов решения, где необходимо выявить ситуацию, когда A должно принимать конкретные значения, чтобы выражение оставалось тождественно истинным.

Ученики достаточно хорошо умеют определять значение сложного логического выражения при известных значениях подвыражений, но для многих остается проблемой проанализировать логическое выражение и сделать нужные выводы. Также ученикам не всегда понятен смысл выполнения первых двух шагов.

Решение. Для того, чтобы ученикам стал понятен смысл выполнения первых двух шагов, необходимо при разных вариантах расстановки значений истинности простых высказываний, не зависящих от A, потренироваться определять истинность сложного высказывания при различных значениях x, y и A. Таким образом у них появляется осознание того, в каких случаях значение выражения зависит от A, а в каких — нет.

После этого необходимо для различных сложных выражений потренироваться расставлять значения истинности простых высказываний. Например, дается выражение $A\to B_1\vee B_2$. Объясняем, что сначала необходимо расставить скобки $(A\to B_1)\vee B_2$. Затем последовательно расставлять значения истинности тех подвыражений, которые не зависят от A таким образом, чтобы именно от значения A зависело значение всего выражения. То есть сначала определяем $B_2=0$, получаем $(A\to B_1)\vee 0$. Теперь, чтобы выражение было истинным, необходимо, чтобы $(A\to B_1)$ было истинным. Далее определяем $B_1=0$, получаем $(A\to 0)$. Теперь, чтобы выражение было истинным, необходимо, чтобы A было ложным.

Проблема 2. Следующая сложность возникает, когда необходимо сформулировать условие, которое будет далее анализироваться для решения задачи, как на шестом шаге предложенного решения.

Решение. На самом деле этой проблемы не должно возникать, если в самом начале изучения раздела «Логика» в курсе информатики научить детей формализовать задачу, выявляя в высказываниях, записанных на простом языке простые высказывания и записывая сложные высказывания с использованием логических связок. Например, в выражении «Мы собираем только съедобные грибы», выделяем простые высказывания и обозначаем их заглавными латинскими буквами: A — «Мы собираем гриб», B — «Гриб является съедобным». Тогда на языке логики выражение будет записано как $A\to B$ — «Если мы собираем гриб, то он съедобный». Далее необходимо научиться

ГАУ ДПО ИРО, РЦОИ

решать обратную задачу: имея сложное логическое высказывание и простые высказывания из которых оно строится, научиться формулировать правильное соответствующее ему выражение на русском языке.

То есть при последовательном и целостном подходе к изучению раздела «Логика» в курсе информатики данной проблемы возникнуть не должно.

Проблема 3. Самую большую сложность составляет у учеников провести правильную цепочку рассуждений, приводящих к верному ответу. Например, все действия задачи по отдельности выполняют хорошо. Для приведенного примера: расставляют значения истинности, составляют условие, которое должно быть выполнено, строят графики. Но не понимают, например, как выполнить шаг 6 в приведенном решении.

Решение. Необходимо решать на уроках большое количество текстовых логических задач, как с математическим содержанием, так и без него, постепенно повышая уровень сложности и заставляя учеников рассуждать над каждым выполненным действием.

Задание 19.

В программе используется одномерный целочисленный массив с индексами от 0 до 9. Значения элементов равны 20, 19, 17, 41, 15, 12, 24, 16, 4, 13 соответственно, т.е. A[0]=20, A[1]=19 и т.д. Определить значение переменной s после выполнения следующего фрагмента этой программы (программа приведена на языке программирования Паскаль):

```
s:=0;
n:=6;
for i:=0 to 9 do
if a[i]<=a[n] then begin
s:=s+i;
t:=a[i mod n];
a[i mod n]:=a[n];
a[n]:=t;
end;
```

Решение:

1. Проанализируем работу цикла, в котором происходит вычисление переменной *s*.
2. Переменная *s* вычисляется как сумма индексов элементов массива, удовлетворяющих условию **a[i]<=a[n].**
3. При *n=6* все элементы массива сравниваются с *a[6]*.
4. Кроме того происходит обмен значениями между элементом массива с индексом 6 и элементом массива с индексом, равным остатку от деления параметра *i* на 6.

Выполним прокрутку приведенной программы и определим, как будет изменяться массив и вычисляться значение переменной s. Исходный массив выглядит следующим образом:

20 19 17 41 15 12 24 16 4 13

ГАУ АПО ИРО РЦОИ

При *i=0* условие **a[i]<=a[6]** — истинно, следовательно, происходит вычисление переменной *s* (она остается равной 0) и выполняется обмен между a[0] и a[6]. Массив принимает следующий вид:

24 19 17 41 15 12 20 16 4 13

При *i=1* условие также истинно, *s* становится равной 1, а в массиве происходит перестановка:

24 20 17 41 15 12 19 16 4 13

При *i=2* условие также истинно, *s* становится равной 1+2=3, а в массиве происходит перестановка:

24 20 19 41 15 12 17 16 4 13

Следующая перестановка происходит при *i=4*. Переменная *s* становится равной 3+4=7, массив изменяется:

24 20 19 41 17 12 15 16 4 13

При *i=5* условие также истинно, *s* становится равной 7+5=12, а в массиве происходит перестановка:

24 20 19 41 17 15 12 16 4 13

При *i=6* условие истинно, так как в нем использован знак «≤». Переменная *s* становится равной 18. В массиве происходит обмен значениями между элементами массива с индексом 6 и индексом 0 (*a[6 mod 6]*). Массив приобретает вид:

12 20 19 41 17 15 24 16 4 13

При *i=7* условие истинно, *s* становится равной 25, массив изменяется:

12 24 19 41 17 15 20 16 4 13

При *i=8* условие истинно, *s=33*, в массиве происходит перестановка:

12 24 20 41 17 15 19 16 4 13

Наконец, при *i=9* условие истинно, *s=33+9=42*, в массиве происходит последняя перестановка:

12 24 20 19 17 15 41 16 4 13

Итак, ответ: 42.

Проблема 1. Ученики путают индекс элемента массива и его значение, забывают, что при присваивании переменной или элементу массива нового значения, старое стирается и т.п.

Решение. Для устранения таких ошибок нужна практика: то есть необходимо научиться выполнять различные действия с элементами массива на простых задачах, в которых будут присутствовать: 1) перебор определенного количества элементов массива, 2) изменение значений элементов массива с последующим их использованием.

Проблема 2. Многие не умеют или допускают ошибку в интерпретации действий алгоритма, необходимых для выполнения. Например, в текущем примере необходимо понять, что «если текущий элемент с индексом *i* меньше шестого элемента, необходимо выполнить обмен значениями между элементом массива с индексом 6 и элементом массива с индексом, равным остатку от деления параметра i на 6». То есть могут изменяться значения только первых семи элементов.

ГАУ АПО ИРО, РЦОИ

Зачастую ученики либо не могут сформулировать таким образом задачу, выполняя алгоритм пошагово и допуская при этом различного рода ошибки, либо допускают ошибки при самой интерпретации.

Решение. Необходимо давать разнообразные алгоритмы циклической обработки массива, в качестве решения которых ученики должны объяснять, какие действия необходимо произвести для выполнения алгоритма, и уже затем выполнить эти действия.

Проблема 3. При изучении раздела «Алгоритмизация и программирования» уделяется много внимания другим темам данного раздела и не хватает времени для изучения темы «Работа с массивами».

Решение. Для решения многих проблем, с которыми сталкиваются ученики при выполнении заданий, необходим, как уже упоминалось, последовательный и целостный подход при изучении каждого раздела курса.

Задание 21.

Напишите в ответе число, которое будет выведено в результате выполнения следующего алгоритма (программа приведена на языке Паскаль):

```
var a,b,t,M,R:longint;
function F(x:longint):longint;
 begin
  F:=2*(x*x-16)*(x*x-16)+41;
 end;
 begin
  a:=-20; b:=20;
  M:=a; R:=F(a);
  for t:=a to b do begin
   if (F(t)<=R) then begin
    M:=t;
    R:=F(t);
   end;
  end;
 writeln(M+R);
end.
```

Решение:

1. Заметим, что в программе есть цикл, в котором переменная *t* принимает последовательно все целые значения в интервале от *a* до *b*:

```
for t:=a to b do begin
...
end;
```

2. До начала цикла в переменную *M* записывается значение *a*, а в переменную *R* – значение функции в точке *a*:

```
M:=a; R:=F(a);
```

3. Внутри цикла есть условный оператор, в котором вычисляется значение функции *F(t)* и сравнивается со значением переменной *R*:

```
if (F(t)<=R)then begin
M:=t;
```

ГАУ ДПО ИРО, РЦОИ

```
R:=F(t);
end;
```

если новое значение функции меньше либо равно, чем значение *R*, в *R* записывается значение функции в точке *t*, а в переменной *M* запоминается само значение *t* (аргумент функции, соответствующий значению в *R*).

4. В результате анализа пп. 1-3 можно сделать вывод, что цикл ищет минимум функции *F(t)* на интервале от *a* до *b*, и после выполнения цикла в переменной *M* оказывается значение аргумента *t*, при котором функция достигает минимума на заданном интервале (здесь это интервал [-20, 20]). Само же значение минимума запоминается в переменную *R*.

5. Функция, которая используется в программе, – это квадратичная парабола: $y = 2 \cdot (x^2 - 16)^2 + 41$, её ветви направлены вверх (коэффициент при x^4 положительный, равен 2); она имеет два минимума в точках $x = -4$ и $x = 4$

6. Обе точки минимума находятся на отрезке [-20;20], поэтому программа найдёт одну из этих точек.

7. Для квадратичной параболы обе точки минимума имеют одинаковую y-координату, а запоминание новой точки минимума происходит только тогда, когда только что вычисленное значение *F(t)* станет **меньше либо равно**, чем хранящееся в переменной *R*:

```
if (F(t) <= R)then begin
M:=t;
R:=F(t);
end;
```

8. Поэтому будет найден **второй** минимум $x = 4$, и в переменной *M* окажется значение «4». Значение функции в этой точке будет равным 41 (*R=41*).

9. Обратим внимание, что на экран выводится *M*+*R*, поэтому результат будет равен 4+41=45.

Ответ: 45.

Проблема. При решении данной задачи ученики сталкиваются с теми же проблемами, что и в предыдущей задаче:

1. Интерпретация необходимых для выполнения действий алгоритма.
2. Нехватка времени на изучение темы «Процедуры и функции».

Решение. Соответственно, для решения этих проблем рекомендации аналогичные тем, что были даны в предыдущей задаче.

Задание 23.

Сколько существует различных наборов значений логических переменных *x1, x2, … x6, y1, y2, … y6*, которые удовлетворяют всем перечисленным ниже условиям:

$$(x1 \vee \neg x2) \wedge (y1 \vee \neg y2) \wedge (x1 \vee \neg y1) = 1$$
$$(x2 \vee \neg x3) \wedge (y2 \vee \neg y3) \wedge (x2 \vee \neg y2) = 1$$
$$\ldots$$
$$(x5 \vee \neg x6) \wedge (y5 \vee \neg y6) \wedge (x5 \vee \neg y5) = 1$$

ГАУ ДПО ИРО, РЦОИ

$$x6 \vee \neg y6 = 1$$

В ответе не нужно перечислять все различные наборы значений переменных *x1, x2, ... x6, y1, y2, ... y6*, при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких наборов.

Решение:

Для начала заметим, что наиболее часто для решения систем логических уравнений используются два метода: «цепочек» и «отношений». Первый в основном используется, если мы видим, что выражения можно свести к уравнениям вида:

$$\mathbf{(A \to B) \wedge (B \to C) \wedge (C \to D) \wedge \ldots = 1}$$

или

$$\mathbf{(A \equiv B) \wedge (B \equiv C) \wedge (C \equiv D) \wedge \ldots = 1.}$$

Представленная задача именно такого типа, поэтому покажем, как она решается методом «цепочек».

1. Сначала перегруппируем выражения в скобках и запишем систему в видоизмененном виде:

$$(x1 \vee \neg x2) \wedge (x2 \vee \neg x3) \wedge (x3 \vee \neg x4) \wedge (x4 \vee \neg x5) \wedge (x5 \vee \neg x6) = 1 \quad (1)$$
$$(y1 \vee \neg y2) \wedge (y2 \vee \neg y3) \wedge (y3 \vee \neg y4) \wedge (y4 \vee \neg y5) \wedge (y5 \vee \neg y6) = 1 \quad (2)$$
$$(x1 \vee \neg y1) \wedge (x2 \vee \neg y2) \wedge (x3 \vee \neg y3) \wedge (x4 \vee \neg y4) \wedge (x5 \vee \neg y5) \wedge (x6 \vee \neg y6) = 1 \quad (3)$$

Проанализировав видоизмененную систему, можно сделать вывод о том, что выражения (1) и (2) однотипны и не зависят друг от друга. Поэтому сначала найдем количество наборов, являющихся решением уравнения (1). Это уравнение состоит из дизъюнкций, связанных знаками конъюнкции, и оно истинно. Из таблицы истинности операции конъюнкции известно, что она истинна, если истинны все выражения, связанные знаками логического умножения.

Следовательно, для того, чтобы найти количество наборов решений для уравнения (1), необходимо исключить ситуации, при которых выражения в скобках **ложны**. Запишем для выражения (1) все возможные наборы (цепочки) аргументов *x1, x2,..., x6*, исключив эти ситуации. Выражения в скобках будут **ложны** только тогда, когда слева от знака дизъюнкции стоит константа «**0**», а под знаком отрицания стоит константа «**1**». Эти ситуации необходимо исключить. Возможные наборы решений уравнения (1):

x1	*x2*	*x3*	*x4*	*x5*	*x6*
0	0	0	0	0	0
1	0	0	0	0	0
1	1	0	0	0	0
1	1	1	0	0	0
1	1	1	1	0	0
1	1	1	1	1	0
1	1	1	1	1	1

ГАУ АПО ИРО, РЦОИ

Все остальные наборы должны быть исключены, так как в них будет встречаться комбинация «**01**», которая даст для всего выражения значение «**0**». Следовательно, всего существует 7 различных наборов, которые являются решением уравнения (1).

Рассуждая аналогично, получим, что для решения уравнения (2) существует также 7 наборов значений переменных *y1, y2,…, y6.*

y1	*y2*	*y3*	*y4*	*y5*	*y6*
0	0	0	0	0	0
1	0	0	0	0	0
1	1	0	0	0	0
1	1	1	0	0	0
1	1	1	1	0	0
1	1	1	1	1	0
1	1	1	1	1	1

Так как выражения (1) и (2) не зависят друг от друга, то каждому набору решений для выражения (1) соответствуют все возможные наборы решений для выражения (2). Поэтому, систему уравнений (1) и (2) решают 7·7=49 наборов решений.

Осталось только выяснить, сколько наборов из этих 49 не являются решением выражения (3), входящего в систему. Для того чтобы исключить эти наборы, составим таблицу следующего вида:

x1x2x3x4x5x6	*y1y2y3y4y5y6*						
	000000	100000	110000	111000	111100	111110	111111
000000	+	+	+	+	+	+	+
100000	+	+	+	+	+	+	+
110000	+	+	+	+	+	+	+
111000	+	+	+	+	+	+	+
111100	+	+	+	+	+	+	+
111110	+	+	+	+	+	+	+
111111	+	+	+	+	+	+	+

Значения «+» в ячейках этой таблицы соответствуют всем возможным комбинациям решений уравнений (1) и (2).

Будем анализировать уравнение (3) и заменять «+» на «–» в тех клетках таблицы, где набор не является решением этого уравнения.

Проанализируем первую скобку в уравнении (3) ($x1 \vee \neg y1$). Эта скобка будет иметь значение «**истина**» на всех возможных наборах, кроме набора «**01**». Значит, в таблице необходимо поставить «–» на пересечение тех наборов, в которых $x1=0$, а $y1=1$. Таблица примет следующий вид:

x1x2x3x4x5x6	*y1y2y3y4y5y6*						
	000000	100000	110000	111000	111100	111110	111111

ГАУ ДПО ИРО РЦОИ

000000	+	–	–	–	–	–	–
100000	+	+	+	+	+	+	+
110000	+	+	+	+	+	+	+
111000	+	+	+	+	+	+	+
111100	+	+	+	+	+	+	+
111110	+	+	+	+	+	+	+
111111	+	+	+	+	+	+	+

Аналогично анализируем все остальные скобки и исправляем «+» на «–» та, где это необходимо. В конце рассуждений таблица выглядит следующим образом:

x1x2x3x4x5x6	*y1y2y3y4y5y6*						
	000000	100000	110000	111000	111100	111110	111111
000000	+	–	–	–	–	–	–
100000	+	+	–	–	–	–	–
110000	+	+	+	–	–	–	–
111000	+	+	+	+	–	–	–
111100	+	+	+	+	+	–	–
111110	+	+	+	+	+	+	–
111111	+	+	+	+	+	+	+

Осталось только подсчитать допустимые наборы. Их всего 28.

Ответ: 28.

Проблема 1. Ученики не умеют выполнять преобразования логических выражений. Не могут прогнозировать результат, который может получиться в результате выполненных преобразований.

Решение. Здесь опять же требуется какую-то часть практических занятий на уроках информатики посвятить развитию соответствующих навыков, чтобы ученики выполняли действия с логическими операциями с такой де легкостью, как и с арифметическими операциями. Вспомните, сколько часов на уроках математики посвящается определению порядка действий в выражении, раскрытию скобок, упрощению выражений.

При развитии навыков выполнения логических преобразований необходимо сначала выполнять действия, аналогичные арифметическим: то есть научиться применять законы коммутативности, ассоциативности, дистрибутивности, идемпотентности, научиться выполнять действия с константами, а затем расширить этот список до остальных логических загонов и представления одних логических операций через другие.

Проблема 2. При использовании любого метода решения данной задачи возникает необходимость анализировать исходную ситуацию с целью выбора метода решения, анализировать полученные результаты, делать вывод. Выполнение же этих действий, как уже отмечалось, вызывает у учеников затруднения.

ГАУ АПО ИРО, РЦОИ

Решение. Зачастую использование рабочих тетрадей приводит к механическому запоминанию отдельных действий, но ученики теряют навыки анализа и рассуждения. Необходимо решать большое количество задач, в которых ученикам необходимо анализировать предложенную ситуацию, рассуждать, делать выводы.

Задание 24.

На обработку поступает натуральное число, не превышающее 10^9. Надо написать программу, которая выводит на экран количество цифр числа, делящихся на 3. Если в числе нет цифр, делящихся на 3, требуется на экран вывести «NO». Программист написал программу неправильно. Ниже эта программа для Вашего удобства приведена на пяти языках программирования (мы рассмотрим только вариант на языке Pascal).

```
var N,digit,count: longint;
begin
   readln(N);
   count:=0;
   while N > 0 do
   begin
      digit := N mod 10;
      if digit mod 3 = 0 then
         count:=count+digit;
      N := N div 10;
   end;
   if count = digit then
      writeln(count)
   else
      writeln('NO')
end.
```

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе числа 125.

2. Приведите пример числа N, которое является степенью числа 5, и при вводе которого программа будет выводить правильный результат.

3. Найдите в программе все ошибки (их может быть одна или несколько). Для каждой ошибки выпишите строку, в которой она допущена, и приведите эту же строку в исправленном виде.

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

1) выпишите строку, в которой сделана ошибка;

2) укажите, как исправить ошибку, т.е. приведите правильный вариант строки.

Известно, что в тексте программы можно исправить ровно две строки так, чтобы она стала работать правильно.

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

ГАУ АПО ИРО, РЦОИ

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

Решение. Введение:

При решении задачи 24 подразумевается, что выпускник умеет решать следующие алгоритмические задачи, перечисленные в кодификаторе:

1. Запись натурального числа в позиционной системе с основанием, меньшим или равным 10. Обработка и преобразование такой записи числа.
2. Нахождение сумм, произведений элементов данной конечной числовой последовательности (или массива).
3. Использование цикла для решения простых переборных задач (поиск наименьшего простого делителя данного натурального числа, проверка числа на простоту и т.д.).
4. Нахождение минимального (максимального) значения в данном массиве и количества элементов, равных ему, за однократный просмотр массива.
5. Операции с элементами массива, отобранными по некоторому условию (например, нахождение минимального четного элемента в массиве, нахождение количества и суммы всех четных элементов в массиве).
6. Заполнение элементов одномерного и двумерного массивов по заданным правилам.

Для каждого такого умения ученик должен знать, что и как необходимо сделать для написания правильного программного кода.

Задание 1

1. Выполним алгоритм, заполнив таблицу (в таблицу включаем максимальное количество полей, чтобы потом можно было легко проверить правильность решения). Названия полей вводим последовательно, исходя из алгоритма. До цикла в приведенной программе задано два действия. Считывается значение, введенное пользователем в переменную N (readln(N)), значит в таблицу в поле N в строку 1 записываем 947. Второе действие count:=0 записывает в переменную count значение 0, что тоже отражаем в таблице:

№ строки	N	count	N>0	digit	digit mod 3	if digit mod 3 = 0	N div 10
1.	947	0					

2. Действия до цикла отчеркнем жирной линией. Продолжим ниже нее. Условие N > 0 выполняется, поэтому ставим в таблице + и выполняем действия цикла, заключенные между операторными скобками begin и end. Вычисляем N mod 10, получаем 947 mod 10 = 7 и записываем результат в переменную digit. Вычисляем digit mod 3, получаем 1, результат записываем в соответствующее поле и сравниваем полученное значение с 0. Так как значения не равны, в соответствующем поле ставим «–» и действие count:=count+digit не выполняем, соответственно, во второй строке поле count остается пустым. Выполняем следующее действие цикла N div 10, получаем 94 и результат записываем в поле N div 10, а затем, так как результат вычисления

ГАУ АПО ИРО, РЦОИ

присваивается переменной N, записываем в поле N. Действия цикла закончились. Ниже приведен результат заполнения второй строки таблицы:

№ строки	N	count	N>0	digit	digit mod 3	if digit mod 3 = 0	N div 10
1.	947	0					
2.	94		+	7	1	–	

3. Заполнение третьей строки аналогично второй, поэтому сразу приведем результат заполнения таблицы:

№ строки	N	count	N>0	digit	digit mod 3	if digit mod 3 = 0	N div 10
1.	947	0					
2.	94		+	7	1	–	94
3.	9		+	4	1	–	9

4. При заполнении четвертой строки, при вычислении digit mod 3 (9 mod 3), получаем 0, результат записываем в соответствующее поле и далее при проверке условия if digit mod 3 = 0 получаем положительный ответ и в соответствующем поле ставим «+». Теперь действие count:=count+digit необходимо выполнить, соответственно, в четвертой строке в поле count записываем результат выполнения операции count+digit: 0 + 9 = 9. Ниже результат заполнения таблицы:

№ строки	N	count	N>0	digit	digit mod 3	if digit mod 3 = 0	N div 10
1.	947	0					
2.	94		+	7	1	–	94
3.	9		+	4	1	–	9
4.	0	9	+	9	0	+	0

5. Условие цикла N > 0 теперь ложно, следовательно, действия цикла больше не выполняются. После цикла необходимо только проверить выполнение условия if count = digit, смотрим на последнюю строку таблицы, где сохранены последние значения переменных count и digit. Они равны, соответственно, 9 и 9. Следовательно, условие выполняется, и мы должны, согласно приведенной программе, вывести значение переменной count, то есть 9.

Ответ на задание 1: 9.

Задание 3

1. В момент заполнения таблицы из задания 1 смотрим, какие действия противоречат стандартным действиям, выполняемым при реализации указанного алгоритма. В следующей таблице покажем, как это делать. Сначала будем выделять из условия задачи проверяемое умение, затем перечислять, какие пункты должны быть реализованы, для подтверждения этого умения, оценивать правильность написания кода по соответствующему пункту и при необходимости исправлять ошибки.

ГАУ АПО ИРО, РЦОИ

	Необходимое действие	Код, используемый в программе	Правильность	Исправление (при необходимости)
Проверяется умение подсчитывать количество элементов, удовлетворяющих определенному условию				
1.	Инициализация счетчика	`count:=0;`	Верно	
2.	Наличие условия, при выполнении которого увеличивается значение счетчика	`if digit mod 3 = 0 then`	Верно	
3.	Увеличение значения счетчика	`count:=count+digit;`	Не верно	`count:=count+1;`
4.	Проверка полученного результата	`if count = digit then`	Не верно	`if count >0 then`
Проверяется умение записывать натуральное числа в позиционной системе с основанием, меньшим или равным 10, обрабатывать и преобразовывать такие записи числа				
5.	Выделение цифры числа	`digit := N mod 10;`	Верно	
6.	Удаление проверенной цифры из числа	`N := N div 10;`	Верно	
7.	Проверка, все ли цифры числа были проверены	`while N > 0 do`	Верно	
8.	Проверка, кратна ли цифра числа трем	`if digit mod 3 = 0 then`	Верно	

ГАУ АПО ИРО, РЦОИ

2. Заметим, что если ученик точно знает, что и как должно быть реализовано при подтверждении каждого умения, перечисленного во введении к решению этой задачи, тогда ему не составит труда найти допущенные ошибки. Из построенной таблицы мы четко видим, что было допущено две ошибки и как их исправить.

Оформим ответ на это задание, согласно требованиям, приведенным в условии задачи: 1) выпишем ошибочную строку; 2)

Ошибка 1: 1) ошибка: count:=count+digit;
2) исправление: count:=count+1;

Ошибка 2: 1) ошибка: if count = digit then
2) исправление: if count > 0 then

Слова «ошибка» и «исправление» можно не писать. Также в задаче не требуется описывать, почему строка является ошибочной, поэтому приведенного оформления ответа будет достаточно.

Можно ответ оформить также в виде таблицы:

№ ошибки	Строка с ошибкой	Исправление
Ошибка 1	`count:=count + digit;`	`count:=count+1;`
Ошибка 2	`if count = digit then`	`if count >0 then`

Задание 2

Решение этого задания намеренно вынесено в самый конец, так как оно является наиболее сложным для понимания и для его решения необходимо: 1) знать допущенные в программе ошибки; 2) уметь анализировать полученную информацию и делать необходимые выводы.

1. Мы знаем, найдя первую допущенную ошибку, что к `count` добавляется не 1 при выполнении условия, а проверяемая цифра: то есть там хранится сумма цифр, кратных 3. То есть мы должны подобрать пример, в котором сумма цифр, кратных 3 совпадает с их количеством.

2. Из второй ошибки, мы знаем, что найденное значение должно совпадать с первой цифрой числа, чтобы вывелось именно количество, а не «No» (а у нас должно вывестись количество, так как по условию задания 3 в приводимом примере должна быть хотя бы одна цифра кратная 3). Следовательно, должно выполняться равенство: 1) количества цифр кратных трем; 2) суммы цифр, кратных трем; 3) первой цифры числа. И все это одновременно.

3. При этом число должно быть трехзначным. То есть количество цифр, кратных трем, не должно быть больше трех. Соответственно, и сумма и первая цифра не должны быть больше трех. Значит в состав числа могут входить только две цифры, кратные трем: 0 и 3 или цифры, не кратные трем.

4. Подберем число, удовлетворяющее всем перечисленным условиям. Таким числом будет 300.

5. Последним шагом необходимо убедиться в правильности подобранного примера. Для этого заполним такую же таблицу, как и при выполнении задания 1. Приведем результат заполнения таблицы:

№ строки	N	count	N>0	digit	digit mod 3	if digit mod 3 = 0	N div 10
1	300	0					
2	30	0	+	0	0	+	30
3	3	0	+	0	0	+	3
4	0	3	+	3	0	+	0

6. Хотя по ходу заполнения таблицы значение count принимало неверные значения в конце оно стало равно 3, что является верным значением. Также найденное значение count совпадает со значением переменной digit, поэтому программа выведет 3. Следовательно, при вводе числа 300 приведенная программа выдает правильный ответ.

Ответ на задание 2: 300.

ГАУ ДПО ИРО, РЦОИ

Проблемы, с которыми сталкиваются ученики при решении этой задачи.

Проблема 1. Ученики путаются в написании программы, решающей ту или иную алгоритмическую задачу, описанную во введении к решению.

Решение. Как уже говорилось ранее, для устранения данной проблемы необходимо последовательно подходить к обучению учеников. Например, в кодификаторе сказано, что ученики должны уметь «выполнять операции с элементами массива, отобранными по некоторому условию (находить минимальный четный элемент в массиве, находить количество и сумму всех четных элементов в массиве».

Рассмотрим последовательное обучение данному умению на примере нахождения суммы элементов, удовлетворяющих одному или нескольким условиям.

1. Ученики должны понимать, что такое массив, знать различные способы его заполнения, уметь отличать фрагменты программного кода, отвечающие за заполнение, обработку и вывод массива. То есть очевидно, что для начала ученики должны научиться понимать, что, например, следующая программа заполняет массив, состоящий из 5 элементов, значениями, введенными пользователем.

```
Program Hello;
const N=5;
var
   a: array [1..N] of longint;
   i, j, k: longint;
begin
   for i := 1 to N do
         readln(a[i]);
end.
```

2. Дальше добавляем к этой программе код, отвечающий за подсчет суммы элементов:

```
Program HelloWorld;
const N=5;
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 do
            k:=k+a[i];
      writeln(k);
end.
```

Акцентируем внимание на том, что должны быть реализованы и как четыре пункта: 1) должна быть переменная, в которой будет накапливаться сумма; 2) должна быть инициализация начального значения этой переменной; 3) должен осуществляться перебор всех суммируемых элементов; 4) должно быть увеличение значения этой переменной.

ГАУ АПО ИРО, РЦОИ

3. Следующим шагом добавляем условие, например, проверки числа на четность (далее будем приводить только код, отвечающий за нахождение суммы элементов, удовлетворяющих одному или нескольким условиям)

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

Акцентируем внимание на том, где должно добавляться условие и какие условия могут быть. Тренируемся решать подобные задачи с разными условиями. Например:

Измените программу таким образом, чтобы выводилась

1) сумма нечетных элементов массива;
2) сумма элементов массива, кратных трем;
3) сумма элементов массива, не делящихся нацело на 5.

4. Теперь, учимся работать с двумя условиями. Объясняем, например, на задании «найти сумму элементов кратных 5 и не превышающих 10»

```
k:=0;
   for i := 1 to N do
         if (a[i] mod 5 = 0) and (a[i]<=10) then
               k:=k+1;
   writeln(k);
```

5. Объясняем, что логические связки могут быть разными. Приводим пример с «или» и «не». Также даем отработать данную тему на различных заданиях. Например:

Измените программу таким образом, чтобы выводилась

1) сумма четных элементов, больших 5;
2) сумма элементов, не превышающих 3 и заканчивающихся на 2.

6. Рассматриваем разницу в задачах, когда «точно известно, что есть элементы, удовлетворяющие заданному условию», и когда «не гарантируется, что есть хотя бы один элемент, удовлетворяющий заданному условию».

Важно. Заметим, что лучше не проходить все шаги со 2-ого по 6-ой в глубину для одной только суммы, а параллельно на каждом этапе рассмотреть те же действия для подсчета количества элементов, нахождения минимального и максимального (причем проходить материал в паре: сначала рассматривать задачи на нахождение количества и суммы, а потом задачи на нахождение минимума и максимума). При усвоении всех четырех базовых алгоритмов на первом уровне, переходим к добавлению условия, потом двух и так далее.

Проблема 2. Ученикам сложно анализировать алгоритмическую составляющую программы в поисках допущенных ошибок, и они начинают искать синтаксические и подобные им ошибки. Как правило, введенные учениками изменения, не добавляют ошибок, но исправляют верный код. А за то, что ученик не может отличить правильный код от неправильного, с него снимаются баллы.

ГАУ ДПО ИРО, РЦОИ

Решение. Необходимо акцентировать внимание, что в задаче 24 требуется искать алгоритмические ошибки по типу тех, которые описаны в примере решения.

Перечислим наиболее частые ошибки, которые допускают ученики при поиске и исправлении ошибок в программе:

1) изменение типа переменных;
2) добавление или удаление «;»;
3) замена вложенных условий на сложные и наоборот;
4) изменение имени переменных;
5) замена write на writeln, read на readln и наоборот.

Необходимо объяснить разницу между алгоритмическими и приведенными в последнем списке ошибками, чтобы ученики не тратили время на бесполезную работу и не теряли баллы из-за неверных исправлений.

Проблема 3. Ученики не умеют анализировать, обобщать информацию и делать выводы, что необходимо для правильного выполнения задания 2.

Решение. Необходимо рассматривать различные задачи, реализующие указанные во введении алгоритмы. Проговаривать, для каких входных данных, каким должен быть ответ, анализировать возможные исключения.

Задание 25.

Дан целочисленный массив из 30 элементов. Элементы массива могут принимать натуральные значения от 1 до 10 000 включительно. Опишите на одном из языков программирования алгоритм, который сумму элементов массива, не больших 197 и при этом имеющих нечетное значение, а затем заменяет каждый такой элемент на число, равное найденной сумме. Гарантируется, что хотя бы один такой элемент в массиве есть. В качестве результата необходимо вывести изменённый массив, каждый элемент выводится с новой строчки.

Решение. Решение данной задачи не представляет никакой алгоритмической сложности, поэтому мы его здесь приводить не будем. Остановимся только на проблемах, с которыми встречаются ученики при ее выполнении.

Заметим также, что в основном ученики, которые берутся за решение данной задачи, справляются с ней полностью или частично. Отсюда вытекает первая проблема, которая аналогична описанной в предыдущей задаче.

Проблема 1. Ученики не умеют записывать алгоритм, решающий ту или иную задачу, проверяемую в данном задании.

Решение. Процесс решения данной проблемы подробно описан в решении проблемы 1 задачи 24.

Проблема 2. Много ошибок допускается по невнимательности, из-за чего ученики теряют 1 или 2 балла. Наиболее распространенными являются ошибки типа:

1) забыли задать цикл (кто-то считает очевидным, что его надо задавать и не пишет соответствующую строчку, кто-то не понимает, что цикл, в котором происходит заполнение массива, уже закончился, и надо начать новый, кто-то просто забывает написать);

ГАУ АПО ИРО, РЦОИ

2) не инициализируют или неверно инициализируют переменную;
3) перепутали div и mod;
4) перепутали строгое и нестрогое неравенство;
5) перепутали логическую операцию в условии;
6) не выводят ответ или выводят его неправильно.

Решение. В процессе подготовки к экзамену необходимо обратить внимание учеников на критерии оценивания, приведенные в демонстрационном варианте КИМ ЕГЭ, в котором указываются ошибки, за допущение которых оценка снижается на 1 балл. Если ученики выучат эти ошибки и будут проверять свое решение на их наличие, тогда, очевидно, снизится количество ошибок, допущенных по невнимательности.

Ну и при той формулировке задачи, как она дается последние два года, когда необходимо изменить массив, а затем вывести его, необходимо обращать внимание, что вывод нужных значений без изменения массива не является верным решением задачи, хотя строка вывода выглядит верно. То есть необходимо выполнить все пункты задания: найти нужное значение, изменить массив, вывести массив.

Задание 26.

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в одну из куч (по своему выбору) один или четыре камня или увеличить количество камней в куче в три раза. Для того чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не менее 61. Победителем считается игрок, сделавший последний ход, т.е. первым получивший такую позицию, при которой в кучах будет 61 или больше камней.

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. В описание выигрышной стратегии не следует включать ходы играющего по этой стратегии игрока, не являющиеся для него безусловно выигрышными, т.е. не являющиеся выигрышными независимо от игры противника.

Выполните следующие задания.

Задание 1. Для каждой из начальных позиций (3,2 7), (8, 26) укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию.

Задание 2. Для каждой из начальных позиций (6, 26), (7, 26), (8, 25) укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию.

Задание 3. Для позиции (7,25) укажите, кто из игроков имеет выигрышную стратегию. Опишите выигрышную стратегию.

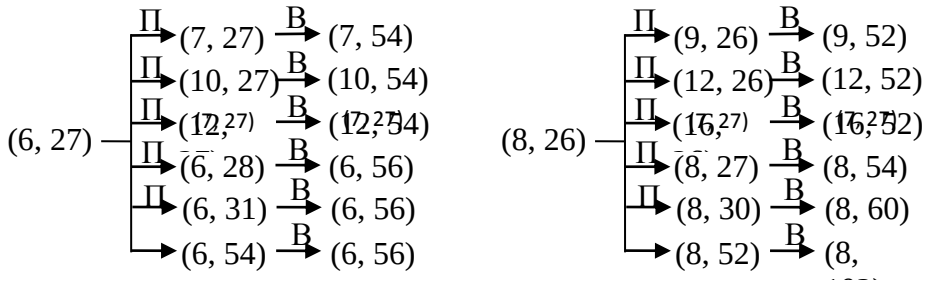
Постройте дерево всех партий, возможных при указанной Вами выигрышной стратегии (в виде рисунка или таблицы). В узлах дерева

ГАУ АПО ИРО, РЦОИ

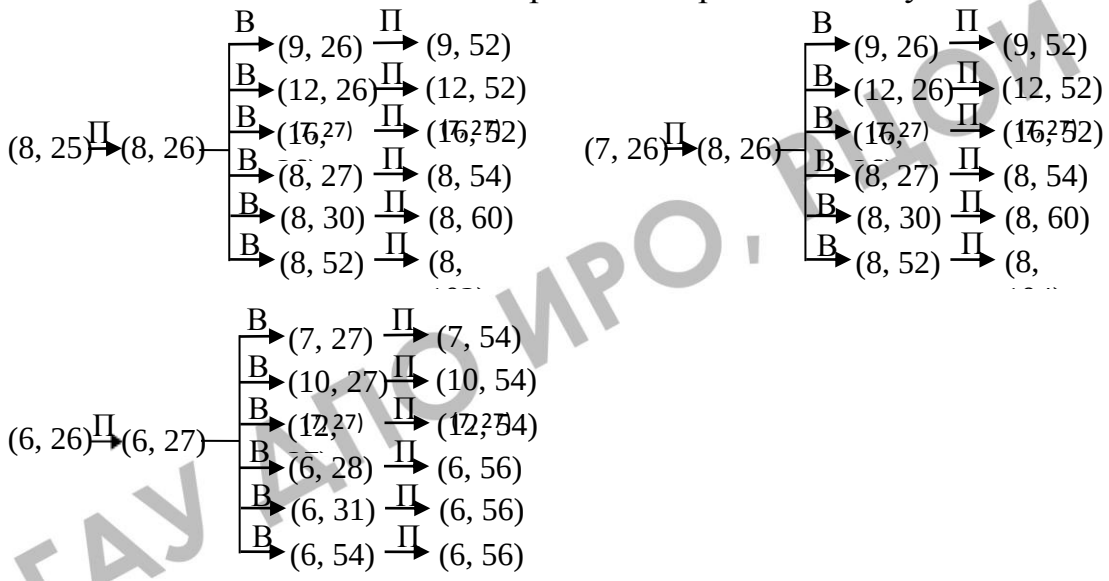
указывайте позиции, на рёбрах рекомендуется указывать ходы. Дерево не должно содержать партии, невозможные при реализации выигрывающим игроком своей выигрышной стратегии. Например, полное дерево игры не является верным ответом на это задание.

Решение. Приведем самую короткую запись решения, так как зачастую ошибки возникают не вычислении правильных значений, а в записи стратегии.

Задание 1. В обеих позициях выигрышная стратегия есть у Вани.



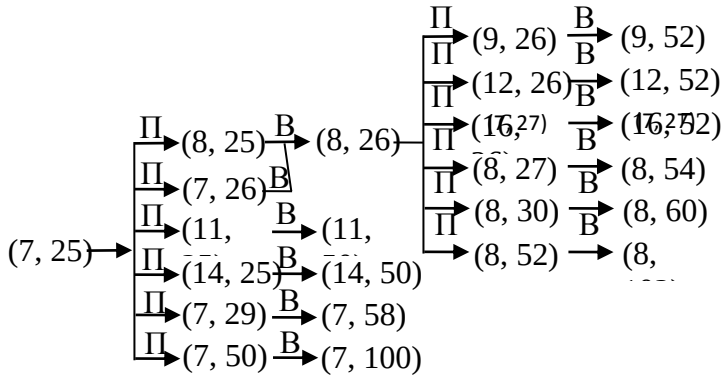
Задание 2. Во всех позициях выигрышная стратегия есть у Пети.



ГАУ ДПО ИРО РЦОИ

Можно еще сократить запись, объединив деревья из позиций (8, 25) и (7, 26), так как, начиная со второго хода, они полностью идентичны.

Задание 3. Выигрышная стратегия есть у Вани.



Теперь рассмотрим, с какими проблемами сталкиваются ученики при выполнении данного задания.

Проблема 1. Неполная запись ответа. Например, при решении данной задачи неполный ответ мог быть следующего вида:

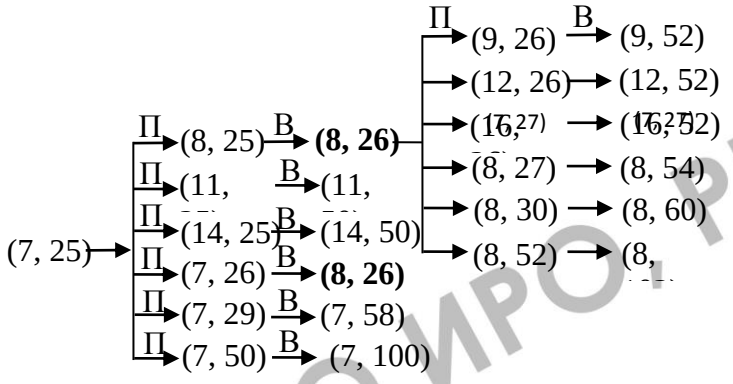
1) в заданиях 1 и 2 приводилась стратегия только для одной из позиций (ученики считают, что, если стратегия для второй позиции выглядит почти также, как и для первой, ее не надо записывать);

2) ученики не указывали ходы «умножение большей кучи», считая, что в этом случае игрок, сделавший такой ход, обязательно проиграет;

3) привыкнув работать только с одной кучей, ученики в описании стратегии рассматривали варианты увеличения только одной кучи;

4) в задании 3 при построении дерева ученики ссылаются на поддеревья, описанные в предыдущих заданиях, не заканчивая построение;

5) в задании 3 при построении дерева, если промежуточный результат содержит одинаковые позиции, ученики продолжают построение дерева только для одной позиции. Например, как показано на рисунке ниже:



Решение. Когда ученики говорят, что они опускают что-то в записи, считая это очевидным, надо дать понять, что они готовятся при решении таких задач научить выигрывать в такой игре, например, робота, для которого нет понятия очевидного. Робот должен точно знать, в какой ситуации, как он должен сходить, поэтому стратегия должна быть записана полностью и для всех позиций.

Проблема 2. Ученики не всегда понимают, что значит, описать стратегию, несмотря на то, что в самой формулировке задания дается определение, что значит описать стратегию.

Решение. Необходимо на уроках, посвященных изучению стратегии, при решении задач, давать ученикам простые стратегические игры и учить их правильно играть, чтобы они прежде чем сделать ход рассуждали: «если противник сделает такой ход 1, тогда я сделаю ход 11, а если противник сделает ход 2, я сделаю ход 21 и т.д.» Играя, они начинают понимать, что для выигрыша они для себя выбирают один наилучший ход, но, чтобы убедиться, что они действительно выиграют, они рассматривают всевозможные ходы, которые может сделать противник, чтобы не допустить их выигрыш.

Проблема 3. Неверное построение дерева: либо рассматриваются не все ходы проигрывающего игрока, либо за выигрывающего игрока, наоборот, строятся ветки, не приводящие его к победе.

ГАУ ДПО ИРО РЦОИ

Решение. Эта проблема является частным случаем первой и второй проблемы. Мы выделили ее в отдельную проблему, так как она достаточно часто встречается и лишает учеников 2-х первичных баллов. Решение же этой проблемы описано выше.

Задание 27. На вход программы поступает последовательность из n целых положительных чисел. Рассматриваются все пары элементов последовательности a_i и a_j, такие что $i < j$ и $a_i > a_j$ (первый элемент пары больше второго; i и j – порядковые номера чисел в последовательности входных данных). Среди пар, удовлетворяющих этому условию, необходимо найти и напечатать пару с максимальной суммой элементов, которая делится на m = 114. Если среди найденных пар максимальную сумму имеют несколько, то можно напечатать любую из них.

Описание входных и выходных данных

В первой строке входных данных задаётся количество чисел n ($2 \le n \le 12\ 000$). В каждой из последующих n строк записано одно целое положительное число, не превышающее 10 000.

В качестве результата программа должна напечатать элементы искомой пары. Если таких пар несколько, можно вывести любую из них. Гарантируется, что хотя бы одна такая пара в последовательности есть.

Требуется написать эффективную по времени и памяти программу для решения описанной задачи.

Программа считается эффективной по времени, если при одновременном увеличении количества элементов последовательности n и параметра m в k раз время работы программы увеличивается не более чем в k раз.

Программа считается эффективной по памяти, если память, необходимая для хранения всех переменных программы, не превышает 4 килобайта и не увеличивается с ростом n.

Максимальная оценка за правильную (не содержащую синтаксических ошибок и дающую правильный ответ при любых допустимых входных данных) программу, эффективную по времени и памяти, – 4 балла.

Максимальная оценка за правильную программу, возможно, неэффективную по памяти или время выполнения которой существенно зависит от величины m, – 3 балла.

Максимальная оценка за правильную программу, не удовлетворяющую требованиям эффективности, – 2 балла.

Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

Перед текстом программы обязательно кратко опишите алгоритм решения.

Укажите использованный язык программирования и его версию.

Решение. Здесь, как и в задаче 25, не будем приводить само решение, так как задачи, предлагаемые ученикам, могут быть очень разнообразны, и у учеников должен быть соответствующий уровень подготовки, позволяющий

ГАУ АПО ИРО, РЦОИ

справиться с задачами более высокого уровня. Поэтому остановимся только на тех проблемах, с которыми сталкиваются ученики при решении задачи 27.

Проблема 1. Далеко не все ученики, приступающие к написанию решения, оцениваемому из четырех баллов, понимают, что значит программа, эффективная по времени, и программа, эффективная по памяти.

Решение. Для того, чтобы ученики понимали, что значит программа, эффективная по памяти, необходимо научить их вычислять максимальный объем памяти, необходимый для хранения всех переменных программы.

Для того, чтобы ученики понимали, что значит программа, эффективная по времени, необходимо научить их вычислять максимальное количество операций, выполняемых программой.

Далее необходимо внимательно читать, что понимается под эффективностью, так как формулировки могут меняться в зависимости от конкретной задачи (например, формулировка эффективности по времени на экзамене 2019 года отличалась от предыдущих лет).

Проблема 2. Для большинства учеников возникают сложности с разработкой эффективного алгоритма, решающего поставленную задачу.

Проблема 3. При написании эффективной программы возникают сложности с выделением исключительных ситуаций, требующих дополнительной обработки в программе, или разработки более сложного алгоритма решения.

Решение. Для решения второй и третьей проблемы недостаточно дать пару рекомендаций, которые приведут сразу к видимому результату, так как для их решения должна быть проведена последовательная работа по обучению программированию и решению сложных алгоритмических задач. Поэтому дадим две самые важные, с нашей точки зрения, рекомендации:

1. Начать разработку алгоритма, решающего задачу 27, можно с примера на произвольном наборе данных. Разработав алгоритм для частного случая, необходимо продумать, как расширить его для любых допустимых входных данных.

2. Использовать на уроках информатики для обучения программированию на разных уровнях автоматизированных тестирующих систем, которые содержат базы задач, на которых ученики могут оттачивать свои навыки. Также некоторые автоматизированные тестирующие системы позволяют создавать свои базы задач, необходимые, например, для отработки конкретных навыков.

Проблема 4. У учеников есть возможность написать две программы. Например, содержащую: 1) переборный алгоритм решения, оцениваемый из 2-х баллов; 2) эффективный алгоритм решения, оцениваемый из 4-х баллов. Ученики, уверенные в своих силах, пишут только одну эффективную по памяти и времени программу, но допускают ошибку, не позволяющую оценить решение более чем в 1 балл (например, забывают закрыть операторную скобку). Хотя эти же ученики с легкостью могли бы написать переборный вариант решения и получили бы на 1 балл больше.

ГАУ АПО ИРО, РЦОИ

Решение. На этапе подготовки к экзамену акцентировать внимание учеников на том, что и они могут допускать ошибки, а запись второго решения не займет у них много времени.

Для подготовки по предмету «Информатика и ИКТ» в регионе используются следующие УМК:

УМК	Примерное количество обучающихся по данному УМК	Из них, количество сдававших экзамен по предмету	Средний балл по предмету
Информатика (базовый уровень) / Семакин И. Г., Хеннер Е. К., Шеина Т. Ю. – Издательство: Бином. Лаборатория знаний	6 151	745	57,3
Информатика (базовый уровень) / Угринович Н. Д. – Издательство: Бином. Лаборатория знаний	2 655	262	51,3
Информатика. Базовый уровень / Босова Л. Л., Босова А. Ю. – Издательство: Бином. Лаборатория знаний	1 735	148	50,7
Информатика (базовый и углубленный уровни) (в 2 частях) / Поляков К. Ю., Еремин Е. А. – Издательство: Бином. Лаборатория знаний	1 459	386	66,9

ГАУ ДПО ИРО. РЦОИ

Наиболее высокий средний балл при использовании последнего УМК связан с тем, что:

1. Он используется в основном в профильных классах, уровень подготовки в которых достаточен для освоение данной программы.
2. В данном УМК наиболее последовательно и всесторонне прорабатывается тема «Алгоритмизация и программирование», с которой у учеников возникают наибольшие проблемы.

Остальные УМК предназначены для подготовки на базовом уровне, поэтому используются в основном в общеобразовательных классах и не предусматривают достаточного практикума по разделам «Логика и программирование».

3.4. Выводы по разделу

Элементы содержания / умений и видов деятельности, усвоение которых всеми школьниками региона ***в целом можно считать достаточным***:

– формировать для логической функции таблицу истинности и логическую схему;
– искать кратчайший путь в графе, осуществлять обход графа;
– осуществлять перевод из одной системы счисления в другую;
– формулировать запросы к базам данных и поисковым системам;
– формулировать запросы к базам данных и поисковым системам;
– оценить результат работы известного программного обеспечения.

Элементы содержания / умений и видов деятельности, усвоение которых ***участниками первой и второй группы нельзя считать достаточным***:

– оперировать массивами данных;

– использовать стандартные алгоритмические конструкции при программировании;

– описывать свойства двоичной последовательности по алгоритму ее построения;

– анализировать результат исполнения алгоритма;

– анализировать текст программы с точки зрения соответствия записанного алгоритма поставленной задаче и изменять его в соответствии с заданием;

– построить дерево игры по заданному алгоритму и обосновать выигрышную стратегию.

Элементы содержания / умений и видов деятельности, усвоение которых всеми школьниками региона ***в целом нельзя считать достаточным***:

– осуществлять преобразования логических выражений;

– реализовывать сложный алгоритм с использованием современных систем программирования.

Исходя из анализа статистики выполнения отдельных заданий ЕГЭ, можно дать следующие ***рекомендации по преподаванию*** Информатики и ИКТ в регионе:

– акцентировать внимание на работу с алгоритмами, а именно:

▪ выполнение алгоритмов, выполняющих действия с числами, с объектами на координатной плоскости, со словами и т.д.;

▪ в алгоритмах должны использоваться последовательные действия, ветвление, циклические конструкции разных типов;

▪ подбор примеров для анализа алгоритмов;

▪ умение анализировать результат выполнения алгоритма (в частности умение определить, что может быть результатом работы алгоритма, а что нет);

▪ обобщение результатов работы алгоритмов;

▪ составление алгоритмов, в которых содержатся последовательные действия, ветвление, циклические конструкции разных типов;

▪ составление алгоритмов, использующих различные структуры данных;

– уделить особое внимание практике программирования;

– развивать навыки обобщения и систематизации информации, формулирования выводов;

– развивать логическое мышление школьников.

ГАУ ДПО ИРО, РЦОИ

IV. РЕКОМЕНДАЦИИ ПО ПОДГОТОВКЕ К ЕГЭ

Исходя из приведенного анализа используемых УМК в регионе и описанных в выводах рекомендаций, желательно при подготовке учеников по предмету «Информатика и ИКТ» использовать дополнительный учебный материал, расширяющий практические навыки по указанным темам в рассмотренных направлениях.

В связи с тем, что пробелы подготовки учеников по предмету «Информатика и ИКТ» нередко связаны с недостаточной подготовкой учителей информатики, можно предложить следующие темы курсов повышения квалификации:

1. Развитие логического и алгоритмического мышления школьников.
2. Использование автоматизированных тестирующих систем при обучении программированию.
3. Олимпиадное программирование (для учителей, работающих в классах с углубленным изучением информатики и ИКТ).
4. Методика подготовки учащихся к итоговой аттестации по предмету «Информатика и ИКТ».

Для обсуждения на методических объединениях учителей информатики рекомендуются следующие темы:

а) возможности организации дифференцированного обучения школьников, желающих сдавать ЕГЭ, в конкретной образовательной организации и муниципалитете на уроках и во внеурочной деятельности;

б) разбор типичных ошибок школьников, допущенных на экзамене, и методические способы их преодоления;

в) особенности работы с кодификаторами и спецификацией ЕГЭ по информатике;

г) возможности совершенствования предметной и методической компетентности учителей информатики средствами образовательной организации и/или муниципальной методической службы;

д) особенности подготовки школьников к ЕГЭ с учётом специфики конкретной образовательной организации и/или муниципального образования.

ГАУ АПО ИРО, РЦОИ

ГАУ ДПО ИРО, РЦОИ

Результаты государственной итоговой аттестации в форме единого государственного экзамена по информатике и ИКТ в Иркутской области в 2019 году

Методические рекомендации

Авторы-составители:

Наталия Леонидовна Семичева,
Светлана Юрьевна Лебедева

ГАУ ДПО ИРО, РЦОИ

Подписано в печать 19.08.2019
Формат бумаги 60×84 1/16
Объем 3,13 усл. печ. л.
Заказ 19–302. Тираж 10 экз.
Отпечатано в оперативной типографии ГАУ ДПО ИРО
664023, г. Иркутск, ул. Лыткина, 75А, оф.106
тел./факс: :8(3952)50-09-04
e-mail: info@iro38.ru