

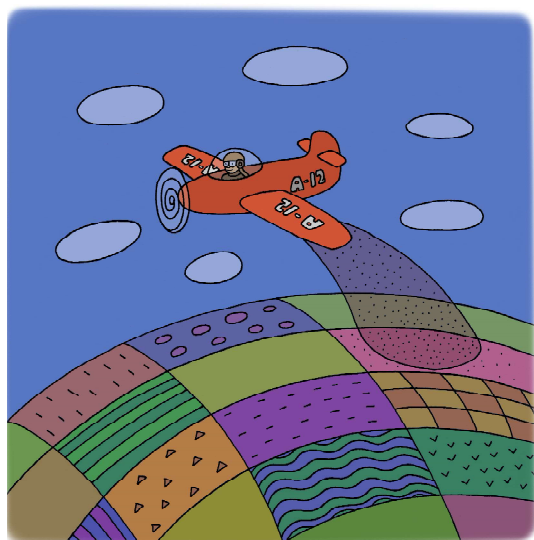
Богомолова Ольга Борисовна,
Усенков Дмитрий Юрьевич

ЕГЭ-2018: ОЧЕРЕДНЫЕ «НОВЫЕ ГРАБЛИ»

Разработчики заданий ЕГЭ по информатике и те, кто утверждает эти задания к использованию на экзамене, продолжают всю «любовь» школьников. Новые задачи, заявленные в «демонстрационном варианте» и предлагаемые в текущих тренажах и диагностиках, стали еще сложнее.

Ну что же – знакомимся с очередными «подарками» одиннадцатиклассникам от «добрых дядь и тетей», которые вызвали наибольшие трудности при их решении.

Задача 2¹. Логическая функция F задана выражением $(a \vee b) \rightarrow (b \equiv c)$. Имеется частично заполненный фрагмент таблицы истинности функции F , содержащий *неповторяющиеся* строки.



Требуется определить соответствие каждой из переменных (a, b, c) столбцам таблицы истинности. В ответе нужно без пробелов и разделителей записать буквы a, b и c в том порядке, в каком в таблице следуют соответствующие им столбцы.

Переменная 1	Переменная 2	Переменная 3	Функция
???	???	???	F
0		0	0
		0	0

Подобные задачи уже встречались на ЕГЭ ранее. Усложнение здесь в том, что в ячейках таблицы имеется слишком мало исходных данных, и это существенно усложняет рассуждения.

Решение 1 (метод рассуждений)

Рассмотрим заданную функцию:

$$(a \vee b) \rightarrow (b \equiv c).$$

1. Согласно таблице, в обоих случаях значение F равно 0. А операция следования (\rightarrow) дает нуль только в одном единственном случае: $1 \rightarrow 0$. Следовательно, для обеих строк таблицы должно обязательно соблюдаться условие: $(a \vee b) = 1$ И $(b \equiv c) = 0$.

2. Начнем анализ таблицы со столбца, в котором обе ячейки заполнены нулями («Переменная 3»).

Предположим, что переменная b , которая в обоих случаях равна нулю, – это «Переменная 3». Тогда из условия $(b \equiv c) = 0$ получаем, что переменная c может быть

¹ Номера задач соответствуют типовой нумерации заданий ЕГЭ. – Прим. авт.

только ненулевой (нам нужно, чтобы тождество было ложным, а значит, переменные b и c должны быть разными). Но если $b = 0$, то из условия $(a \vee b) = 1$ переменная a должна быть равна только единице. Делаем вывод: если в какой-то строке таблицы $b = 0$, то в этой строке **не должно быть** больше никаких других нулей! А первая строка таблицы этому противоречит. Следовательно, b не может быть переменной «Переменная 3».

Предположив, что b – это «Переменная 1», и рассуждая аналогично, получаем опять-таки противоречие в первой строке таблицы: для нулевого значения b в той же строке ни в какой ячейке нулей не должно быть. Значит, b не может быть и переменной «Переменная 1».

Остается только одна возможность: b – это «Переменная 2»:

Переменная 1	Переменная 2	Переменная 3	Функция
???	b	???	F
0		0	0
		0	0

3. Теперь требуется решить, где находится переменная a , а где c .

Опять-таки начинаем анализ с полностью заполненного нулями столбца.

Предположим, что «Переменная 3» – это a . Тогда:

– из условия $(a \vee b) = 1$ при **обоих** нулевых значениях a получаем, что $b = 1$;

– из условия $(b \equiv c) = 0$ для $b = 1$ в **обеих** строках таблицы c должна быть равна 0.

Но тогда таблица примет вид:

Переменная 1	Переменная 2	Переменная 3	Функция
c	b	a	F
0	1	0	0
0	1	0	0

В этом случае у нас получаются две одинаковые строки, тогда как по условию эти строки обязаны быть неповторяющимися. Вывод: наше предположение неверно, тогда остается только один-единственный вариант: «Переменная 3» – это c , тогда «Переменная 1» – это a . Таблица в этом случае имеет вид:

Переменная 1	Переменная 2	Переменная 3	Функция
a	b	c	F
0		0	0
		0	0

Ответ: abc .

Примечание. Если бы мы начали с предположения, что «Переменная 3» – это c , то получили бы, что из условия $(b \equiv c) = 0$ для **обоих** нулевых значений c обязательно $b = 1$, и, соответственно, из условия $(a \vee b) = 1$ нам допустимо любое значение a , то есть сделанное предположение вполне может быть правильным. Но тогда обязательно надо было бы проверить и вторую возможность – предположение, что «Переменная 3» – это a , поскольку иначе анализ был бы неполным. И мы бы в этом случае получили, что данное предположение неверно, а, следовательно, правильно наше первое предположение – что «Переменная 3» – это c .

Решение 2 (построение полной таблицы истинности)

Если приведенные выше логические рассуждения покажутся слишком сложными, то можно «пойти в лобовую атаку» и построить полную таблицу истинности (табл. 1), благо переменных здесь всего три – не то, что раньше, когда их могло быть пять или шесть. (Напомним, что все возможные со-

Табл. 1

a	b	c	$(a \vee b)$	$(b \equiv c)$	$F = (a \vee b) \rightarrow (b \equiv c)$
0	0	0	0	1	1
0	0	1	0	0	1
0	1	0	1	0	0
0	1	1	1	1	1
1	0	0	1	1	1
1	0	1	1	0	0
1	1	0	1	0	0
1	1	1	1	1	1

четания значений переменных a, b, c в левой части таблицы – это последовательные двоичные числа от 000 до 111.)

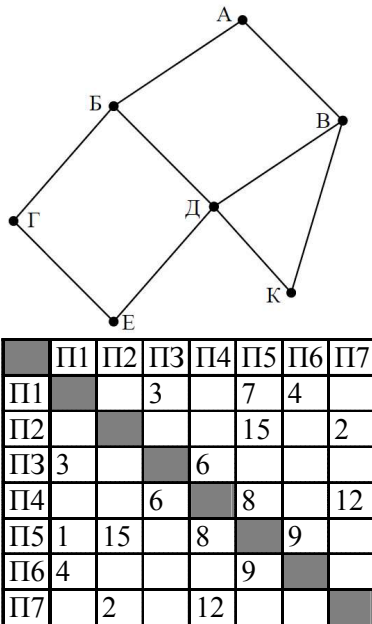
Выделим в этой таблице строки, в которых значение функции F нулевое, и посмотрим на соответствующие значения исходных переменных:

a	b	c
0	1	0
1	0	1
1	1	0

Видим, что у нас в этой таблице есть единственная строчка с двумя нулями (первая) – и только один столбец, где есть два нуля (столбец переменной c). Сравнивая эти строку и столбец с заданной в условии «частичной» таблицей, нетрудно заметить, что «Переменной 3» может быть только c (поскольку только для c мы получили в соответствующем столбце два нуля). А по сравнению первой строки нашей таблицы со строкой с двумя нулями в исходной таблице получаем, что этот второй нуль для переменной a может быть только «Переменной 1». Соответственно то, что «Переменная 2» – это b , определяется методом исключения.

Ответ: abc .

Задача 3. На рисунке схема дорог изображена в виде графа, а в таблице содержатся сведения о длине этих дорог в километрах.



Нумерация населённых пунктов в таблице никак не связана с буквенными обозначениями этих пунктов на графе. Требуется определить длину кратчайшего пути из пункта A в пункт D , если передвигаться можно только по указанным дорогам. В ответе записывается целое число – длина кратчайшего пути из A в D в километрах.

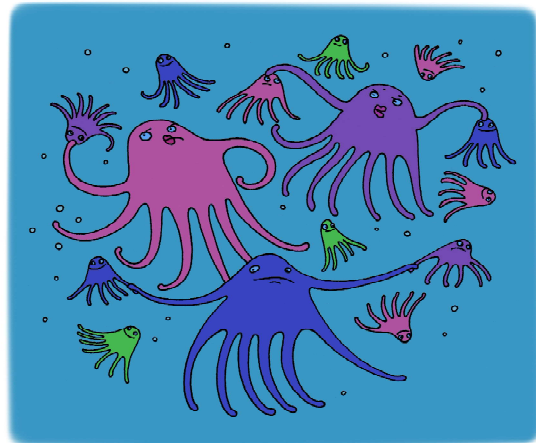
Комментарии к решению

Эта задача усложнена по сравнению со всеми ее аналогами, предлагавшимися в прошлые годы, но не намного. На самом деле нам просто предлагают «объединение» двух простых, ранее уже встречавшихся задач:

- сначала надо, анализируя граф и таблицу, определить, какие буквы соответствуют каким столбцам/строкам таблицы;
- затем уже по полученной таблице нужно определить кратчайший путь от A до D , как это делалось в задачах на графы, предлагавшихся на ЕГЭ примерно четыре года назад.

Научиться решать обе упомянутые разновидности «графовых» задач можно, например, по книге: *Богомолова О. Б.* Информатика. Новый полный справочник для подготовки к ЕГЭ (лучше – наиболее «свежего» года издания, так как справочник дополняется и корректируется каждый год). Поэтому здесь полное решение не рассматривается.

Задача 4. Даны фрагменты двух таблиц из базы данных. Каждая строка таблицы Б содержит информацию (идентификационный номер ID) о ребёнке и одном из его родителей; полную информацию о каждом ре-



ID	Фамилия И.О.	Пол	Год рождения
866	Кравец Д. К.	ж	1942
867	Тошич Б. Ф.	м	1938
879	Гонтарь В. А.	ж	1998
885	Крон К. Г.	м	1990
900	Кислюк Л. А.	ж	2012
904	Петраш А. И.	м	1981
911	Тошич А. Б.	ж	1971
932	Петраш П. А.	ж	2016
938	Тошич И. Б.	м	1974
949	Седых Г. Р.	ж	1966
970	Кислюк А. П.	м	1968
995	Тошич Т. И.	ж	2002
1017	Тошич П. И.	м	2003
1026	Мухина Р. Г.	ж	1983
1041	Сайко М. А.	ж	2010
1056	Кислюк П. А.	м	1989
...

ID Родителя	ID Ребёнка
866	911
866	938
867	911
867	938
911	879
911	1041
1026	900
938	995
938	1017
949	995
949	1017
970	879
970	1041
1026	932
904	900
904	932
...	...

бенке или родителе можно найти по его ID в таблице А.

На основании данных таблиц определите, у скольких детей в момент рождения матерям было больше 35 полных лет. При вычислении ответа учитывайте только информацию из приведённых фрагментов таблиц.

Комментарии к решению

Решение подобных задач с базами данных уже неоднократно описывалось ранее (например, в том же «Новом полном справочнике для подготовки к ЕГЭ» или в более ранней статье авторов: Богомолова О. Б., Усенков Д. Ю. Родственники и прочие реляции (новые задачи с базами данных) // Информатика. 2012. № 3. С. 38–47). Но ранее требовалось искать для заданного ФИО персонажа его родственников – родителей, детей, бабушек-дедушек, дядьев, теток, племянников и племянниц, внуков и внучек. Теперь же нам предлагается простейшая родственная связь «мать – ребенок», полностью соответствующая правой таблице: достаточно перебирать ее сверху вниз строка за строкой. Но при этом для каждой такой пары ID-номеров надо по левой таблице проверять условия:

– пол родителя должен быть (понятно) женский;

– разница года рождения ребенка и родителя должна превышать заданное число (в данном случае 35). Или, что несколько проще, сумма года рождения родительницы и заданного числа 35 должна быть не больше года рождения ребенка.

Например, для первой строки таблицы Б («866 – 911») в таблице А видим, что мать – Кравец Д. К., и разница года ее рождения (1942) и года рождения ребенка – Тошич А. Б. (1971) меньше нужных 35 лет, значит, эта строка нам не подходит. В третьей строке «867 – 911» родитель – вообще мужчина, и эта строка нам не годится по этой причине. А, например, в шестой строке («911 – 1041») разница возрастов ребенка и родительницы равна $2010 - 1971 = 39$, и эту строку можно отметить (например, маркером).

Проделав эту операцию для всех строк таблицы Б, остается только подсчитать их количество.

Примечание. В других вариантах такой задачи, встречавшихся в тренажах, может потребоваться, например, указать количество детей, у которых отец старше матери на ука-

занное количество лет. Подобная задача решается аналогично, но для каждого ребенка, ID которого содержится в таблице Б, нужно определить (по одинаковому ID ребенка) идентификационные номера обоих его родителей, найти сведения о них в таблице А, определить (по графе «Пол»), кто из них отец, а кто мать, и сравнить их годы рождения.

Задача 6. Автомат обрабатывает четырёхзначное число (не начинающееся с нуля). По нему строится новое число по следующим правилам.

1. Складываются отдельно первая и вторая, вторая и третья, третья и четвертая цифры заданного числа.

2. Наименьшая из полученных трёх сумм отбрасывается.

3. Оставшиеся две суммы записываются друг за другом подряд в порядке неубывания.

Нужно определить **наименьшее** число, при обработке которого автомат выдаёт результат 613.

Решение

В целом задача аналогична ранее встречавшимся аналогам, но усложнена наличием предполагаемой неизвестной (отброшенной в ходе записи нового числа) наименьшей суммы цифр.

1. Число 613 можно разбить на два числа, которые могли бы являться суммами цифр, только одним способом: как 6 и 13 (и записаны эти суммы, как и требуется в условии, по неубыванию).

2. Итак, какие-то две соседние цифры исходного числа дают в сумме число 6, а другие две соседние цифры (среди которых, может быть, одна цифра – от предыдущей суммы) дают при сложении число 13. Нам же надо не просто найти подходящие цифры, но и такие, чтобы составить из них минимально возможное число.

Число 6 можно получить как $0 + 6$, $1 + 5$, $2 + 4$ или $3 + 3$.

Число 13 можно получить как $9 + 4$, $8 + 5$, $7 + 6$.

Чтобы новое число получалось наименьшим, нужно располагать «большие» цифры

правее. Значит, справа должна быть пара цифр, которая дает сумму 13: это пара 67, 58 или 49.

То, что давало нам сумму 6, попробуем расположить левее: 06, 15, 24 или 33.

Таким способом можно было бы получить различные четырехзначные числа как комбинации этих пар (кроме разве что использования крайней слева пары 06, так как с нуля число начинаться не может). Но мы можем и пойти на хитрость – ведь у нас есть две такие пары цифр: 06 и 67, которые можно «срассить» по одинаковой шестерке в трехзначное число (пока не смущаясь нулем слева) и получить еще меньшее число 067, которое дает обе требуемые суммы. (Другие подобные варианты «сращиваний»: 158 или 249 – больше чем 067, поэтому нам не подходят.)

3. Остается только «дорастить» полученное число до четырехзначного, добавив слева самую маленькую ненулевую цифру из возможных – единицу и получив число 1067.

Проверяем: попарное суммирование цифр этого числа дает нам суммы 1, 6 и 13, из которых наименьшую (1) мы отбрасываем и получаем требуемое нам число 613. А в том, что данное число меньше, чем все четырехзначные, которые можно было бы составить из ранее выписанных нами пар цифр, вы можете убедиться сами.

Ответ: 1067.

Комментарий к решению

А что, если требовалось бы найти не наименьшее, а **наибольшее** возможное исходное число?

В этом случае тоже потребуются сначала разделить заданное число 613 на два числа – суммы цифр (6 и 13), а затем расписать все возможные пары цифр, которые могут давать такие суммы.

Первой парой, очевидно, будет 94 (это максимально возможные цифры, которые нам «выгодно» записать как можно левее). После нее надо записать еще одну цифру – такую, чтобы она была максимально возможной, но сумма с предыдущей цифрой 4 была не больше 6 (иначе шестерку нам пришлось бы при вычислениях отбросить). Очевидно, что это цифра 2, и мы получаем уже

число 942, дающее при сложении соседних цифр искомые 13 и 6.

Остается дописать справа еще одну цифру – тоже максимально возможную, но дающую с последней полученной цифрой 2 сумму, не большую 6. Мы можем взять опять-таки четверку и получить число 9424.

Проверка: при попарном сложении цифр получаются суммы 13, 6 и 6, из которых можно отбросить любую шестерку на выбор и получить число 613, что нам и требовалось. Значит, 9424 – это и есть искомый ответ.

Задача 9. Фотокамера снимает растровые изображения размером 800×600 пикселей. При этом объём графического файла не может превышать 600 кб. Упаковка данных не производится. Какое максимальное количество цветов может быть в палитре?

Решение

1. Зная разрешение изображения, можно вычислить количество пикселей в нем: $800 \times 600 = 480000$.

2. Максимальный объём графического файла равен 600 кб – значит, на один пиксель может приходиться: $(600 \times 2^{13} \text{ бит}) / 480000 = 10 \text{ бит}$ (получаемое не целое частное 10,24 округляется до целого в *меньшую* сторону).

3. Этим 10 бит будет достаточно, чтобы закодировать $2^{10} = 1024$ цвета.

Ответ: 1024.

Примечание. Наиболее распространенная ошибка при решении такой задачи – округление получаемого дробного количества бит в большую сторону. Почему этого делать нельзя? Потому что, хотя мы обяза-

ны выполнить округление (количество битов не может быть дробным), если округлить значение в большую сторону, то при заданном числе пикселей мы тем самым увеличим получаемый объём графического файла сверх обозначенного максимума. Поэтому округление нужно выполнять до меньшего целого значения.

Задача 10. Все трёхбуквенные слова, составленные из букв К, О, Д, Е, Р, записаны в алфавитном порядке и пронумерованы, начиная с 1. Начало списка выглядит так:

1. ООО
2. ООК
3. ООД
4. ООР
5. ООЕ
6. ОКО

...

Под каким номером в списке идёт первое слово, которое начинается с буквы Д?

Решение

Подобные задачи тоже встречались ранее. Напомним, как выполняется их решение.

1. Очевидно, что приведенный в условии список слов идентичен трехзначным числам в пятеричной системе, где буквы соответствуют цифрам: О – 0, К – 1, Д – 2, Р – 3, Е – 4. Тогда нужно искать первое трехзначное число, начинающееся с цифры 2, то есть число 200_5 или, в десятичном представлении, 50.

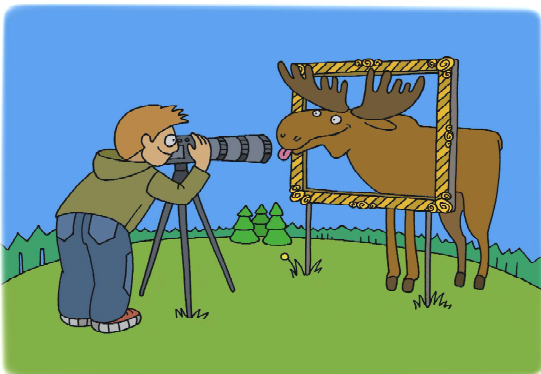
2. Заметим, что в списке число 000 идет под номером 1, – значит, номер каждого числа на 1 больше самого этого числа. Тогда искомое число (а значит, и слово) в списке стоит под номером 51.

Ответ: 51.

Задача 11. Имеется рекурсивный алгоритм F.

```

procedure F(n: integer);
begin
  if n > 0 then begin
    F(n - 3);
    F(n div 3);
    write(n)
  end
end;
    
```



Требуется записать подряд без пробелов и разделителей все числа, которые будут напечатаны на экране при выполнении вызова **F(9)**. Числа должны быть записаны в том же порядке, в котором они выводятся на экран.

Решение

1. Составляем таблицу вызовов функций (эти вызовы функций для наглядности также выделены цветом в тексте программы в условии задачи):

Исходный вызов	Генерируемые им вызовы	
	F(n - 3)	F(n div 3)
F(9)	F(6)	F(3)
F(6)	F(3)	F(2)
F(3)	F(0)	F(1)
F(2)	F(-1)	F(0)
F(1)	F(-2)	F(0)



2. Добавляем к этой таблице еще одну графу справа – для выводимых данных. Заполняем эту графу *снизу вверх*, при этом сначала записываются значения, выводимые генерируемыми вызовами (по порядку *слева направо*), а затем к ним дописывается значение, выводимое исходной функцией. Также не забываем, что выводятся только значения функций с $n > 0$ (табл. 2).

Получаемая в результате последовательность выводимых значений записана в первой ячейке правого столбца (для исходного вызова функции): **1326139**.

Ответ: 1326139.

Задача 13. При регистрации на сайте каждый пользователь получает индивидуальный идентификатор из 22 символов (используются только латинские буквы – 26 заглавных и 26 строчных). В базе данных для хранения сведений о каждом пользователе отводится одинаковое минимально возможное целое число байт, при этом каждый символ кодируется одинаковым минимально возможным количеством бит. Кроме идентификатора, для каждого пользователя в системе хранятся дополнительные сведения (25 байт на каждого пользователя).

Сколько байтов потребуется для хранения сведений о 30 пользователях?

Решение

1. Мощность алфавита составляет 52 буквы. Тогда для хранения каждой буквы достаточно 6 бит ($2^6 = 64$, а $2^5 = 32$ – слишком мало).

2. Для хранения идентификатора из 22 букв потребуется $22 \times 6 = 132$ бита или (с округлением в большую сторону) 17 байтов.

3. Вместе с дополнительными сведениями запись об одном пользователе занимает $17 + 25 = 42$ байта.

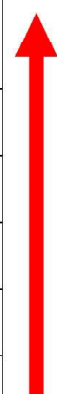
4. Для хранения записей о 30 пользователях тогда потребуется $30 \times 42 = 1260$ байтов.

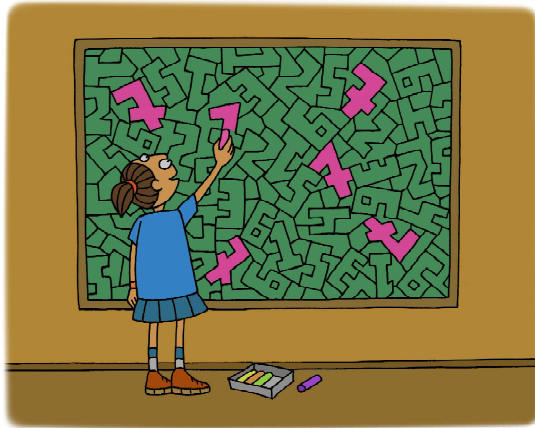
Ответ: 1260.

Задача 16. Значение арифметического выражения $64^{100} + 4^{200} - 16$ записано в системе счисления с основанием 8. Сколько в полученной записи будет цифр «7»?

Табл. 2

Исходный вызов	Генерируемые им вызовы		Выводимые значения	Комментарий
	F(n - 3)	F(n div 3)		
F(9)	F(6)	F(3)	1326 13 9	Выводятся значения для F(6), F(3) и для исходной функции
F(6)	F(3)	F(2)	13 2 6	Выводятся значения для F(3), F(2) и для исходной функции
F(3)	F(0)	F(1)	1 3	Выводятся значения для F(1) и для исходной функции
F(2)	F(-1)	F(0)	2	Выводится только значение для исходной функции
F(1)	F(-2)	F(0)	1	Выводится только значение для исходной функции





Решение

На первый взгляд, эта задача принципиально новая. Но, как нетрудно будет увидеть, она сводится к уже известной задаче на определение количества единиц.

1. Запишем все слагаемые как степени двойки по убыванию значений показателя степени:

$$64^{100} + 4^{200} - 16 = 2^{600} + 2^{400} - 2^4.$$

2. Слагаемое 2^{600} «создает» в двоичном числе одну единицу в разряде с номером 600.

Слагаемое 2^{400} аналогичным образом добавляет одну единицу в разряде с номером 400.

Получаем число, подобное показанному на рис. 1.

Остается выяснить, как меняется полученное двоичное число при вычитании 2^4 . Вспомним правила двоичного вычитания: при вычитании единицы из нуля результатом будет единица, а из расположенного левее более старшего раздела производится заём, то есть тоже вычитание единицы из нуля. И такая «цепочка» вычитаний-заёмов

будет продолжаться справа налево, пока очередным битом, из которого производится вычитание, не окажется бит с единицей. Тогда эта единица «обнуляется», а вычитания прекращаются. В нашем случае результат будет таким (см. рис. 2).

Таким образом, в этой задаче можно быстро вычислить количество единиц по правилу: если выражение имеет вид $2^{600} + 2^{400} - 2^4$, то первое слагаемое дает одну единицу, а последующая разность дает количество единиц, равное разности их показателей степени. То есть двоичная запись значения этого выражения содержит $1 + (400 - 4) = 397$ единиц. В прежних подобных задачах это число и было бы ответом.

3. Нас же спрашивают, сколько значение этого выражения будет содержать семёрок, если его записать в восьмеричной системе счисления².

Вспомним правило преобразования двоичного числа в восьмеричное «по триадам»: двоичное число справа налево разбивается на тройки («триады») двоичных цифр, а затем каждая такая триада заменяется на ее восьмеричное значение (как трехзначное двоичное число). Тогда можно определить, сколько в восьмеричной записи семёрок, подсчитав в ней количество триад единиц («111»).

В нашем случае отдельная первая единица гарантированно не дает такую триаду, так что ее мы исключаем из рассмотрения. А вот в полученной вычитанием длинной цепочке из 396 единиц можно выделить такие триады (то есть – нужные нам «семёрки»). И их количество будет равно результа-

600	599	...	402	401	400	399	398	...	6	5	4	3	2	1	0
1	0	...	0	0	1	0	0	...	0	0	0	0	0	0	0

Рис. 1

600	599	...	402	401	400	399	398	...	6	5	4	3	2	1	0
1	0	...	0	0	0	1	1	...	1	1	1	0	0	0	0

$$\underbrace{\hspace{10em}}_{\text{Кол-во единиц} = 400 - 4 = 396}$$

Рис. 2

² Решение предложено учащимся 11 класса Шаховым Иваном.

ту целочисленного деления количества единиц в цепочке на 3 (очевидно, что одна или две единицы, соответствующие остатку от такого деления, нужной триады не дают и потому отбрасываются).

Получаем³: $396 \text{ div } 3 = 132$.

Ответ: 132.

Задача 18. Сколько существует целых значений числа A , при которых формула

$$((x < 6) \rightarrow (x^2 < A)) \wedge ((y^2 \leq A) \rightarrow (y \leq 6))$$

тождественно истинна при любых целых неотрицательных x и y ?

Решение (1-й способ)

1. В заданном выражении самой последней выполняется логическая операция И. Поэтому для истинности всего выражения необходимо, чтобы обе соединяемые ей части выражения были истинными:

$$\begin{cases} ((x < 6) \rightarrow (x^2 < A)) = 1 \\ ((y^2 \leq A) \rightarrow (y \leq 6)) = 1 \end{cases}$$

При этом каждая из этих частей зависит только от одной переменной, так что их можно рассматривать по отдельности.

2. Рассматриваем первое логическое выражение: $((x < 6) \rightarrow (x^2 < A))$. Заменяем в нем операцию следования по правилу « $a \rightarrow b = \neg a \vee b$ »:

$$(x < 6) \rightarrow (x^2 < A) = \neg(x < 6) \vee (x^2 < A) = (x \geq 6) \vee (x^2 < A).$$

3. Начинаем рассмотрение с условия $(x \geq 6)$.

Всегда берем в качестве «опорного» условие без параметра A и смотрим, каким для его истинности/ложности должно тогда быть условие с A .

Если условие $(x \geq 6)$ истинно, то всё вышеприведенное выражение тоже истинно (операция ИЛИ). Поэтому такой случай мы не рассматриваем.

Если условие $(x \geq 6)$ ложно, то для истинности всего выражения условие $(x^2 < A)$ обязательно должно быть истинным.

4. Когда условие $(x \geq 6)$ ложно, будет истинным противоположное условие: $(x < 6)$.

Так как неравенство строгое, берем в качестве x ближайшее подходящее значение: $x = 5$.

Тогда, подставив найденное значение $x = 5$ во второе условие, получаем:

$$(x^2 < A) = (5^2 < A) = (25 < A) \text{ или } A > 25.$$

Это – первое граничное условие на значения A , которое, очевидно, определяет *минимальное* возможное значение этого параметра.

5. Рассматриваем второе логическое выражение: $((y^2 \leq A) \rightarrow (y \leq 6))$, тоже заменяя операцию следования на операцию ИЛИ: $(y^2 \leq A) \rightarrow (y \leq 6) = (y^2 > A) \vee (y \leq 6)$.

6. Начинаем рассмотрение с условия $(y \leq 6)$, которое не содержит параметра A .

Если условие $(y \leq 6)$ истинно, то всё вышеприведенное выражение тоже истинно (операция ИЛИ). Поэтому такой случай мы не рассматриваем.

Если условие $(y \leq 6)$ ложно, то для истинности всего выражения условие $(y^2 > A)$ обязательно должно быть истинным.

7. Когда условие $(y \leq 6)$ ложно, будет истинным противоположное условие: $(y > 6)$. Так как неравенство строгое, берем в качестве y ближайшее подходящее значение: $y = 7$.

Тогда, подставив найденное значение $y = 7$ во второе условие, получаем:

$$(y^2 > A) = (7^2 > A) = (49 > A) \text{ или } A < 49, \text{ что определяет } \textit{максимальное} возможное значение A .$$

8. Итак, для A мы получили интервал возможных значений $(25, 49)$, в котором обе границы в интервал *не входят* (соответствующие знаки неравенств – строгие). Остается определить количество целых чисел, входящих в этот интервал: $49 - 25 - 1 = 23$ (единица дополнительно вычитается, так как в интервал не входят *обе* границы).

Ответ: 23.

Решение (2-й способ)

1. Аналогично предыдущему способу решения, преобразуем исходное выражение в систему из двух логических выражений, которые оба должны быть истинными:

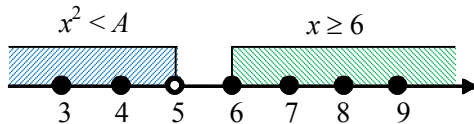
³ div – это обозначение операции целочисленного деления по стандарту языка программирования Паскаль. – Прим. авт.

$$\begin{cases} (x < 6) \rightarrow (x^2 < A) = 1, \\ (y^2 \leq A) \rightarrow (y \leq 6) = 1. \end{cases}$$

2. Рассмотрим первое из них, заменяя в нем операцию следования по правилу « $a \rightarrow b = \neg a \vee b$ »:

$$(x < 6) \rightarrow (x^2 < A) = \neg(x < 6) \vee (x^2 < A) = (x \geq 6) \vee (x^2 < A).$$

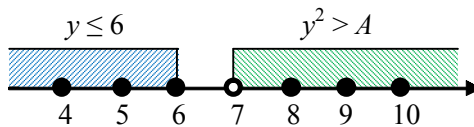
3. Изобразим оба соответствующих интервала на координатной прямой, не забывая при этом, что речь идет о **дискретном наборе целых чисел**. Сначала изображаем интервал, соответствующий условию $x \geq 6$ (значение 6 входит в интервал). Теперь нам нужно условием $x^2 < A$ покрыть оставшуюся часть координатной прямой – от значения 5 и меньше. Получаем, что $5^2 < A$, откуда $A > 25$.



4. Аналогичным образом преобразуем второе уравнение:

$$(y^2 \leq A) \rightarrow (y \leq 6) = \neg(y^2 \leq A) \vee (y \leq 6) = (y^2 > A) \vee (y \leq 6).$$

5. Изобразим эти два интервала на такой же «дискретной» координатной прямой. Сначала изображаем интервал для условия $y \leq 6$, а затем покрываем вторым условием оставшуюся часть чисел от 7 и больше. Получаем, что $7^2 > A$, откуда $A < 49$.



6. Таким образом, для A мы получили интервал возможных значений $(25, 49)$, в котором обе границы в интервал не входят (соответствующие знаки неравенств – строгие). Определим количество целых чисел, входящих в этот интервал: $49 - 25 - 1 = 23$.

Ответ: 23.

Есть и другая вариация данной задачи, в которой квадрат переменных сочетается не с параметром A , а с константой. Решение в этом случае получается чуть посложнее, но в целом рассуждения аналогичны.

Задача 18. Сколько существует целых значений числа A , при которых формула $((x < A) \rightarrow (x^2 < 100)) \wedge ((y^2 \leq 64) \rightarrow (y \leq A))$ тождественно истинна при любых целых неотрицательных x и y ?

Решение

1. Две операции следования соединены операцией И, тогда должны быть истинными оба эти следования:

$$\begin{cases} (x < A) \rightarrow (x^2 < 100) = 1, \\ (y^2 \leq 64) \rightarrow (y \leq A) = 1. \end{cases}$$

Избавившись от операции следования, получаем систему уравнений:

$$\begin{cases} (x \geq A) \vee (x^2 < 100) = 1, \\ (y^2 > 64) \vee (y \leq A) = 1. \end{cases}$$

2. Рассмотрим первое уравнение $(x \geq A) \vee (x^2 < 100) = 1$. Рассмотрение начинаем с условия $(x^2 < 100)$, которое не содержит параметра.

Если условие $(x^2 < 100)$ истинно, то всё вышеприведенное выражение тоже истинно (операция ИЛИ). Поэтому такой случай мы не рассматриваем.

Если условие $(x^2 < 100)$ ложно, то для истинности всего выражения условие $(x \geq A)$ обязательно должно быть истинным.

3. Когда условие $(x^2 < 100)$ ложно, будет истинным противоположное условие: $(x^2 \geq 100)$. Тогда $x = 10$ (неравенство нестрогое).

Подставив найденное значение $x = 10$ во второе условие, получаем:

$(x \geq A) = (10 \geq A)$ или $A \leq 10$. Это *верхняя граница* допустимого интервала параметра.

4. Рассмотрим второе уравнение $(y^2 > 64) \vee (y \leq A) = 1$. Рассмотрение начинаем с условия $(y^2 > 64)$, которое не содержит параметра.

Если условие $(y^2 > 64)$ истинно, то всё вышеприведенное выражение тоже истинно (операция ИЛИ). Поэтому такой случай мы не рассматриваем.

Если условие $(y^2 > 64)$ ложно, то для истинности всего выражения условие $(y \leq A)$ обязательно должно быть истинным.

5. Когда условие $(y^2 > 64)$ ложно, будет истинным противоположное условие: $(y^2 \leq 64)$. Тогда $y = 8$ (неравенство нестрогое).

Подставив найденное значение $y = 8$ во второе условие, получаем:
 $(y \leq A) = (8 \leq A)$ или $A \geq 8$. Это нижняя граница допустимого интервала параметра.

6. Получили интервал значений $[8, 10]$, где обе границы *входят в интервал* (соответствующие знаки неравенств – нестрогие). Поэтому количество подходящих целых значений A вычисляется как $10 - 8 + 1 = 3$.

Ответ: 3.

Задача 20. Считав число x , приведенная ниже программа печатает два числа: a и b . Укажите наименьшее из чисел x , при вводе которого программа напечатает сначала 2, а потом 4.

```
var x, a, b: longint;
begin
  readln(x);
  a := 0;
  b := 0;
  while x > 0 do begin
    if x mod 2 = 0 then a := a + 1
    else b := b + x mod 10;
    x := x div 10;
  end;
  writeln(a);
  write(b);
end.
```

Решение

1. Анализируем программу:

– цикл с условием **while x > 0 do** и выполняемый в цикле оператор **x := x div 10** – это хорошо знакомый учащимся алгоритм «разбора» числа на отдельные цифры справа налево (от младших разрядов к старшим);

– операция **if x mod 2 = 0 then a := a + 1** при помощи **mod** определяет, является ли последняя цифра числа x четной, и если да, то счетчик a увеличивается на 1, то есть в переменной a определяется количество четных цифр в числе x ;

– если же последняя цифра числа нечетна, то срабатывает ветвь **else**, и в переменной b вычисляется сумма таких нечетных цифр.

2. Если программа выводит сначала число 2, а затем число 4, то это означает, что в числе x было две четные цифры (причем



нули тоже считаются четными цифрами), но о количестве нечетных цифр ничего не говорится (!), а сумма нечетных цифр равна 4 (но четные цифры в подсчет не входят!).

3. Нам нужно найти наименьшее такое число. Значит, требуется:

– минимально возможное количество нечетных цифр, сумма которых равна 4;

– две четные, но наименьшие из возможных цифры (допустимо – два нуля, с условием, что они не могут стоять в начале числа);

– наибольшие цифры в числе нужно размещать правее.

4. Сумму 4 надо представить как сложение двух нечетных цифр, – это можно сделать только одним способом: $1 + 3$.

5. Тогда наименьшее возможное число – 1003.

Ответ: 1003.

Комментарий к решению

А если требовалось бы найти наибольшее возможное число?

Тогда для этого нужно:

– максимально возможное количество нечетных цифр, сумма которых равна 4;

– две четные, но наибольшие из возможных цифры;

– наибольшие цифры надо в числе размещать левее.

Число 4 в данном случае можно разложить на нечетные цифры так: $1 + 1 + 1 + 1$. А наибольшие возможные четные цифры – это 88. Тогда искомое число – 881111.

Еще одна интересная разновидность задачи 20 была встречена в одном из тренажных вариантов. Сама по себе она не сложна, но имеет «подковырку», с которой справились далеко не все школьники.

Задача 20. Получив на вход число x , программа печатает два числа a и b . Укажите наименьшее из таких чисел x , при вводе которых программа напечатает сначала число 3, а потом число 16.

```
var x, a, b: integer;
begin
  readln(x);
  a:=0;
  b:=1;
  while x>0 do begin
    a := a+1;
    b := b + (10 - (x mod 10));
    x := x div 10;
  end;
  writeln(a);
  write(b);
end.
```

Решение

1. Проанализировав текст программы, нетрудно понять, что алгоритм обработки в ней заложен самый простой: исходное число раскладывается на десятичные цифры; переменная a – это счетчик количества проходов цикла, то есть значение a определяет, сколько цифр было в исходном числе; переменная же b – это сумма... но чего именно?

2. Раньше в таких задачах суммировались получаемые цифры. Теперь же к b прибавляется значение $(10 - (x \text{ mod } 10))$, которое, очевидно, представляет собой дополнение очередной цифры исходного числа до 10. То есть, например, если очередная цифра равна 3, то к b прибавляется число 7.

3. По условию, программа выводит сначала число 3, а затем – число 16. Значит, после выполнения программы $a = 3$, а $b = 16$. Следовательно, исходное число содержало 3 цифры. Но как определить эти цифры по заданному значению b ?

Пусть искомые три цифры равны, соответственно, x , y и z (считая справа налево, в том порядке, в каком они выделяются из

числа). Тогда получаемое значение b равно: $b = (10 - x) + (10 - y) + (10 - z) = 10 + 10 + 10 - x - y - z = 30 - (x + y + z)$, откуда искомым сумму цифр $(x + y + z)$ можно определить как $30 - b$ (получаемое при работе программы значение b нужно вычесть из 10, умноженного на количество цифр в числе, то есть на значение a). Итоговая формула: *сумма цифр* $= 10 \times a - b$.

4. Следовательно, надо найти минимально возможное трехзначное число, сумма цифр которого равна $30 - 16 = 14$. Удобнее всего разложить эту сумму на две цифры, из которых одна наименьшая из возможных, то есть как $9 + 5$, и записать эти цифры по возрастанию (большая цифра – слева). А между ними добавить в качестве третьей цифры минимально возможную цифру 0.

Ответ: 509.

Задача 21. Найти наибольшее значение входной переменной k , при котором программа выдаёт ответ 9.

```
var k, i : longint;
function f(n: longint): longint;
begin
  f := n * n * n;
end;
begin
  readln(k);
  i := 20;
  while f(i) > k do
    i := i-1;
  writeln(i)
end.
```

Решение

Такие задачи тоже нам уже знакомы. Но раньше требовалось искать значения переменной k , такие же, какие программа выдает при некотором заданном k . А теперь ответ программы приведен в условии явно. Как мы увидим далее, это изменение заметно упрощает решение.

1. Используемая функция f соответствует n^3 .

2. Цикл прекращает выполняться, когда его условие будет нарушено, то есть при $f(i) \leq k$. Цикл выполнялся, когда условие было еще истинным, но при предыдущем

значении цикловой переменной, то есть при $i + 1$. Получаем неравенство: $f(i + 1) > k$. Соединяем эти два неравенства в одно двойное по одинаковому k (второе неравенство для этого потребуются «зеркально развернуть», меняя его знак): $f(i) \leq k < f(i + 1)$.

3. Из текста программы мы видим, что выводится значение i , полученное сразу после завершения цикла, то есть как раз то самое, которое мы рассматриваем в нашем неравенстве. И оно равно 9.

4. Подставляем в неравенство запись функции f (в данном случае $-i^3$) и заданное значение i :

$$i^3 \leq k < (i + 1)^3 \Rightarrow 9^3 \leq k < (9 + 1)^3 \Rightarrow 9^3 \leq k < 10^3 \Rightarrow 729 \leq k < 1000 \Rightarrow k \in [729, 1000).$$

Заметим, что верхняя граница в интервал не входит (соответствующий знак неравенства – строгий!).

5. Нам требуется наибольшее возможное значение k . Поскольку значение 1000 в найденный интервал не входит, мы берем предыдущее значение $k = 999$.

Ответ: 999.

Комментарий к решению

Заметим: так как нам требовалось только наибольшее значение, определяемое верхней границей интервала, нижнюю границу (9^3) можно было бы не вычислять, сэкономив время.

Задача 23. Сколько существует различных наборов значений логических переменных $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, которые удовлетворяют всем перечисленным условиям?

$$\begin{aligned} (x_1 \rightarrow x_2) \wedge (x_1 \rightarrow y_1) &= 1 \\ (x_2 \rightarrow x_3) \wedge (x_2 \rightarrow y_2) &= 1 \\ &\dots \\ (x_7 \rightarrow x_8) \wedge (x_7 \rightarrow y_7) &= 1 \\ (x_8 > y_8) &= 1 \end{aligned}$$

В ответе **не нужно** перечислять все различные наборы значений переменных $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$, при которых выполнена данная система равенств. В качестве ответа нужно указать только количество таких наборов.

Решение

Такие задачи – это «высший пилотаж» первой части ЕГЭ. Их умеют решать не все, и даже несколько обидно, что за правильное решение «двадцать третьей» задачи дается всего один «сырой» балл, а не два или три, как для задач с номерами 24, 25, 26 и 27.

1. Рассмотрим первое уравнение: $(x_1 \rightarrow x_2) \wedge (x_1 \rightarrow y_1) = 1$. Из него следует, что обе операции следования должны быть истинными (равны 1):

$$\begin{cases} (x_1 \rightarrow x_2) = 1; \\ (x_1 \rightarrow y_1) = 1. \end{cases}$$

2. Операция следования истинна в трех случаях: $0 \rightarrow 0, 0 \rightarrow 1, 1 \rightarrow 1$. То есть из единицы всегда должна следовать единица, а из нуля может следовать что угодно (варианты «размножаются»). Тогда, учитывая, что все «иксы» следуют друг из друга, можно по всем имеющимся уравнениям сразу построить таблицу возможных значений для всех операций следования «с иксами» в виде:

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1
0	0	0	1	1	1	1	1
0	0	0	0	1	1	1	1
0	0	0	0	0	1	1	1
0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0

Полезно запомнить, как выглядит эта матрица значений переменных для цепочки следований, – это еще не раз вам пригодится!

Согласно этой таблице, для истинности всех операций следования «с иксами» существует ровно 9 возможных наборов значений переменных x (столько, сколько строк в полученной таблице).

3. Кроме уже рассмотренных операций «с иксами», у нас в систему уравнений входят операции следования вида $(x_i \rightarrow y_i)$ для всех $i = 1 \dots 8$, которые должны быть тоже истинными. Их истинность зависит от найденных значений переменных x . При этом каждое y_i зависит от значения «одноимен-

Табл. 5

Обе части уравнения	x_1	x_2	y_1	y_2
1	1	1	1	1
	0	0	1	1
0	1	0	1	0
	0	0	0	1

Обе части уравнения	x_2	x_3	y_2	y_3	Кол-во наборов
1	1	1	1	1	2
	0	0	0	0	2
0	1	0	1	0	2
	0	1	0	1	2

2. Для всех уравнений таблицы допустимых значений однотипны. Повторяем для второго уравнения ($x_2 \equiv x_3 \equiv (y_2 \equiv y_3) = 1$) ту же таблицу, добавив справа столбец количества наборов переменных с учетом предыдущей таблицы.

Чтобы определить эти количества наборов переменных, в каждой строке новой таблицы (табл. 5) нужно посмотреть значения x_2 и y_2 (которыми вторая таблица «связана» с первой) и записать, сколько в предыдущей таблице имеется строк с такими значениями x_2 и y_2 .

Например:

– для случая $x_2 = 1$ и $y_2 = 1$ в первой таблице есть две строки (самая первая и самая последняя), – значит, во второй таблице во всех строках, где $x_2 = 1$ и $y_2 = 1$, запишем количество вариантов, равное 2;

– для случая $x_2 = 1$ и $y_2 = 0$ в первой таблице есть тоже две строки (вторая и предпоследняя), – значит, во второй таблице во всех строках, где $x_2 = 1$ и $y_2 = 0$, тоже пишем количество 2;

– аналогично поступаем и для остальных сочетаний x_2 и y_2 (количество вариантов получается везде по два).

Делаем вывод: второе уравнение **удваивает** число вариантов, имевшихся в первом.

3. Аналогично можно составить таблицу для третьего уравнения ($x_3 \equiv x_4 \equiv (y_3 \equiv y_4) = 1$).

Чтобы определить количества наборов переменных, в каждой строке таблицы 6 нужно посмотреть значения x_3 и y_3 , которыми третья таблица «связана» со второй, и записать, сколько в предыдущей таблице имеется вариантов с такими значениями x_3 и y_3 ,

Табл. 6

Обе части уравнения	x_3	x_4	y_3	y_4	Кол-во наборов
1	1	1	1	1	4
			0	0	4
0	0	0	1	1	4
			0	0	4
0	1	0	1	0	4
			0	1	4
0	0	1	1	0	4
			0	1	4

помня, что во второй таблице каждая строка уже содержит некоторое количество вариантов (в нашем случае – по 2).

Делаем вывод: третье уравнение тоже удваивает число вариантов, имевшихся во втором.

4. Обобщив этот вывод, мы можем предположить, что каждое последующее уравнение удваивает число вариантов предыдущего

по каждой строке таблицы, и сразу записать таблицы (см. табл. 7) для оставшихся уравнений.

5. В последней полученной таблице остается только просуммировать количества наборов по всем строкам, тогда общее количество наборов переменных равно $64 \times 8 = 512$.

Ответ: 512.

Табл. 7

Обе части уравнения	x_4	x_5	y_4	y_5	Кол-во наборов
1	1	1	1	1	8
			0	0	8
	0	0	1	1	8
			0	0	8
0	1	0	1	0	8
			0	1	8
	0	1	1	0	8
			0	1	8

Обе части уравнения	x_5	x_6	y_5	y_6	Кол-во наборов
1	1	1	1	1	16
			0	0	16
	0	0	1	1	16
			0	0	16
0	1	0	1	0	16
			0	1	16
	0	1	1	0	16
			0	1	16

Обе части уравнения	x_6	x_7	y_6	y_7	Кол-во наборов
1	1	1	1	1	32
			0	0	32
	0	0	1	1	32
			0	0	32
0	1	0	1	0	32
			0	1	32
	0	1	1	0	32
			0	1	32

Обе части уравнения	x_7	x_8	y_7	y_8	Кол-во наборов
1	1	1	1	1	64
			0	0	64
	0	0	1	1	64
			0	0	64
0	1	0	1	0	64
			0	1	64
	0	1	1	0	64
			0	1	64

Задача 25. Дан массив, содержащий положительные целые числа, не превышающие по значению 32768. Программа должна определить количество элементов этого массива, десятичная и восьмеричная запись которых содержат одинаковое количество цифр.

Исходные данные объявлены, как показано в тексте ниже. Запрещается использовать не описанные в нем переменные, но можно использовать не все описанные переменные.

```
const N = 5000;
var a: array [1..N] of integer;
    b, i, k, l, m: integer;
begin
  for i:=1 to N do
    readln(a[i]);
  ...
end.
```

Решение

В прошлые годы на ЕГЭ уже попадались задания, в которых требовалось сформулировать в программе условие, определяющее, что в анализируемом числе (элементе массива) содержится заданное количество цифр – например три. Такое условие обычно записывается в операторе `if` в виде сравнений с граничными значениями – наименьшим и наибольшим числами с нужным количеством цифр в соответствующей системе счисления (см., например, в статье: Богомолова О. Б., Усенков Д. Ю. Программирование: недесятичные системы счисления // Компьютерные инструменты в школе. 2017. № 3).

Следовательно, для выявления числа с некоторым заданным количеством цифр в конкретной системе счисления требуется записать два граничных условия. Нам же требуется определять и сравнивать *все* возможные количества цифр (от одной до пяти – в соответствии с указанным максимально возможным значением элементов массива) и в двух системах счисления – десятичной и восьмеричной. Тогда потребуется в `if` записывать по два граничных условия для каждого количества цифр и для каждой системы счисления, согласно схеме:



Если ((одна десятичная цифра) И (одна восьмеричная цифра)) ИЛИ ((две десятичные цифры) И (две восьмеричные цифры)) ИЛИ ((три десятичные цифры) И (три восьмеричные цифры)) ИЛИ ((четыре десятичные цифры) И (четыре восьмеричные цифры)) ИЛИ ((пять десятичных цифр) И (пять восьмеричных цифр)).

Итого – 20 операций сравнения! Получается слишком громоздко!

Но ведь в других задачах – например в задании 20, – мы уже встречали алгоритм, позволяющий разбивать число на цифры в соответствующей системе счисления и подсчитывать количество таких цифр! Вот им-то мы и воспользуемся для решения данной задачи.

Приведем требуемый листинг с соответствующими построчными комментариями (см. листинг 1).

Комментарий к решению

Если в используемом языке программирования есть операции / функции / методы для работы с символьными и строковыми данными, – в частности, для преобразования числа в его символьную запись в требуемой системе счисления и для подсчета длины текстовой строки, то можно воспользоваться ими для решения данной задачи. Но, например, на языке Паскаль такие функции предусмотрены только для работы с десятичной системой счисления, а значит, реализовать решение таким способом не удастся.

Листинг 1

<pre>k := 0;</pre>	<p>Счетчик искомого количества чисел, в которых одинаковы количества цифр в десятичной и восьмеричной системе счисления</p>
<pre>for i:=1 to N do begin b := a[i];</pre>	<p>Цикл перебора всех элементов массива Нужно сохранить копию значения текущего элемента массива, так как наш алгоритм подсчета количества цифр «разрушает» число</p>
<p style="text-align: center;"><i>Определяем количество десятичных цифр</i></p> <pre>m := 0; while b > 0 do begin m := m+1; b := b div 10 end;</pre>	<p>Счетчик количества десятичных цифр Цикл подсчета количества цифр в числе Увеличиваем количество цифр на 1 Отбрасываем последнюю десятичную цифру числа Конец цикла подсчета количества цифр</p>
<pre>b := a[i];</pre> <p style="text-align: center;"><i>Определяем количество восьмеричных цифр</i></p> <pre>l := 0; while b > 0 do begin l := l+1; b := b div 8 end;</pre>	<p>Сохраняем копию значения текущего элемента массива Счетчик количества восьмеричных цифр Цикл подсчета количества цифр в числе Увеличиваем количество цифр на 1 Отбрасываем последнюю восьмеричную цифру числа Конец цикла подсчета количества цифр</p>
<p style="text-align: center;"><i>Зная количества десятичных и восьмеричных цифр, сравниваем их</i></p> <pre>if l = m then k := k+1</pre>	<p>Если количества десятичных и восьмеричных цифр равны, увеличиваем на 1 счетчик количества элементов массива Конец цикла перебора элементов массива</p>
<pre>end;</pre>	
<pre>writeln(k)</pre>	<p>Вывод результата</p>



Богомолова Ольга Борисовна,
доктор педагогических наук,
почетный работник сферы
образования Российской Федерации,
учитель информатики
и математики ГБОУ СОШ № 1360,
г. Москва,

Усенков Дмитрий Юрьевич,
Московский государственный
институт индустрии туризма
имени Ю.А. Сенкевича, г. Москва.