

«Мир информатики»

Журнал для тех, у кого информатика – любимый школьный предмет

Выпуск № 99, апрель 2025 г.

ГИА

О заданиях 8 демонстрационных вариантов ОГЭ по информатике. Продолжение (см. предыдущий выпуск)

4. Задания с тремя словами в запросе

4.1. Общие вопросы

Если в запросах фигурируют три слова, то круги Эйлера, соответствующие страницам с этими словами, в общем случае будут иметь вид:

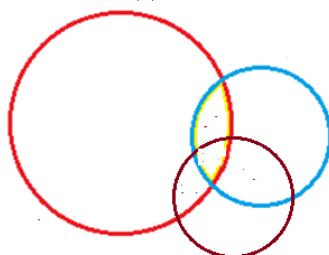


Рис. 7

Обсудим, каким множества страниц соответствуют те или иные области.

На рис. 8 круги, рядом с которыми записаны числа 1, 2 и 3, представляют множества страниц, на которых имеются слова, соответственно, А., Б. и В.

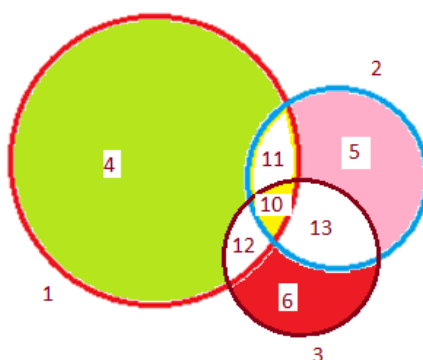


Рис. 8

Каким множествам страниц соответствуют области, закрашенные зеленым, розовым, красным и жёлтым цветами (их номера – 4, 5, 6 и 10), установите самостоятельно.

Еще шесть областей выделены на рис. 9.

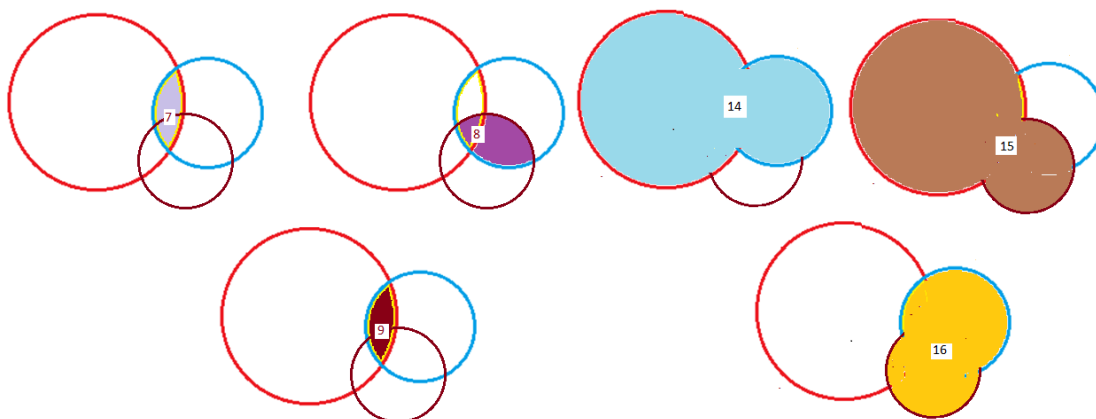


Рис. 9

Каким множествам страниц соответствует каждая из них?

Самыми сложными являются области, на рис. 7 обозначенные номерами 11, 12 и 13. Сравнив, например, область 11 с областью 7, можем сказать, что она соответствует тем страницами, на которых имеются оба слова А. и Б., но нет слова В. (убедитесь в этом!). Особенности областей 12 и 13 установите самостоятельно.

Теперь – о запросах на поиск в Интернете страниц с тремя словами. Понятно, что общее число вариантов запросов будет больше 6 (см. таблицу 1 в предыдущем выпуске нашего журнала). Все они перечислены в табл. 2.

Таблица 2

Запрос	Номера областей на рис. 8 и 9
А.	1
Б.	2
В.	3
А. & Б.	7
А. & В.	9
Б. & В.	8
А. Б.	14
А. В.	15
Б. В.	16
А. Б. В.	10
А. & Б. & В.	Вся площадь рисунков

Второй столбец табл. 2 заполните самостоятельно. Заполните также второй столбец табл. 3. До этого сформулируйте, каким страницам в Интернете соответствуют эти области.

Таблица 3

Номера областей на рис. 8 и 9	Запрос, по котором будут отобраны соответствующие страницы
7	
8	
9	
10	
14	
15	
16	

4.2. Пример задания и методика его выполнения

Обсудим задание: «В таблице приведены запросы к поисковому серверу. Для каждого запроса указан его код – соответствующая буква от А до Г. Расположите коды запросов слева направо в порядке убывания количества страниц, которые нашёл поисковый сервер по каждому запросу. По всем запросам было найдено разное количество страниц.

Для обозначения логической операции ИЛИ в запросе используется символ «|», а для логической операции И – символ «&».

Код	Запрос
А	Блин & Мёд & Масленица
Б	Блин Мёд
В	Блин & Мёд
Г	Блин Мёд Масленица

Решение

Если условно принять, что слова «Блин», «Мёд» и «Масленица» – это слова А., Б. и В., множества страниц которых изображены на рис. 8 и 9, то соответствующие им области будут такими, как в новом столбце таблицы:

Код	Запрос	Номер области на рис. 8 и 9
А	Блин & Мёд & Масленица	10
Б	Блин Мёд	14
В	Блин & Мёд	7
Г	Блин Мёд Масленица	Вся площадь рисунков

Проанализировав соответствующие области, можем записать результат сравнения областей: $10 < 7 < 14 < \text{Вся площадь}$. Значит коды соответствующих запросов в порядке убывания: ГБВА.

Ответ: ГБВА.

4.3. Задания для самостоятельной работы

Во всех предложенных ниже заданиях:

- 1) в таблице приведены запросы к поисковому серверу. Для каждого запроса указан его код – соответствующая буква от А до Г. Расположите коды запросов слева направо;
 - в порядке возрастания количества страниц, которые нашёл поисковый сервер по каждому запросу;
 - в порядке убывания количества страниц, которые нашёл поисковый сервер по каждому запросу;

По всем запросам было найдено разное количество страниц;

- 2) в запросах использованы указанные ранее знаки логических операций.

8.

Код	Запрос
А	Мартышка & Очки & Басня
Б	Мартышка & Очки
В	Мартышка Очки Басня
Г	Мартышка Очки

9.

Код	Запрос
А	Красавица & Чудовище & Сказка
Б	Красавица Чудовище Сказка
В	Красавица & Чудовище

Г	Красавица Чудовище
---	----------------------

10.

Код	Запрос
А	Царевна Лягушка Сказка
Б	Царевна & Лягушка
В	Царевна & Лягушка & Сказка
Г	Царевна Лягушка

Можем сделать вывод о том, что при отборе страниц тремя словами по запросу с двумя логическими операциями И будет отобрано минимальное количество страниц, по запросу с двумя логическими операциями ИЛИ – максимальное.

5. Задания с тремя словами и скобками в запросе

5.1. Примеры заданий и методика их выполнения

Еще одно задание аналогичного типа: «Расположите коды запросов, приведённых в таблице, слева направо в порядке возрастания количества страниц, которые нашёл поисковый сервер по каждому запросу».

Код	Запрос
А	(Блин & Мёд) Масленица
Б	Блин Мёд Масленица
В	Блин & Мёд & Масленица
Г	Блин & Мёд

Решение

Анализ таблицы показывает, что в ней появился запрос с кодом А нового вида – со скобками. Если, как и ранее, условно принять, что слова «Блин», «Мёд» и «Масленица» – это слова А., Б. и В., множества страниц которых изображены на рис. 8 и 9, то какая область соответствует этому запросу? Для части запроса в скобках это область 7 на рис. 9. Логическая операция ИЛИ «добавит» к этой области все страницы со словом В. («Масленица»). Вся полученная в итоге область на рис. 10 закрашена.

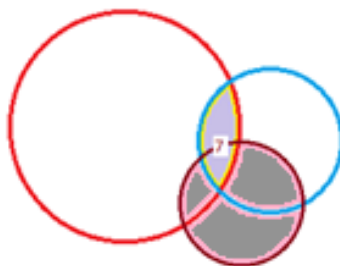


Рис. 10

Видно, что эта область больше области 7 (соответствует запросу «Блин & Мёд»).

Другая известная информация записана в последнем столбце таблицы:

Код	Области
А	Закрашена на рис. 10
Б	10
В	Вся площадь рисунков
Г	7

Так как мы знаем, что (область 7) > (области 10), то можем записать коды запросов в требуемом в условии порядке возрастания: БГАВ.

Ещё задание со скобками в запросе: «Расположите коды запросов, приведённых в таблице, слева направо в порядке *возрастания* количества страниц, которые нашёл поисковый сервер по каждому запросу».

Код	Запрос
А	Солнце Воздух Вода
Б	Солнце & Воздух & Вода
В	(Солнце Воздух) & Вода
Г	Солнце Воздух

Решение

Здесь новым является запрос В.

Какие страницы будут отобраны по запросу в скобках («Солнце | Воздух»)? Какая область им соответствует? – Это закрашенная область на рис. 11.

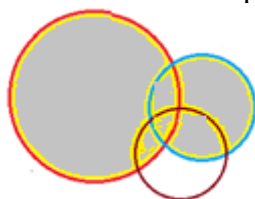


Рис. 11

А по запросу «Вода»?

Как объединить «закрашенную область» и «нижний круг»? Это будут только те страницы из «нижнего круга», которые входят также в «закрашенную область». На рис. 12 эта часть выделена темно-серым цветом:



Рис. 12

Можем заполнить таблицу:

Код	
А	минимум
Б	максимум
В	Закрашенная область на рис. 12
Г	Закрашенная область область на рис. 11

Сравнение размеров областей показывает, что коды запросов в порядке возрастания количества страниц: АВГБ.

5.2. Задания для самостоятельной работы

Во всех предложенных ниже заданиях:

- 1) в таблице приведены запросы к поисковому серверу. Для каждого запроса указан его код – соответствующая буква от А до Г. Расположите коды запросов слева направо;
 - в порядке возрастания количества страниц, которые нашёл поисковый сервер по каждому запросу;
 - в порядке убывания количества страниц, которые нашёл поисковый сервер по каждому запросу;

По всем запросам было найдено разное количество страниц;
2) в запросах использованы указанные ранее знаки логических операций.

11.

Код	Запрос
А	Контрольная & Зачёт & Экзамен
Б	(Контрольная & Зачёт) Экзамен
В	Контрольная Зачёт Экзамен
Г	Контрольная & Зачёт

12.

Код	Запрос
А	Ворона Лисица Сыр
Б	Ворона & Лисица & Сыр
В	Ворона & Лисица
Г	(Ворона Лисица) & Сыр

13.

Код	Запрос
А	Эльфы Гномы Орки
Б	Эльфы & Гномы & Орки
В	(Эльфы Гномы) & Орки
Г	Эльфы Гномы

5.3. Вопросы для закрепления

- 1) в каком случае будет отображено большее количество страниц – по запросу «Б. & А.» или «А. & Б. & В.»?
- 2) в каком случае будет отображено меньшее количество страниц – по запросу «А. | Б.» или «В. | Б. | А.»?
- 3) в каком случае будет отображено большее количество страниц – по запросу «А. & Б.» или «(А. & Б.) | В.»?
- 4) в каком случае будет отображено большее количество страниц – по запросу «А. | Б.» или «(А. | Б.) & В.»?

6. Задания с четырьмя словами в запросе

1. Примеры заданий и методика их выполнения

Пример 1

В таблице приведены запросы к поисковому серверу. Для каждого запроса указан его код – соответствующая буква от А до Г. Расположите коды запросов слева направо в порядке убывания количества страниц, которые нашёл поисковый сервер по каждому запросу. По всем запросам было найдено разное количество страниц.

Для обозначения логической операции ИЛИ в запросе используется символ «|», а для логической операции И – «&».

Код	Запрос
А	Мартышка Осёл Козёл
Б	Мартышка & Осёл & Козёл & Мишка
В	Мартышка & Осёл & Козёл
Г	(Мартышка & Осёл) Козёл

Решение

Сначала следует определить запросы, по которым будет отобрано самое большое количество страниц (это запрос А с тремя операциями ИЛИ) и самое малое количество (это запрос Б с четырьмя операциями И) и записать их коды в требуемом порядке:

А ? ? Б

Запрос В (с тремя операциями И) даст большее количество страниц, чем запрос Б:

А ? В ? Б

Осталось определить «место запроса Г. В нем есть операция ИЛИ (увеличивающая количество отбираемых страниц), то есть по этому запросу будет отобрано второе по величине максимальное количество:

А Г В Б

Ответ: АГВБ.

Пример 2

В таблице приведены запросы к поисковому серверу. Для каждого запроса указан его код – соответствующая буква от А до Г. Расположите коды запросов слева направо в порядке возрастания количества страниц, которые нашёл поисковый сервер по каждому запросу. По всем запросам было найдено разное количество страниц.

Код	Запрос
А	ПирогИ & Пышки & Бублики & Баранки
Б	Пышки & Бублики & Баранки
В	ПирогИ Пышки Бублики Баранки
Г	ПирогИ (Пышки & Бублики & Баранки)

Решение

Учитывая ответы на четыре вопроса, заданные перед данным пунктом, можно заполнить таблицу:

Код	Кол-во страниц
А	Минимальное
Б	Больше, чем по запросу А
В	Максимальное
Г	Больше, чем по запросу Б

Ответ: АБГВ.

И ещё один, самый сложный пример.

Пример 3

В таблице приведены запросы к поисковому серверу. Для каждого запроса указан его код – соответствующая буква от А до Г. Расположите коды запросов слева направо в порядке возрастания количества страниц, которые нашёл поисковый сервер по каждому запросу. По всем запросам было найдено разное количество страниц.

Код	Запрос
А	Мороз & Солнце & День & Чудесный
Б	Мороз Солнце День Чудесный
В	Мороз & Солнце & Чудесный
Г	Чудесный (Мороз & Солнце & День)

Решение

Здесь очевидная информация такая:

Код	Запрос	Кол-во страниц
А	Мороз & Солнце & День & Чудесный	Минимальное
Б	Мороз Солнце День Чудесный	Максимальное
В	Мороз & Солнце & Чудесный	Больше, чем А
Г	Чудесный (Мороз & Солнце & День)	Больше, чем А, но меньше, чем Б

Но эта информация не дает возможности сравнить результаты запросов В и Г.

Можно рассуждать так. Количество страниц, отобранных по запросу Г для условия «Чудесный» не может быть меньше количества страниц, отобранных по запросу В, в котором слово «Чудесный» также представлено (сравните на рис. 8.8 области 10 и 1, или 10 и 2, или 10 и 3). Это значит, что запрос Г «вернет» больше страниц, чем запрос В.

Ответ: АВГБ.

6.2. Задания для самостоятельной работы

Во всех предложенных ниже заданиях:

1) в таблице приведены запросы к поисковому серверу. Для каждого запроса указан его код – соответствующая буква от А до Г. Расположите коды запросов слева направо;

– в порядке возрастания количества страниц, которые нашёл поисковый сервер по каждому запросу;

– в порядке убывания количества страниц, которые нашёл поисковый сервер по каждому запросу;

По всем запросам было найдено разное количество страниц;

2) в запросах использованы указанные ранее знаки логических операций.

14.

Код	Запрос
А	Рыжий Честный Влюблённый Лис
Б	Рыжий Честный Влюблённый
В	Рыжий & Честный
Г	Рыжий Честный

15.

Код	Запрос
А	Солнце & Воздух
Б	Солнце Воздух Вода
В	Солнце Воздух Вода Огонь
Г	Солнце Воздух

16.

Код	Запрос
А	Лебедь Рак
Б	Лебедь Рак Щука
В	Лебедь & Рак
Г	Лебедь Рак Щука Озеро

17.

Код	Запрос
А	Мороз Солнце День Чудесный
Б	(Мороз & Солнце) День

В	Мороз & Солнце
Г	Мороз & Солнце & День & Чудесный

18.

Код	Запрос
А	Осёл & Козёл & Мишка
Б	Мартышка (Осёл & Козёл & Мишка)
В	Мартышка Осёл Козёл Мишка
Г	Мартышка & Осёл & Козёл & Мишка

19.

Код	Запрос
А	Мороз & Солнце
Б	Мороз & Солнце & День & Чудесный
В	Мороз Солнце День Чудесный
Г	(Мороз & Солнце) День

20.

Код	Запрос
А	Мартышка & Осёл & Козёл & Мишка
Б	Осёл & Козёл & Мишка
В	Мартышка Осёл Козёл Мишка
Г	Мартышка (Осёл & Козёл & Мишка)

Методика выполнения задания 16 демонстрационного варианта ЕГЭ по информатике 2025 года

Условие [1]

Алгоритм вычисления значения функции $F(n)$, где n – натуральное число, задан следующими соотношениями:

$$F(n) = 1 \text{ при } n = 1;$$

$$F(n) = (n - 1) \times F(n - 1), \text{ если } n > 1.$$

Чему равно значение выражения $(F(2024) + 2 \times F(2023)) / F(2022)$?

Решение

Приведённая в условии функция является рекурсивной, то есть при расчётах она использует саму себя (обращается) как вспомогательную, но с меньшим значением аргумента. Так как начальные значения аргументов в заданном выражении достаточно большие числа, то при расчётах будет много рекурсивных вызовов, что потребует значительного времени для получения искомого значения.

Вместо этого лучше преобразовать заданное выражение с учётом того, что:

$$F(2023) = 2022 \times F(2022)$$

$$F(2024) = 2023 \times F(2023) = 2023 \times 2022 \times F(2022)$$

Обратим внимание на то, что сначала записано значение функции с меньшим аргументом.

Тогда заданное выражение примет вид:

$$\frac{2023 \times 2022 \times F(2022) + 2 \times 2022 \times F(2022)}{2022 \times F(2022)} = 2023 \times 2022 + 4044.$$

Результат последнего выражения можно определить, используя стандартную программу Калькулятор или электронную таблицу.

«Не так страшен чёрт...», или Методика выполнения задания 27 демоварианта ЕГЭ по информатике 2025 года

Д.М. Златопольский

Условие [1]

Учёный решил провести кластеризацию некоторого множества звёзд по их расположению на карте звёздного неба. Кластер звёзд – это набор звёзд (точек) на графике, лежащий внутри прямоугольника высотой H и шириной W . Каждая звезда обязательно принадлежит только одному из кластеров¹.

Истинный центр кластера, или центроид, – это одна из звёзд на графике, сумма расстояний от которой до всех остальных звёзд кластера минимальна. Под расстоянием понимается расстояние Евклида между двумя точками $A(x_1, y_1)$ и $B(x_2, y_2)$ на плоскости, которое вычисляется по формуле:

$$d(A, B) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

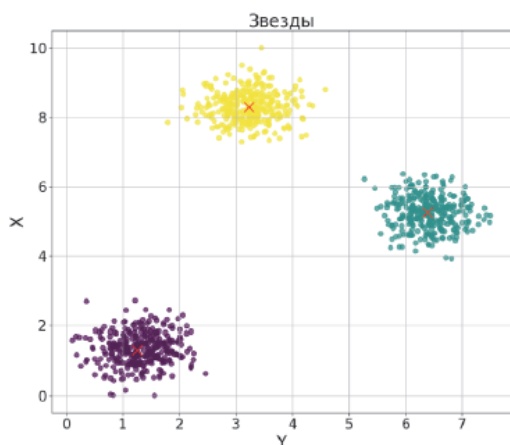
В файле А хранятся данные о звёздах **двух** кластеров, где $H = 3$, $W = 3$ для каждого кластера. В каждой строке записана информация о расположении на карте одной звезды: сначала координата x , затем координата y . Значения даны в условных единицах. Известно, что количество звёзд не превышает 1000.

В файле Б хранятся данные о звёздах **трёх** кластеров, где $H = 3$, $W = 3$ для каждого кластера. Известно, что количество звёзд не превышает 10 000. Структура хранения информации о звездах в файле Б аналогична файлу А.

Для каждого файла определите координаты центра каждого кластера, затем вычислите два числа: P_x – среднее арифметическое абсцисс центров кластеров, и P_y – среднее арифметическое ординат центров кластеров.

В ответе запишите четыре числа: в первой строке сначала целую часть произведения $P_x \times 10\,000$, затем целую часть произведения $P_y \times 10\,000$ для файла А, во второй строке – аналогичные данные для файла Б.

Возможные данные одного из файлов иллюстрированы графиком.



Внимание! График приведён в иллюстративных целях для произвольных значений, не имеющих отношения к заданию. Для выполнения задания используйте данные из прилагаемого файла.

¹ Обратите внимание на эту особенность. (Прим. автора.)

О т в е т :		

Решение

Основные этапы решения задачи:

1. Чтение исходных данных из прилагаемого файла и запись их в массив.
2. Распределение всех звёзд по кластерам.
3. Нахождение центроидов (см. условие) каждого кластера.
4. Расчёт среднего арифметического абсцисс и среднего арифметического ординат центров кластеров.
5. Вывод результатов.

Начнём с этапа 2. Для выполнения этого этапа нужно знать границы кластеров, то есть координаты вершин соответствующих прямоугольников. Эту задачу удобно решать с помощью электронной таблицы, учитывая, что исходные данные заданы не только в виде текстового файла, но и в формате *xlsx*.

Откроем лист электронной таблицы и получим в столбце E минимальные и максимальные значения координат звёзд, используя стандартные функции МИН и МАКС.

	A	B	C	D	E
1	x	y			
2	-0,332725	1,728698		мин_x	-1,788519
3	-0,454333	0,774187		макс_x	4,153360
4	-0,687317	1,831811			
5	-0,276924	1,504291		мин_y	0,076099
6	-0,530721	1,317191		макс_y	6,069280
7	-0,899409	1,132319			
...	...	...			

Примечание. Чтобы узнать номер строки с последней заполненной ячейкой в столбце, нужно, установив указатель активной ячейки в данном столбце, нажать комбинацию клавиш **Ctrl** + ↓, для возврата наверх – комбинацию **Ctrl** + ↑.

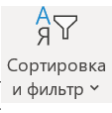
Найденные значения принадлежат некоторым «крайним» точкам. Определим также вторую координату каждой из четырёх точек:

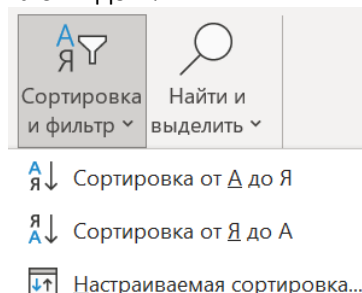
	A	B	C	D	E	F	G
1	x	y				Вторая координата	
2	-0,332725	1,728698		мин_x	-1,788519		
3	-0,454333	0,774187		макс_x	4,153360		
4	-0,687317	1,831811					
5	-0,276924	1,504291		мин_y	0,076099		
6	-0,530721	1,317191		макс_y	6,069280		
7	-0,899409	1,132319					
...	...	...					

Искомые значения лучше всего находить, используя сортировку данных в столбцах A и B.

Так, чтобы найти ординаты точек с экстремальными абсциссами, следует:

- выделить диапазон A2:B97. Быстро это можно сделать, установив указатель активной ячейки в ячейку A2 и нажать комбинацию клавиш **Ctrl** + Shift + ↓, для возврата наверх – комбинацию **Ctrl** и ↑;

- щелкнуть на инструменте Сортировка и фильтр () и появившемся окне выбрать пункт Сортировка от А до Я:

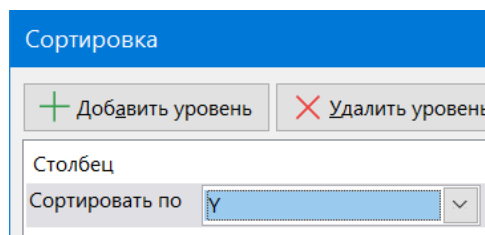


Лист примет вид:

	A	B	C	D	E	F	G
1	x	y				Вторая координата	
2	-1,788519	1,978343		мин_x	-1,788519	1,978343	
3	-1,748975	1,279293		макс_x	4,153360	3,406143	
4	-1,640905	0,234446					
5	-1,631061	1,401784		мин_y	0,076099		
6	-1,596882	2,173577		макс_y	6,069280		
7	-1,596625	1,100607					
...	...	...					
96	4,080261	5,262642					
97	4,153360	3,406143					

Найденные значения надо выписать или получить их в ячейках F2 (по формуле =B2) и F32 (по формуле =B97).

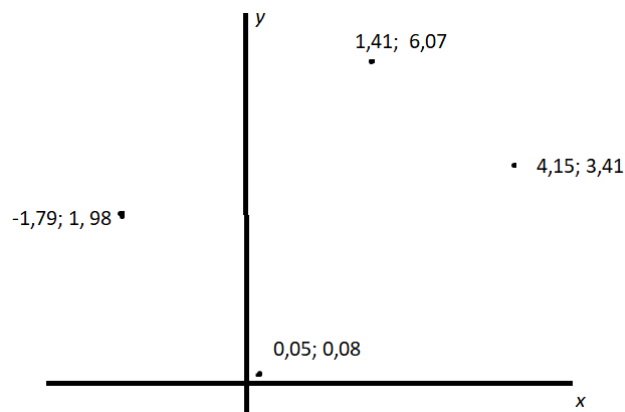
Для нахождения абсцисс точек с экстремальными ординатами следует после выделения диапазона A2:B97 и щелчка на инструменте Сортировка и фильтр выбрать вариант Настраиваемая сортировка (см. выше) и появившемся окне Сортировка в выпадающем списке поля Сортировать по выбрать пункт Y:



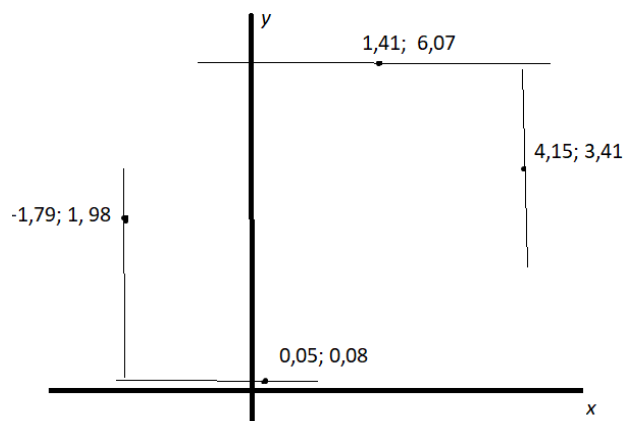
и щелкнуть на кнопке ОК. Лист примет вид:

	A	B	C	D	E	F	G
1	x	y				Вторая координата	
2	0,052110	0,076099		мин_x	-1,788519	1,978343	
3	-1,025970	0,173247		макс_x	4,153360	3,406143	
4	0,013837	0,179348					
5	-1,019088	0,202639		мин_y	0,076099	0,052110	
6	-0,859812	0,209913		макс_y	6,069280	1,411393	
7	-1,640905	0,234446					
...	...	...					
96	2,423478	6,06547					
97	1,411393	6,069280					

После нахождения координат «крайних» точек можно определить границы кластеров. Для этого следует на листе бумаги нарисовать эскиз следующего вида:



Анализ расстояний между точками с учётом известных размеров кластеров-прямоугольников ($H = 3$, $W = 3$) показывает, что две звезды в верхней правой части эскиза являются крайними точками одного кластера, а остальные две – второго. Звезда с координатами (1,41; 6,07) является крайней верхней точкой первого кластера, с координатами (4,15; 3,41) – крайней правой. Звезда с координатами (0,05; 0,08) является крайней нижней точкой второго кластера, с координатами (-1,79; 1,98) – крайней левой точки этого кластера. Отметим это на схеме:



После этого, зная размеры кластеров, можно найти границы двух кластеров. Это удобно сделать на том же листе электронной таблицы:

	...	D	E	F	G	H	I	J	K
1				Вторая координата					
2	...	мин_x	-1,788519	1,978343					
3	...	макс_x	4,153360	3,406143	макс_x_1	4,153360		мин_x_1	1,153360
4	...				макс_y_1	6,069280		мин_y_1	3,069280
5	...	мин_y	0,076099	0,052110					
6	...	макс_y	6,069280	1,411393	мин_x_2	-1,788519		макс_x_1	1,211481
7	...				мин_y_2	0,076099		макс_y_2	3,076099
8	...								

Примечания

1. Стрелками показаны зависимые ячейки.
2. Значения в столбце K определяются по формулам с использованием +3 и -3.

Удобно границы кластеров оформить в порядке их увеличения в виде:

1-й кластер			2-й кластер		
x	1,153360	4,153360	x	-1,788519	1,211481
y	3,069280	6,069280	y	0,076099	3,076099

Найденные границы будут использоваться на остальных этапах решения задачи, которые будут выполняться с помощью программы на языке программирования.

Прежде чем описывать программу, обсудим структуру строки файла с исходными данными. Открыв прилагаемый к заданию файл А (его имя – demo_2025_27_A.txt), например, с помощью программы Блокнот, можно увидеть следующую верхнюю часть файла;

X	Y
-0,3327251201948310	1,7286976051060200
-0,4543325769267230	0,7741868621101550
-0,6873170750307440	1,8318107339950500

В первом столбце есть положительные значения без знака «+» (!), во втором таких значений нет (в данном случае!). Число знаков в дробной части чисел – 16. Между столбцами имеется не ряд пробелов, а знак табуляции (его можно увидеть, открыв файл с помощью редактора Microsoft Word).

Теперь – программа.

Исходных данные (координаты всех звёзд) будем хранить в двумерном массиве с именем *коорд* из 1000 элементов вещественного типа (в первом столбце массива – координата *x*, во втором – *y*).

Абсциссу и ординату звёзд первого кластера будем хранить в двух массивах *x1* и *y1* из 1000 элементов каждый, а звёзд первого кластера – *x1* и *y1* того же размера. Фактическое количество звёзд, координаты которых будут записаны в эти массивы, будут хранить переменные *к1* и *к2*.

Ещё три переменные величины:

к – количество строк в файле (количество звёзд);

ст – очередная строка;

ф – переменная файлового типа.

Итак, заполнение массива *коорд*:

```

ф := открыть на чтение("demo_2025_27_A.txt")
| Читаем 1-ю строку (заголовки)
ввод ф, ст
к := 0
| Читаем и обрабатываем остальные строки
нц пока не конец файла (ф)
  ввод ф, ст | Очередная строка ст
  к := к + 1 | Учитываем её в количестве строк к
  | Заменяем в ней запятые на точки
  нц для j от 1 до длин(ст)
    если ст[j] = ","
      то
        ст[j] := " "
    все
  кц
  | Выделяем в ней абсциссу и ординату,
  | учитывая возможные знак «минус»
  | Обратите внимание на индексы символов при выделении
  если ст[1] = "-"
    то

```

```

        коорд[к, 1] := лит_в_вещ(ст[1 : 19], успех)
        коорд[к, 2] := лит_в_вещ(ст[21 : длин(ст)], успех)
        | 20-й символ – знак табуляции
    иначе
        коорд[к, 1] := лит_в_вещ(ст[1 : 18], успех)
        коорд[к, 2] := лит_в_вещ(ст[20 : длин(ст)], успех)
    все
кц
заккрыть (ф)

```

Зная границы двух кластеров, можно распределить все звёзды по ним:

```

| Распределяем по кластерам
к1 := 0
к2 := 0
нц для i от 1 до к | Рассматриваем координаты всех звёзд
    если коорд[i, 1] >= 1.153360 и коорд[i, 1] <= 4.153360
        и коорд[i, 2] >= 3.069280 и коорд[i, 2] <= 6.069280
            | Границы кластеров целесообразно скопировать
            | с листа электронной таблицы
        то | Текущая звезда относится к 1-му кластеру
            к1 := к1 + 1
            x1[к1] := коорд[i, 1]
            y1[к1] := коорд[i, 2]
        иначе | ко 2-му кластеру
            к2 := к2 + 1
            x2[к2] := коорд[i, 1]
            y2[к2] := коорд[i, 2]
    все
кц

```

Этап 3 (нахождение центроидов каждого кластера) выполняется следующим образом. Рассматриваются все звёзды кластера и для каждой из них определяются и суммируются расстояния от неё до остальных звёзд этого кластера. Из всех сумм запоминается минимальная сумма и координаты звезды, при которых она достигается.

Для нахождения расстояния между звёздами целесообразно создать вспомогательную функцию, которая на школьном алгоритмическом языке имеет вид:

```

алг вещь рас(арг вещь x1, y1, x2, y2)
нач
    | Значение функции
    знач := sqrt((x1 - x2) ** 2 + (y1 - y2) ** 2)
кон

```

Прежде чем представлять фрагмент программы, относящийся к этапу 3, заметим, что с целью упрощения можно для каждой звезды определять также расстояние «до самой себя» (оно будет равно нулю).

В этом фрагменте будут использоваться также следующие величины:

сумма – сумма расстояний от текущей звезды данного кластера;

мин_сумма – минимальная сумма;

х_центра1 и *у_центра1* – координаты звезды с минимальной суммой расстояний в первом кластере;

х_центра2 и *у_центра2* – то же, второго.

```

| Этап 3
| 1-й кластер
мин_сумма := 100000 | Условное начальное значение
нц для i от 1 до k1
    сумма1 := 0
    нц для j от 1 до k1
        сумма := сумма + рас(x1[i], y1[i], x1[j], y1[j])
    кц
    | Сравниваем
    если сумма < мин_сумма
        то | Получена новая минимальная сумма
            | Запоминаем её
            мин_сумма := сумма
            | и координаты текущей звезды
            x_центра1 := x1[i]
            y_центра1 := y1[i]
    все
кц
| 2-й кластер
мин_сумма := 100000
нц для i от 1 до k2
    сумма := 0
    нц для j от 1 до k2
        сумма := сумма + рас(x2[i], y2[i], x2[j], y2[j])
    кц
    если сумма < мин_сумма
        то
            мин_сумма := сумма
            x_центра2 := x2[i]
            y_центра2 := y2[i]
    все
кц

```

Расчёт и вывод среднего арифметического абсцисс и среднего арифметического ординат центров кластеров:

```

Px := int((x_центра1 + x_центра2 )/2 * 10000)
Py := int((y_центра1 + y_центра2 )/2 * 10000)
вывод нс, Px, " ", Py

```

Задания для самостоятельной работы

1. Разработайте программу выполнения задания применительно к файлу А на языке программирования, которым владеете.
2. Выполните обсуждаемое задание применительно к файлу В (задача решается аналогично).

Литература

1. Демонстрационный вариант контрольных измерительных материалов Единого государственного экзамена 2025 года по информатике (Демоверсии, спецификации, кодификаторы 2025. Информатика). <https://fipi.ru/ege/demoversii-specifikacii-kodifikatory#!/tab/151883967-5>

Когда человек хочет передвинуть гору, он начинает с того, что убирает маленькие камни.

Восточная мудрость

ТрОечные (не троичные) числа

Назовём натуральное число «троечным», если оно делится на 3 или содержит тройку в десятичной записи. Возьмём некоторое число N . Среди чисел от 1 до N есть какое-то количество троечных чисел. Например, при $N = 16$ троечных чисел – 6 (3, 6, 9, 12, 13, 15). Нас же будут интересовать только такие числа N , что среди чисел от 1 до N ровно половина троечных. Сколько есть таких чисел, не превышающих 1000?

Для ответа на вопрос разработаем компьютерную программу. Как принято в нашем журнале, используем школьный алгоритмический язык (система программирования КуМир). Русский синтаксис этого языка и большое число комментариев делает программу максимально понятной и легко переводимой на любой другой язык программирования.

Итак.

Общая схема работы программы:

Надо перебрать все числа N от 4 до 1000

 Для каждого из них рассмотреть все натуральные числа, меньшие N
 и проверить, являются ли троечными.

 Если являются

 то

 учесть в их количестве троечных чисел

Сравнить количество троечных чисел и текущее значение N

Если $N = \text{количество троечных чисел} * 2$

 то

 учесть текущее значение N в искомом количестве.

Вывести на экран искомое количество чисел

Узнать, что некоторое число делится на 3, можно определив остаток от его деления на 3. Если остаток равен нулю, то это число кратно трём.

Для того чтобы узнать, имеется ли цифра 3 в записи числа, нужно рассмотреть все его цифры. Для числа m любой «значности» это можно сделать так:

есть3 := **нет** | Сначала принимаем, что цифры 3 в числе нет

нц пока $m > 0$

 | Проверяем последнюю цифру числа m

если $\text{mod}(m, 10) = 3$

то

 есть3 := **да**

выход

все

 | Отбрасываем последнюю цифру числа m

$m := \text{div}(m, 10)$

кц

где **есть3** – переменная логического типа, фиксирующую факт «встречи» цифры 3 (можно также использовать переменную целого типа, принимающую значения 0 и 1). Значение этой переменной будет говорить о том, имеется ли цифра 3 в числе n или нет.

Остальные переменные, которые будем использовать:

N – проверяемое число;

m – числа, меньшие N (среди них надо будет проводить поиск троечных чисел);

$кол$ – количество троечных чисел среди чисел от 1 до N ;

$кол_иск$ – количество искоемых чисел.

Прежде чем представлять программу (в которой будет использована еще одна переменная – $m2$), заметим, что проверять можно только чётные числа N (почему – установите самостоятельно).

```
алг Подсчёт
нач цел N, m, m2, кол3, кол_иск, лог есть3
  кол_иск := 0 | Начальное значение искомого количества
  нц для N от 4 до 1000 шаг 2 | Проверяем только чётные числа
    кол := 0
    нц для m от 1 до N
      | Проверяем, есть ли цифра 3
      m2 := m
      есть3 := нет
      нц пока m2 > 0
        | Проверяем последнюю цифру числа m2
        если mod(m2, 10) = 3
          то
            есть3 := да
        все
        | Отбрасываем последнюю цифру
        m2 := div(m2, 10)
      кц
      | Значение переменной есть3 определено
      | Проверяем, является ли число m троечным
      если mod(m, 3) = 0 или есть3
        то
          кол3 := кол3 + 1
      все
    кц
    | Сравниваем N и кол3
    если N = 2 * кол3
      то | Встретилось ещё одно подходящее число
        кол_иск := кол_иск + 1
    все
  кц
  | Вывод ответа
  вывод кол_иск
кон
```

Вопрос: почему использована переменная $m2$?

Читаемость приведённой программы улучшится, если проверку, является ли некоторое число троечным, проводить с помощью вспомогательной функции логического типа. На школьном алгоритмическом языке такая функция имеет вид:

```
алг лог троечное (арг цел m)
нач цел m2, лог есть3
  | Проверяем
  m2 := m
  есть3 := нет
  нц пока m2 > 0
    | Проверяем последнюю цифру числа m2
    если mod(m2, 10) = 3
```

```

        то
            есть3 := да
    все
        | Отбрасываем последнюю цифру
        m2 := div(m2, 10)
    кц
    | Значение функции
    если mod(m, 3) = 0 или есть3
        то
            знач := да
        иначе
            знач := нет
    все
кон

```

С использованием этой функции основная часть программы становится краткой и понятной:

```

алг Подсчёт_2
нач цел N, m, кол3, кол_иск
    кол_иск := 0
    нц для N от 4 до 1000 шаг 2
        кол3 := 0
        нц для m от 1 до N
            | Проверяем, является ли число m троечным
            если троечное(m)
                то
                    кол3 := кол3 + 1
            все
        кц
        если N = 2 * кол3
            то
                кол_иск := кол_иск + 1
        все
    кц
    вывод нс, кол_иск
кон

```

И ещё одно полезное замечание. Если бы вспомогательная функция *троечное(m)* была бы сложной и трудночитаемой, то и её можно было упростить, разработав для неё вспомогательную функцию логического типа, фиксирующую факт наличие в числе *m* цифры 3:

```

алг лог есть_цифра_3(арг цел m)
нач цел m2, лог есть3
    m2 := m
    есть3 := нет
    нц пока m2 > 0
        | Проверяем последнюю цифру числа m2
        если mod(m2, 10) = 3
            то
                есть3 := да
        все
        | Отбрасываем последнюю цифру
        m2 := div(m2, 10)
    кц
    знач := есть3
кон

```

С её использованием функция, проверяющая, является ли число m троечным, упрощается:

```
алг лог троечное_2 (арг цел m)
нач
  | Значение функции
  если mod(m, 3) = 0 или есть_цифра_3(m)
  то
    знач := да
  иначе
    знач := нет
все
кон
```

Итак, вывод – когда программа объемная и трудночитаемая, то в ней целесообразно выделить фрагменты, в которых решается некоторая частная задача, и оформить его в виде вспомогательной функции или процедуры². Аналогично при необходимости можно упростить и вспомогательную функции/процедуру.

Разработав любой вариант соответствующей программы на языке программирования, который вы изучаете, определите искомое количество и все соответствующие числа N . Ответы присылайте в редакцию.

Книжная полка

Полезная книга

Издательство ДМК Пресс предлагает книгу «1400 задач по программированию» <https://dmkpress.com/catalog/computer/handbooks/978-5-97060-827-2/>.



Если вы, уважаемый читатель, хотите стать программистом, то имейте в виду, что одним из условий этого является большой опыт решения задач по программированию. Решайте задачи, абстрактные и содержательные, на 5 минут и на день работы. Все это обязательно вам пригодится. Чем больше опыт решения задач, тем больше вероятность, что уже решенная задача или аналогичная ей встретится вам в будущем. В приобретении такого опыта вам поможет данная книга.

Книга будет полезной также учителям и преподавателям информатики.

В ней представлено более 1400 разноуровневых задач по всем разделам курса программирования в средней школе. Задачи эти могут быть использованы при объяснении нового материала, для организации самостоятельной работы учащихся и проведения контрольных мероприятий. Ряд задач предназначен для внутришкольных олимпиад, для использования в кружках по программированию и т. п. Такие задачи помечены символом «*» или вынесены в раздел «Задачи повышенной сложности». Все задачи могут быть решены на любом языке программирования (Паскале, Python, Си, школьном алгоритмическом языке и т. д.).

Через сайт издательства <https://dmkpress.com/> книгу можно приобрести гораздо дешевле, чем в магазине.

² В языке программирования Python оба эти вида вспомогательных подпрограмм называются «функциями».

Итоги конкурса № 71 «Магические квадраты»

Напомним, что в качестве задания этого конкурса было предложено решить две задачи, связанные с так называемыми «магическими квадратами».

Магический (или волшебный) квадрат – это квадратная таблица, заполненная числами без их повторений таким образом, что сумма чисел в каждой строке, каждом столбце и на обеих диагоналях одинакова. Эту сумму называют «магической суммой».

На рис. 1–4 показаны части магических квадратов размером 3×3 .

Задача 1. Условие

Возможны единственные магические квадраты, части которых показаны на 1–3. Получите квадраты полностью, заполнив свободные клетки.

5	6	
	7	

Рис. 1

7		
8		13

Рис. 2

	4	
6		10

Рис. 3

Решение

Как указывалось в условии, для решения удобно использовать электронную таблицу Microsoft Excel или аналогичную программу.

Для случая на рис. 1 оформим лист следующим образом.

В ячейку C1 будем вводить то или иное число («переменное»), а в остальных ячейках квадрата и в оттенённых ячейках запишем формулы для расчёта значений в них, соответствующие правилам магического квадрата.

	A	B	C	D	E
1	5	6			
2		7			
3					
4					
5					

Формулы в ячейках:

1) в E1 – сумма чисел в первой строке квадрата ($=\text{СУММ}(A1:C1)$)

Её можно скопировать в ячейки E2:E3;

2) в A5 – сумма чисел в первом столбце квадрата ($=\text{СУММ}(A1:A3)$)

Её можно скопировать в ячейки B5:C5;

3) в D4 – сумма чисел на диагонали квадрата ($=A1+B2+C3$);

4) в B5 – известная сумма в E1 минус известные числа ($=E1-B1-B2$)

5) в A5: $=E1-C1-B2$;

6) в C3: $=E1-A3-B3$;

7) в A2: $=E1-A1-A3$;

8) в C2: $=E1-A2-B2$.

Пробуем.

Если в ячейку C1 ввести число 4, то картина будет такой:

	A	B	C	D	E
1	5	6	4		15
2	6	7	2		15
3	4	2	9		15
4				21	
5	15	15	15		

Такой вариант не подходит (есть повторы и сумма на одной диагонали отличается). Аналогично при C1 = 8:

	A	B	C	D	E
1	5	6	8		19
2	10	7	2		19
3	4	6	9		19
4				21	
5	19	19	19		

Значение 10 подходит:

	A	B	C	D	E
1	5	6	10		21
2	12	7	2		21
3	4	8	9		21
4				21	
5	21	21	21		

Аналогично можно получить решения для вариантов на рис. 2-3:

	A	B	C	D	E
1	7	11	12		30
2	15	10	5		30
3	8	9	13		30
4				30	
5	30	30	30		

	A	B	C	D	E
1	11	4	9		24
2	6	8	10		24
3	7	12	5		24
4				24	
5	24	24	24		

Решение задачи 2 будет описано в следующем выпуске нашего журнала.

А здесь перечислим читателей, которые стали победителями конкурса и будут награждены дипломами:

— Голуб Марк, Москва, школа № 1530 «Школа Ломоносова», учитель **Каменецкая Т.С.**;

— Лукьянов Кирилл, Барнаульский государственный педагогический колледж имени В.К. Штильке, преподаватель **Лукьянова Н.В.**;

— Пусь Иван, г. Пенза, школа № 73, учитель **Диков А.В.**;

— Чиркина Юлия, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Поздравляем победителей!

Ответы и решения задач и головоломок, опубликованных в журнале «Мир информатики», выпуск № 97 (февраль 2025 г.), представили:

— Абрикосова Марина и Ножкин Сергей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Александрова Кристина, Бондарчук Павел, Галкина Анастасия, Гребенькова Вера, Ковальчук Иван, Короткина Анна, Мурзина Евгения, Наумов Виктор, Норкулова Бону и Саргсян Диана, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Брагин Алексей, Макаревич Тимур, Назарян Артём и Шухман Марк, г. Пенза, школа № 73, учитель **Диков А.В.**;

— Голуб Марк, Николенко Максим, Олонецкий Олег и Юдин Илья, Москва, школа № 1530 «Школа Ломоносова», учитель **Каменецкая Т.С.**;

— Замилацкая София, Порохин Даниил, Свинарёв Денис, Степанян Даниэль, Хамуратов Адам и Юсюфзай Аджира, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Марков Максим, Владимирская обл., г. Карабаново, школа № 7 им. А.П. Чулкова, учитель **Тощева Т.В.**;

— Незванова Ирина и Чиркина Юлия, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Прокопенко Анастасия и Хорькова Анна, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

Спасибо всем! Молодцы!

Уважаемые коллеги!

В летних номерах нашего журнала, как и в предыдущие годы, будет представлена рубрика «Доска почёта “Мира информатики”», в которой будут опубликованы фотографии учащихся, ставших победителями конкурсов, а также принявшие наиболее активное участие в решении задач и головоломок в текущем учебном году.

Просим в мае месяце прислать фотографии таких учащихся (можно «неформальные», в том числе групповые).

Напоминаем наш адрес: mirinfo@infojournal.ru

Ждём!

Книга «Подготовка к успешной сдаче ЕГЭ по информатике»

Издательство «ДМК Пресс» предлагает книгу «Подготовка к успешной сдаче ЕГЭ по информатике» (<https://dmkpress.com/catalog/computer/handbooks/978-5-93700-236-5/>)

Большинство книг, связанных с Единым государственным экзаменом по информатике, представляют собой сборники заданий для самопроверки или «решебники», в которых приводится решение заданий без подробных комментариев. Данная книга отличается от них тем, что в ней после соответствующей теоретической подготовки рассказывается о типовых вариантах всех 27 заданий ЕГЭ и особенностях каждого из них, приводятся различные методы их выполнения, даются полезные советы и рекомендации, обеспечивающие ускорение выполнения заданий и уменьшающие вероятность ошибки. После знакомства с этими вопросами читатель может проверить свои знания и умения, выполнив предлагаемые в книге задания для самостоятельной работы.

В книге учтены задания из демонстрационных вариантов ЕГЭ по информатике нескольких последних лет до варианта 2024 года включительно.

При изложении методики выполнения заданий с помощью собственных программ логика действий в них описывается на школьном алгоритмическом языке (система программирования КуМир). Русский синтаксис этого языка и большое число комментариев к программам делают приводимые методики и программы максимально понятными, и читатель сможет разработать аналогичные программы на языке программирования, которым владеет.

Методика выполнения заданий в среде электронной таблицы описывается применительно к программе Microsoft Excel. При использовании читателем другой программы, естественно, должны быть учтены ее особенности.

Кроме учащихся 11-го класса, готовящихся к сдаче ЕГЭ, книгу могут использовать учителя и преподаватели информатики. Конечно, она будет полезна и учащимся 9–10-х классов, планирующим сдавать экзамен в будущем.

Через сайт издательства <https://dmkpress.com/> книгу можно приобрести дешевле, чем в магазине (даже с учетом стоимости доставки). Книга предлагается также на [avito.ru](https://www.avito.ru/).