



## ЕГЭ-2013: что год грядущий нам готовит?

**О.Б. Богомолова,**  
д. п. н., учитель  
информатики  
и математики  
ГОУ СОШ № 1360,  
Восточный округ  
г. Москвы

**Д.Ю. Усенков,**  
ст. н. с. Института  
информатизации  
образования  
Российской академии  
образования,  
Москва

▶ Начался очередной учебный год, начались и тренировочные работы по подготовке к ГИА и ЕГЭ. Очередная тренировочно-диагностическая работа по типу ЕГЭ откровенно удивила и учителей, и школьников. Прежде всего изменилась технология ее выполнения: не на бумаге, а непосредственно на сайте ФИПИ через Интернет. Соответственно, поскольку в процессе выполнения работы учащийся имеет в своих руках компьютер, целый ряд заданий тренировочно-диагностической работы стал носить сугубо практический характер и проверять навыки работы с программными средствами — электронными таблицами и текстовыми редакторами (нововведение вполне ожидаемое в духе нового ФГОС и планируемого перехода к компьютерной форме сдачи ЕГЭ). И наконец, как и в прошлые годы, появился целый ряд новых заданий “теоретического” содержания.

К сожалению, организаторы тренировочно-диагностической работы запрещают копирование заданий. Поэтому ниже рассмотрены авторские задачи, подобные по характеру предлагавшимся в ходе тренировочно-диагностической работы (можно предполагать, что такие задачи с большой вероятностью будут предложены и на ЕГЭ-2013).

### Задача А4

Информация для ее передачи по каналу связи кодируется шестизначными двоичными кодами. Поскольку при их передаче возможны искажения из-за помех, в конце каждого такого кода (справа) добавляется седьмой (контрольный) разряд. Значение этого контрольного разряда выбирается так, чтобы сумма разрядов полученного семиразрядного кода была четной. Например, к коду 111010 справа добавляется 0, а к коду 111000 — 1. В приемнике каждый принятый код обрабатывается следующим образом. Вычисляется сумма его семи разрядов (включая контрольный). Нечетная сумма показывает, что при передаче данного кода произошел сбой, и этот код автоматически заменяется на специальный код 0000000. При четной сумме принятый код не изменяется.

Исходное сообщение 0110000 1000111 0001111 было принято в виде 0111001 1000101 0001111. Каким будет принятое сообщение после его обработки?

- 1) 0111001 1000101 0001111
- 2) 0000000 0000000 0001111
- 3) 0111001 0000000 0001111
- 4) 0110000 1000111 0001111

*Решение*

Сопоставим каждый код исходного сообщения с соответствующим ему кодом в полученном сообщении:

Передано:	0110000	1000111	0001111
Получено:	0111001	1000101	0001111

Проанализируем каждую из этих пар кодов (переданный — полученный) с точки зрения описанного в условии задачи алгоритма обработки:

1. **0110000** ⇒ **0111001**. В коде при его передаче возникло два искажения — 0 изменился на 1 в нулевом и в третьем разрядах (напомним, что в двоичных числах разряды принято нумеровать справа налево начиная с нуля; разряды с искажениями мы будем выделять красным цветом).

Поскольку в полученном коде сумма всех разрядов (или, что то же самое, количество единиц в числе) четна, система обработки принятых кодов “не заметит” такого двойного искажения кода, поэтому данный код будет принят в качестве “правильного” и оставлен в виде **0111001**.

2. **1000111** ⇒ **1000101**. В этом коде при его передаче искаженным оказался только один, первый разряд (вместо 1 в нем появился 0). В результате сумма всех разрядов стала нечетной. Поэтому, согласно описанному в условии задачи алгоритму обработки, система заменит такой код на специальный код **0000000**.

3. **0001111** ⇒ **0001111**. Поскольку этот код передан без искажений и сумма всех его разрядов осталась четной, система обработки оставит этот код без изменений: **0001111**.

В результате после обработки мы получим следующую последовательность кодов:

**0111001 0000000 0001111**

*Ответ:* 0111001 0000000 0001111 (вариант ответа № 3).

### Методический комментарий

Описываемый в этой задаче способ отслеживания возможных ошибок при передаче двоичных кодов, по сути, представляет собой контроль ошибок при помощи *бита четности*. Однако этот метод позволяет отследить ошибку только в одном бите передаваемого кода (что и продемонстрировано в решении данной задачи для первого из трех кодов). Для отслеживания большего количества ошибок в передаваемом коде используется более совершенный метод, который носит название “код Хэмминга”.

### Задача А5

Выполняется запись (оцифровка) стереозвука с частотой дискретизации 16 кГц и 32-битным разрешением. Размер получаемого аудиофайла ограничен (не должен превышать 10 Мб). Сжатие данных не производится.

Какое из значений наиболее близко к максимально возможной длительности записываемого аудиофрагмента?

- 1) 30 секунд
- 2) 60 секунд
- 3) 90 секунд
- 4) 120 секунд

*Решение*

Решение данной задачи незначительно отличается от аналогичных задач про кодирование звука, предлагавшихся в предыдущие годы. Только если раньше речь шла о вычислении объема файла, соответствующего звукозаписи определенной длительности, то теперь нужно, наоборот, вычислить длительность звукозаписи по заданному предельному размеру файла.

Итак:

- разрешение составляет 32 бита, т.е. каждый замер громкости звукового сигнала соответствует 32 битам;

- частота дискретизации 16 кГц соответствует 16 тысячам измерений в секунду (т.е. 16 тысячам 32-битных слов в секунду);

- звук стереофонический, поэтому объем записываемой информации удваивается.

Тогда одной секунде оцифровываемого с такими параметрами звука соответствует объем файла  $32 \times 16\,000 \times 2 = 2^5 \times 2^7 \times 125 \times 2 = 125 \times 2^{13}$ . (По возможности имеет смысл выделять в произведении степени двойки, чтобы облегчить дальнейшие расчеты.)

Максимальный объем файла по условию задачи равен 10 Мб. Поскольку ранее речь шла о битах, необходимо перевести это значение также в биты:

$$10 \text{ (Мб)} = 5 \times 2 \text{ (Мб)} = 5 \times 2 \times 2^{20} \text{ (байт)} = 5 \times 2 \times 2^{20} \times 2^3 \text{ (бит)} = 5 \times 2^{24} \text{ (бит)}.$$

Тогда длительность звукозаписи, соответствующую этому объему файла, можно вычислить так:

$$5 \times 2^{24} \text{ (бит)} / 125 \times 2^{13} \text{ (бит/с)} = 2^{11} / 25 \text{ (с)} = 81,92 \text{ (с)} \approx 90 \text{ с}.$$

*Ответ:* 90 секунд (вариант ответа № 3).

### Задача А10

Дан фрагмент алгоритма.

Переменные  $a, b, c$  имеют строковый тип, а переменные  $i, k$  — целый тип.

Используются функции:

**Длина( $a$ )** — возвращает количество символов в строке  $a$  (целое значение);

**Извлечь( $a, i$ )** — возвращает символ строки  $a$ , стоящий на  $i$ -м месте (позиции символов нумеруются слева направо начиная с 1);

**Склеить**( $a, b$ ) — возвращает строку, в которой сначала записаны все символы строки  $a$ , а затем все символы строки  $b$ .

Строковые значения записываются в апострофах (например,  $a = \text{'строка'}$ ).

Конструкция **ПОКА** <условие> {<операторы>} представляет собой цикл с предусловием. Операторы в теле цикла выполняются, пока условие цикла истинно.

```
i = Длина(a)
k = 3
b = 'X'
ПОКА i > 0
{
c = Извлечь(a, i)
b = Склеить(b, c)
i = i - k
}
b := Склеить(b, 'X')
```

Какое значение будет иметь переменная  $b$  после выполнения этого фрагмента алгоритма, если изначально переменная  $a$  имела значение 'РОБОТОТЕХНИКА'?

*Решение*

Символьные данные в задачах ЕГЭ рассматриваются не впервые, однако в течение последних 3–4 лет разработчики заданий Единого госэкзамена не уделяли им внимания и сосредоточивались исклю-

чительно на работе с числовыми данными. Очевидно, пришла пора при изучении программирования разбирать с учащимися и задания с символьными и строковыми данными.

Метод решения такой задачи тот же, что и для ранее предлагавшихся на ЕГЭ задач с циклами: построение таблицы трассировки (ячейки, в которых при выполнении данного оператора поменялось значение, будем выделять жирным шрифтом и фоновой закраской).

Ответ: 'ХАНТОРХ'.

**Методический комментарий**

В целом работа с символьными данными ненамного сложнее обработки числовой информации, хотя и имеет свою специфику. Для освоения этой области программирования можно порекомендовать статью: Усенков Д.Ю., Богомолова О.Б. “О знаках и строках замолвите слово...”, или Несколько занимательных задач на работу со строками и множествами” // Информатика, 2011, № 16. С. 32–47.

**Задача В6**

Строки (цепочки символов латинских букв) создаются по следующему правилу.

Первая строка состоит из одного символа — латинской буквы “А”. Вторая строка состоит из двух

Оператор	$i$	$k$	$a$	$b$	$c$
$i = \text{Длина}(a)$	<b>13</b>		'РОБОТОТЕХНИКА'		
$k = 3$	13	<b>3</b>	'РОБОТОТЕХНИКА'		
$b = \text{'X'}$	13	3	'РОБОТОТЕХНИКА'	<b>'X'</b>	
<b>ПОКА</b> $i > 0$ (цикл выполняется)	13 (>0)	3	'РОБОТОТЕХНИКА'	'X'	
$c = \text{Извлечь}(a, i)$	13	3	'РОБОТОТЕХНИКА'	'X'	<b>'А'</b>
$b = \text{Склеить}(b, c)$	13	3	'РОБОТОТЕХНИКА'	<b>'ХА'</b>	'А'
$i = i - k$	<b>10</b>	3	'РОБОТОТЕХНИКА'	'ХА'	'А'
<b>ПОКА</b> $i > 0$ (цикл выполняется)	10 (>0)	3	'РОБОТОТЕХНИКА'	'ХА'	'А'
$c = \text{Извлечь}(a, i)$	10	3	'РОБОТОТЕХНИКА'	'ХА'	<b>'Н'</b>
$b = \text{Склеить}(b, c)$	10	3	'РОБОТОТЕХНИКА'	<b>'ХАН'</b>	'Н'
$i = i - k$	<b>7</b>	3	'РОБОТОТЕХНИКА'	'ХАН'	'Н'
<b>ПОКА</b> $i > 0$ (цикл выполняется)	7 (>0)	3	'РОБОТОТЕХНИКА'	'ХАН'	'Н'
$c = \text{Извлечь}(a, i)$	7	3	'РОБОТОТЕХНИКА'	'ХАН'	<b>'Т'</b>
$b = \text{Склеить}(b, c)$	7	3	'РОБОТОТЕХНИКА'	<b>'ХАНТ'</b>	'Т'
$i = i - k$	<b>4</b>	3	'РОБОТОТЕХНИКА'	'ХАНТ'	'Т'
<b>ПОКА</b> $i > 0$ (цикл выполняется)	4 (>0)	3	'РОБОТОТЕХНИКА'	'ХАНТ'	'Н'
$c = \text{Извлечь}(a, i)$	4	3	'РОБОТОТЕХНИКА'	'ХАНТ'	<b>'О'</b>
$b = \text{Склеить}(b, c)$	4	3	'РОБОТОТЕХНИКА'	<b>'ХАНТО'</b>	'О'
$i = i - k$	<b>1</b>	3	'РОБОТОТЕХНИКА'	'ХАНТО'	'О'
<b>ПОКА</b> $i > 0$ (цикл выполняется)	1 (>0)	3	'РОБОТОТЕХНИКА'	'ХАНТО'	'Н'
$c = \text{Извлечь}(a, i)$	1	3	'РОБОТОТЕХНИКА'	'ХАНТО'	<b>'Р'</b>
$b = \text{Склеить}(b, c)$	1	3	'РОБОТОТЕХНИКА'	<b>'ХАНТОР'</b>	'Р'
$i = i - k$	<b>-2</b>	3	'РОБОТОТЕХНИКА'	'ХАНТОР'	'Р'
<b>ПОКА</b> $i > 0$ (цикл прекращается)	-2 (<0)	3	'РОБОТОТЕХНИКА'	'ХАНТОР'	'Р'
$b = \text{Склеить}(b, \text{'X'})$	-2	3	'РОБОТОТЕХНИКА'	<b>'ХАНТОРХ'</b>	'Р'

символов — латинских букв “BC”. Цепочка под номером  $n$  (для  $n > 2$ ) формируется следующим образом: сначала записывается цепочка под номером  $n - 1$ , затем справа к ней дописывается цепочка под номером  $n - 2$ , а затем справа дописывается снова цепочка под номером  $n - 1$ .

Вот первые 4 строки, созданные по этому правилу:

- (1) A
- (2) BC
- (3) BCABC
- (4) BCABCBCABC

Сколько символов, отличных от “B”, находится в восьмой строке?

*Решение*

Снова на ЕГЭ появляются забытые было цепочки...

Конечно, можно решить такую задачу “в лоб” — расписывая все получаемые цепочки вплоть до восьмой и затем подсчитав в ней количество символов, кроме “B”. Однако лучше использовать более рациональный способ: попытаться найти закономерность и вывести формулу зависимости искомого значения от номера цепочки или хотя бы построить соответствующую таблицу (т.е. свести решение к формальному).

Проанализируем алгоритм, по которому формируются строки.

Очевидно, что ни в какой строке не будет содержаться никаких символов, кроме “A”, “B” и “C”, поскольку новые строки получаются лишь копированием предыдущих.

Общая закономерность “размножения” букв: для строки с номером  $n$  удваивается количество букв, бывшее в предыдущей строке (с номером  $n - 1$ ), и к ним прибавляется количество букв, бывшее в строке перед предыдущей (с номером  $n - 2$ ).

На основании этого составим таблицу, отражающую количества букв “A”, “B” и “C” в получаемых строках с различными номерами  $n$ :

$n$	“A”	“B”	“C”
1	1	0	0
2	0	1	1
3	1	2	2
4	2	5	5
5	5	12	12
6	12	29	29
7	29	70	70
8	70	169	169

Проще всего вычислять количества символов в очередной строке так:

- сначала переписываем в очередную строку таблицы удвоенные значения из предыдущей строки таблицы;
- затем к только что записанным значениям прибавляем значения из соответствующих ячеек строки таблицы, расположенной перед предыдущей.

Итого получаем, что в восьмой строке имеется: 70 символов “A”, 169 символов “B” и 169 символов “C”.

Тогда искомое количество символов, отличных от “B”, в восьмой строке равно  $70 + 169 = 239$ .

*Ответ:* 239.

### Методический комментарий

В данном случае, чтобы показать более полное решение задачи, мы вычисляли количества в строках каждого из символов. Однако можно было бы вместо этого искать только количество символов “B” и общее число символов в строке с номером  $n$  (очевидно, оно тоже равно удвоенному числу символов в предыдущей строке плюс число символов в строке перед предыдущей), после чего вычислить их разность.

О решении других задач ЕГЭ с цепочками можно прочитать в статье: Богомолова О.Б., Усенков Д.Ю. Подготовка к ЕГЭ: расплетаем цепочки, или “Туда и обратно” // Информатика, 2011, № 4. С. 3–7.

### Задача В7

Сколько единиц содержится в двоичной записи результата выражения:

$$(2 \cdot 100_8)^{500} - 4^{501} + 2^{502}?$$

*Решение*

Эта задача кажется очень сложной, а вычисление таких огромных степеней — просто невозможным. Но давайте посмотрим, так ли это на самом деле.

Прежде всего отметим: все основания есть степени двойки. Это далеко не случайно и, как мы увидим чуть позже, существенно облегчает решение задачи. Поэтому выразим все числа в заданном выражении через степени числа 2 (попутно переведем восьмеричное число в десятичное):

$$\begin{aligned} (2 \cdot 100_8)^{500} - 4^{501} + 2^{502} &= (2 \cdot (2^3)^2)^{500} - (2^2)^{501} + 2^{502} = \\ &= (2 \cdot 2^6)^{500} - (2^2)^{501} + 2^{502} = \\ &= (2^7)^{500} - (2^2)^{501} + 2^{502} = 2^{3500} - 2^{1002} + 2^{502}. \end{aligned}$$

(При этом нужно не забывать: произведение двух степеней одного основания есть сумма этих степеней для того же основания, а возведение степени в степень — произведение этих степеней.)

Теперь нужно разобраться, что означает операция возведения в степень двойки, выполняемая в двоичной системе счисления.

Начнем преобразовывать степени двойки в двоичную систему счисления:

$$\begin{aligned} 2^0 &= 1 \\ 2^1 &= 10 \\ 2^2 &= 100 \\ 2^3 &= 1000 \\ &\dots \end{aligned}$$

Очевидно, что увеличение показателя степени двойки на 1 соответствует умножению на 2, что в двоичной системе счисления означает дописывание одного нуля справа. Поэтому нетрудно сделать вывод:  $2^n$  есть двоичное число, в котором имеется одна единица (в позиции  $n$ ; напомним, что в двоичных числах разряды (позиции) цифр нумеруются с нуля справа налево) и  $n$  нулей после нее. Запишем это так:

$$2^n = \underbrace{1000\dots000}_{n \text{ нулей}}$$

Тогда:

$$2^{3500} = \underbrace{1000\dots000}_{3500 \text{ нулей}}$$

$$2^{1002} = \underbrace{1000\dots000}_{1002 \text{ нуля}}$$

$$2^{502} = \underbrace{1000\dots000}_{502 \text{ нуля}}$$

Дальше нас ждут рассуждения несколько более сложные.

Рассмотрим операцию вычитания в двоичной системе счисления для чисел, в которых имеется только одна “лидирующая” (первая слева) единица и некоторое количество нулей. При этом заметим, что в нашем случае уменьшаемое заметно больше вычитаемого, поэтому отрицательный результат получаться не будет.

$$\begin{array}{r} 100\dots0000\dots00000 \\ - \quad \quad \quad 100\dots00000 \\ \hline 11\dots1100\dots00000 \end{array}$$

Нетрудно видеть, что при таком вычитании из числа с единицей в позиции  $n$  числа с единицей в позиции  $m$  получается двоичное число, в котором единицы стоят в позициях с  $(n - 1)$  по  $m$ , после которых записаны только нули.

В нашем случае  $2^{3500} - 2^{1002}$  дает число, в котором единицы стоят в позициях с 3499 до 1002.

Теперь рассмотрим операцию сложения в двоичной системе счисления, когда одно число содержит некоторое количество единиц в позициях с  $(n - 1)$  по  $m$ , а другое содержит только одну “лидирующую” единицу в позиции, меньшей  $m$ , и некоторое количество нулей.

$$\begin{array}{r} 1111\dots11100\dots000\dots00000 \\ + \quad \quad \quad \quad \quad \quad 10\dots00000 \\ \hline 1111\dots11100\dots010\dots00000 \end{array}$$

Очевидно, что такое сложение добавляет в получаемое число еще одну единицу в позиции, соответствующей степени двойки в прибавляемом числе.

Тогда в двоичном числе, являющемся результатом вычисления выражения  $2^{3500} - 2^{1002} + 2^{502}$ , единицы расположены в позициях с 3499 по 1002 включительно плюс имеется еще одна единица в позиции 502.

Всего единиц в этом числе  
 $(3499 - 1002 + 1) + 1 = 2499$ .  
 Ответ: 2499.

**Методический комментарий**

Именно то, что все числа в исходном выражении были кратны двойке, позволяет легко возводить их в любые сколь угодно большие степени в двоичной системе счисления и вычислять разность/сумму таких двоичных чисел чисто “умозрительно”, не выполняя практически никаких сложных вычислений.

Возможные условия подобных задач, конечно, могут меняться. Но можно предполагать, что разработчики заданий ЕГЭ будут придерживаться “правил игры”: исходные числа, которые предполагается возводить в степень, кратны 2, а показатели степени двоек в “приведенном” выражении даются “по убыванию” (у вычитаемого меньше, чем у уменьшаемого, а у прибавляемого числа — еще меньше), чтобы обеспечить решаемость задачи.

Единственная сложность, на которую желательно обратить внимание учащихся, — это на правильность определения количества единиц, расположенных с позиции  $x$  до позиции  $y$ . В качестве аналогии можно привести пример определения количества дней между двумя датами одного и того же месяца: например, в интервале от 5 до 15 числа заключено  $(15 - 5 + 1) = 11$  дней. Для наглядности можно показать это графически:

5	6	7	8	9	10	11	12	13	14	15
<span style="display: inline-block; border-top: 1px solid black; width: 100%;"></span> 11 дней										

Поэтому к разности номеров позиций нужно не забывать прибавлять единицу.

**Задача В12**

В первых двух столбцах электронной таблицы (прилагается файл с очень большим количеством строк) приведены названия городов и названия отделений банков, меняющих рубли на доллары и наоборот. В третьем столбце приведены курсы обмена долларов на рубли (за сколько рублей банк покупает один доллар). В четвертом столбце приведены курсы обмена рублей на доллары (за сколько рублей банк продает один доллар). Пустая ячейка означает, что данная обменная операция данным отделением банка не производится.

Определите максимальное количество долларов, которое можно получить при однократном обмене 50 000 рублей. Ответ округлите до копеек (т.е. до двух знаков после запятой).

Фрагмент файла представлен ниже в виде таблицы.

	А	В	С	Д
1	Город	Банк	Курс обмена долларов на рубли	Курс обмена рублей на доллары
2	Москва	Сбербанк	31,15	31,60
3	Москва	ВТБ	31,20	32,00
4	Санкт-Петербург	Сбербанк	31,10	31,50
5	Санкт-Петербург	Альфа-банк	31,25	31,73
6	Москва	Банк Москвы		31,07
7	Москва	Альфа-банк	31,05	31,15
8	Новосибирск	Банк Сибири	31,00	
9	Пермь	Сбербанк	31,17	31,19
10	Пермь	Сельхозбанк	31,90	33,00
...	...	...	...	...
12534	Тула	Альфа-банк	32,00	32,50

### Решение

Итак, разрешите вам представить первое в ЕГЭ практическое задание ☺!

Имеется файл электронной таблицы и имеется компьютер с редактором электронных таблиц (например, Excel, хотя файл предоставляется также в универсальном формате, пригодном для работы в любом редакторе электронных таблиц, например, OpenOffice.org Calc). Все, что требуется от учащегося, — правильно выполнить обработку данных этого файла, т.е. грамотно ввести расчетную формулу (включая требуемую функцию или функции), которая *автоматически* вычисляет требуемое значение. Ничего вычислять вручную, как требовалось ранее в заданиях ЕГЭ с электронными таблицами, не требуется.

В данном случае нам нужно в столбце “Курс обмена рублей на доллары” (так как мы покупаем у банка доллары) найти минимальное значение (минимальную цену одного доллара в рублях) и разделить на него имеющееся количество рублей. Для этого достаточно в любую свободную ячейку таблицы ввести формулу (для Excel):

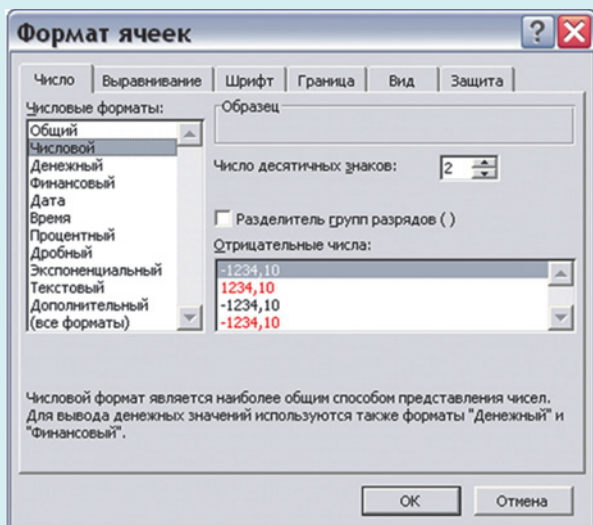
$$= 50000/\text{МИН}(D2:D12534)$$

Что же касается округления до двух значащих цифр после запятой, то здесь возможно два варианта решения задачи:

1) воспользоваться функцией округления, например, в Excel это функция ОКРУГЛ:

$$= \text{ОКРУГЛ}(50000/\text{МИН}(D2:D12534); 2)$$

2) просто установить для ячейки с формулой формат с двумя десятичными знаками (числовой или денежный):



Полученное в ячейке с данной формулой значение и нужно ввести в бланк ЕГЭ в качестве ответа на данное задание.

### Методический комментарий

В целом это задание достаточно простое (проверяется умение работать с формулами и знание основных функций). Единственное, на что нужно обратить внимание учащихся: поскольку в задаче спрашивается, какое *максимально возможное* число долларов можно купить на фиксированную сумму рублей, нужно делить это количество рублей на *минимально возможную* цену одного доллара, т.е.

искать именно *минимум* среди значений курсов обмена рублей на доллары, а не максимум.

### Задача В13

Для того же исходного файла электронной таблицы, что и в предыдущем задании, определить количество отделений банков, у которых разность курса продажи и покупки доллара превышает 1 рубль.

#### Решение

1. В отдельном столбце (например, E) формируем значения разности курсов продажи и покупки. Поскольку в общем случае (хотя реально так не бывает) курс продажи долларов может быть меньше курса их покупки, необходимо для каждой строки таблицы (каждого отделения какого-либо банка) вычислять абсолютное значение (модуль) разности курсов обмена рублей на доллары и долларов на рубли.

Для этого нужно ввести в ячейку E2 формулу:

$$=ABS(C2-D2)$$

а затем “размножить” ее на все строки исходной таблицы путем автозаполнения ячеек (для чего в формуле выбраны *относительные* ссылки).

2. Теперь необходимо для сформированного столбца абсолютных значений разностей курса получить количество ячеек, значения в которых удовлетворяют условию: “больше 1”. Для этого используется функция СЧЕТЕСЛИ (первый ее аргумент определяет диапазон, в пределах которого производится подсчет, а второй аргумент — условие, при выполнении которого та или иная ячейка подсчитывается):

$$=СЧЁТЕСЛИ(E2:E12534;“>1”)$$

3. Поскольку в столбцах, содержащих значения курсов обмена рублей на доллары и долларов на рубли, по условию задания, могут иметься пустые ячейки (и в этом случае вычисление разности курсов продажи и покупки доллара является некорректным), мы должны исключить из подсчета такие строки таблицы. Как это сделать?

В Excel имеется специальная функция — СЧИТАТЬПУСТОТЫ, которая определяет количество именно пустых ячеек в заданном диапазоне. Если же в используемом редакторе электронных таблиц подобной функции нет, то можно опять-таки воспользоваться функцией СЧЕТЕСЛИ, задав в качестве условия подсчета пустую строку: СЧЕТЕСЛИ(<диапазон>;“”).

Однако нужно учесть, что какие-то отделения банков могут вообще не выполнять операции обмена рублей и долларов. Поэтому если мы будем отдельно подсчитывать количества пустых ячеек в столбцах C и D и сложим их (либо если подсчитаем общее число пустых ячеек в объединенном диапазоне), то какие-то строки будут ошибочно учтены дважды. Нам же нужно подсчитать количество строк таблицы, в которых *хотя бы одна* из двух ячеек столбцов C и D пуста. Для этого можно сформировать в еще одном столбце (например, F) логические значения при помощи функции проверки свойств и значений ЕПУСТО (возвращает значение ИСТИНА, если указанная в ней ячейка пуста) и логической функции ИЛИ.

С этой целью достаточно ввести в ячейку F2 формулу:

**=ИЛИ(ЕПУСТО(С2);ЕПУСТО(D2))**

и “размножить” ее на все строки исходной таблицы с помощью автозаполнения, а затем подсчитать в этом диапазоне количество ячеек со значением ИСТИНА:

**=СЧЁТЕСЛИ(F2:F12534;ИСТИНА)**

4. Остается лишь вычесть из вычисленного ранее количества ячеек столбца E (разности курсов продажи и покупки долларов), значения которых по модулю больше 1, вычисленное только что количество строк таблицы, в которых хотя бы одна ячейка в столбце C и D пуста. Следовательно, итоговая формула для получения ответа на данное задание будет иметь вид:

**=СЧЁТЕСЛИ(E2:E12534;>1)-  
СЧЁТЕСЛИ(F2:F12534;ИСТИНА)**

Полученное значение и записывается в качестве ответа на задание.

#### Методический комментарий

Это задание уже существенно сложнее. Оно требует не только знания функций проверки свойств и значений, логических функций и функций подсчета, но и догадки о том, что необходимо исключать из подсчета строки, в которых вместо значений обменных курсов имеются пустые ячейки. На это, как и вообще на необходимость в подобных задачах своевременно отслеживать и исключать или компенсировать некорректные исходные данные, нужно обратить внимание учащих.

#### Задача В14

Используя тот же самый исходный файл, определите максимальный курс обмена долларов на рубли в Москве. Ответ округлите до двух знаков после запятой.

##### Решение

Здесь необходимо вспомнить, что в исходной таблице для одного и того же города могут быть указаны обменные пункты разных банков с разным курсом обмена. Нам же нужно найти максимальное значение в столбце “Курс обмена долларов на рубли” среди строк, в которых в столбце “Город” содержится значение “Москва”.

Проще всего сделать это следующим способом. Воспользуемся логической функцией ЕСЛИ, чтобы в отдельном столбце (например, F) получать в каждой из строк значение из столбца C, если ячейка столбца A содержит слово “Москва”, или пустое значение. Как и в предыдущем задании, для этого нужно ввести в ячейку F2 формулу:

**=ЕСЛИ(A2=“Москва”;C2;“”)**

а затем “размножить” ее на все строки исходной таблицы.

Теперь, когда мы фактически сделали в столбце F выборку значений курсов обмена долларов на рубли для отделений банков в Москве, нам остается лишь определить в столбце F максимум, используя функцию МАКС:

**=МАКС(F2:F12534)**

Останется лишь округлить полученное значение до двух цифр после запятой (при помощи “обрам-

ляющей” функции ОКРУТЛ или соответствующего формата ячейки) и записать его в качестве ответа.

#### Методический комментарий

Это задание проще предыдущего, но для решения требуется знать, как использовать функцию ЕСЛИ для формирования выборки значений одного столбца, для которых значения этого же или другого столбца соответствуют заданному условию.

#### Задача В15

Территория государства Флатландия представляет собой прямоугольник, на котором расположены точки-города. Эти города строго выровнены в горизонтальные и вертикальные линии (т.е. каждый город находится на пересечении соответствующей горизонтали и вертикали), местоположение которых определяется координатами X (по горизонтали) и Y (по вертикали) относительно верхнего левого угла прямоугольника (точно так же, как определяются координаты пикселей на экране ПК).

В верхнем левом углу (с координатами (0,0)) располагается морской порт Флатландии. Доставка товаров из него в города производится авиацией, причем грузы в каждый город доставляются за один прямой перелет из порта в соответствующий город и обратно (без захода на посадку в других городах).

В файле электронной таблицы в некотором диапазоне первой строки записаны расстояния в километрах от верхнего левого угла Флатландии до вертикальных линий (координаты X соответствующих городов), а в некотором диапазоне первого столбца — расстояния от верхнего левого угла до горизонтальных линий (координаты Y соответствующих городов).

Стоимость доставки грузов в тот или иной город определяется расстоянием перелета (каждый километр пути стоит 150 монет).

Требуется вычислить суммарную стоимость доставки грузов во все города Флатландии (округленно до целого значения).

##### Решение

Задание кажется очень сложным, но это впечатление, как часто и бывает, обманчиво. На самом деле вне зависимости от “словесного” условия мы имеем в файле электронной таблицы матрицу, первая строка которой содержит значения одной переменной, а первый столбец — значения второй переменной. Нам же требуется, по сути, вычислять значения некоторой функции двух переменных (аналогичным способом, например, выполняются вычисления при построении графика функции двух переменных).

Сделать это очень просто. Нам нужно вычислять расстояние перелета как корень квадратный из суммы квадратов переменных X и Y, и умножать эти расстояния на указанную в условии стоимость — 150.

Для этого прежде всего нужно ввести в ячейку B2 формулу:

**=КОРЕНЬ(B1\*B1+A2\*A2)\*150**

Файл:

	A	B	C	D	...	DT	DU	DV
1		15,8	29,32	53,00	...	557,02	748,30	989,99
2	13,50							
3	28,07							
4	43,58							
5	51,11							
...	...							
123	637,23							
124	700,00							
125	851,22							

Однако далее нужно правильно сформировать в этой формуле *смешанные ссылки*, чтобы обеспечить корректное их преобразование при “размножении” этой формулы на весь прямоугольный диапазон.

1) Когда формула копируется вправо, нам требуется, чтобы в ссылках на ячейку первого столбца не изменялось имя этого столбца, следовательно, знак абсолютной ссылки (\$) надо поместить перед буквой А.

2) Когда формула копируется вниз, требуется, чтобы в ссылках на ячейку первой строки не изменялся номер строки, следовательно, знак абсолютной ссылки надо также поместить перед номером 1.

В результате окончательная формула примет вид:  
=КОРЕНЬ(B\$1\*B\$1+\$A2\*\$A2)\*150

После этого нужно “размножить” эту формулу при помощи автозаполнения сначала в пределах текущей строки (т.е. в ячейки C2:DV2), а затем уже весь выделенный диапазон B2:DV2 “размножить” вниз на остальные строки с 3-й по 125-ю.

Итак, стоимости доставки грузов в каждый из городов (т.е. значения заданной функции двух переменных) подсчитаны. Остается лишь вычислить сумму всех этих значений. Для этого достаточно применить стандартную функцию СУММ для всего полученного двумерного диапазона:

=СУММ(B2:DV125)

Полученное значение остается округлить до целого при помощи стандартной функции ОКРУГЛ с указанием нулевого значения второго аргумента (другая имеющаяся функция ЦЕЛОЕ нам не годится, так как она просто отбрасывает дробную часть, округляя значение до меньшего целого, а нам требуется математическое округление) либо задав для ячейки числовое значение с нулевым числом десятичных знаков, и записать полученный результат как ответ.

### Методический комментарий

Это задание проверяет владение навыками формирования смешанных ссылок. Для подготовки к решению таких учебных задач можно порекомендовать книгу: Богомолова О.Б. Проектные работы с использованием электронных таблиц MS Excel: практикум (М.: БИНОМ. Лаборатория знаний, 2011), где рассмотрено 20 учебных проектов, реализованных средствами Microsoft Excel, а также подборки задач на использование статистических функций и построение графиков функций одной и двух переменных с соответствующими методическими пояснениями.

### Задачи В16–В18

Дан текстовый файл (достаточно большого объема; на упомянутой тренировочной работе в качестве такого файла предлагалось учебное пособие “Автоматизированное моделирование программных систем”). Требуется, используя предусмотренные в текстовом редакторе стандартные средства поиска, найти в предложенном тексте ответ на заданный вопрос.

### Методический комментарий

Средства поиска, реализованные в большинстве текстовых редакторов, обеспечивают “жесткий поиск” вхождения в текст заданного слова или фразы<sup>1</sup>, в отличие, например, от “мягкого” поиска по ключевой фразе в web, когда заданная ключевая фраза анализируется поисковой машиной и поиск выполняется с учетом всех возможных словоформ и расположения слов в тексте (если, конечно, в составе ключевой фразы не применены какие-либо команды языка запросов, формализующие такой поиск). На это различие нужно обязательно обратить внимание учащихся, которые в большинстве своем привыкли использовать функцию поиска именно в web.

Если речь идет о поиске слова, пропущенного в заданной цитате, то выполнить такое задание достаточно просто: всю заданную часть цитаты от начала до пропущенного слова (либо вторую ее часть от пропущенного слова до конца цитаты, — в зависимости от того, какая из этих частей больше) достаточно ввести в соответствующем поле окна **Найти**, а затем выполнить поиск и прочитать в тексте всю эту цитату.

Если же (как это и требовалось в тренировочной работе) нужно найти пропущенное слово в фразе, не обязательно являющейся цитатой, или ответ на заданный вопрос, то задача несколько усложняется, и речь идет уже не столько о применении технологического навыка использования средств поиска, сколько об умении осмысливать текст как вопроса, так и найденного фрагмента документа (что, надо сказать, в задании ЕГЭ не совсем корректно).

Прежде всего учащийся должен уметь выделять из заданной фразы или вопроса *ключевые слова* — слова, на которые приходится наибольшая часть смысловой нагрузки заданной фразы или вопроса. Это достаточно сложная и “неалгоритмизуемая” задача, здесь практически невозможно дать какие-либо готовые рекомендации “на все случаи жизни”. Очевидно лишь, что в качестве ключевых слов не следует использовать, например, предлоги и междометия и что скорее всего ключевыми словами

<sup>1</sup> В Microsoft Word, правда, предусмотрены такие настройки поиска, как **Все словоформы** и **Подстановочные знаки**, для использования которых нужно раскрыть нижнюю часть окна **Найти и заменить** щелчком мыши на кнопке **Больше**, а затем пометить соответствующие флажки. Однако работа с этими настройками в курсе информатики обычно не рассматривается, а руководств по применению подстановочных знаков для обеспечения поиска, например, фрагмента текста, в котором два заданных слова стояли бы рядом в одном предложении, найти не удалось. Поэтому вряд ли следует думать, что разработчики этих заданий ЕГЭ рассматривали на проверку знания именно этих функциональных возможностей поиска. Соответственно, мы также не будем вести речь об использовании этих возможностей. — Прим. авт.



могут являться содержащиеся в заданной фразе или вопросе термины, конкретные названия и пр.

После этого остается лишь, нажимая кнопку **Найти далее**, искать в текстовом документе вхождения заданного ключевого слова (только одного, здесь нельзя задавать сразу несколько ключевых слов, как при поиске в web!) и уже чисто по смыслу оценивать найденные фрагменты текста: действительно ли они содержат ответ на требуемый вопрос. Если при этом выбранное ключевое слово встречается в тексте слишком часто или если по нему найти информацию не удалось, то можно попытаться повторить поиск с другим ключевым словом. Наконец, в некоторых случаях может потребоваться при помощи поиска по ключевому слову (например, названию программного продукта или его конкретной возможности) найти соответствующий раздел заданного текстового документа, после чего вести поиск ответа на заданный вопрос уже “глазами”.

Скорее всего подобная неформализуемость таких заданий (в отличие, например, от поиска пропущенного слова в заведомо имеющейся в документе цитате) и неразвитость поисковых средств текстовых редакторов заставят разработчиков ЕГЭ отказаться от такого рода заданий либо изменить их форму и содержание, чтобы задания действительно проверяли именно умение пользоваться функционалом текстового редактора, а не понимание смысла предложенного текста (пусть даже фрагментарно).

## Задачи С1–С4

### Методический комментарий

В отличие от прежних версий ЕГЭ *все* задачи группы С в данной тренировочно-диагностической работе были посвящены написанию программ. Сложно сказать, насколько такое зауживание тематики оправданно с методической точки зрения, но можно предполагать, что в дальнейшем, с переходом на компьютерный формат сдачи ЕГЭ, контроль знаний и умений в области программирования будет осуществляться именно таким способом: при помощи заданий, в которых учащийся должен написать программу (на известном ему языке программирования), решающую ту или иную задачу, отладить эту программу в соответствующей среде программирования и отправить в качестве решения соответствующего задания ЕГЭ полученный листинг. Проверка же выполнения этих заданий с большой вероятностью будет осуществляться автоматическим путем “прогона” присланных учащимися листингов на заранее подготовленных разработчиками ЕГЭ наборах тестов, — так, как сейчас выполняется проверка решений на различных олимпиадах по программированию.

Поскольку предлагаемые задачи по программированию могут быть, вообще говоря, любыми, мы не будем здесь рассматривать аналоги заданий, предлагавшихся на упомянутой тренировочной работе (равно как и сами эти задания тренировочной работы, поскольку, как уже было сказано в начале этой

статьи, ФИПИ запрещает копирование контрольно-диагностических материалов этой работы). Укажем лишь характер предлагавшихся заданий.

**С1** — задача на ввод нескольких чисел (более двух), поиск среди них минимального или максимального и вывод результата. Очевидно, это — проверка умения использовать вложенные операторы ветвления<sup>2</sup>.

Однако, судя по всему, разработчики ЕГЭ ждут от учащегося не такого “тривиального” решения, а в случае автоматизированной проверки листингов программ используемая системой проверки версия языка программирования может попросту не содержать таких функций. Поэтому решение (в виде программного кода) лучше строить средствами, предусмотренными в общепринятом стандарте языка программирования, а не в его какой-то конкретной версии, но разработчики заданий ЕГЭ должны будут тогда явно оговаривать этот факт в условии заданий. — *Прим. авт.*

**С2** — работа с массивами (ввод значений элементов массива, их обработка и/или вывод на экран в заданной последовательности).

**С3** — ввод нескольких чисел, рассматриваемых в качестве координат двух или более точек на координатной плоскости, на которой вычерчено несколько графиков функций и закрашена какая-либо из образуемых ими криволинейных областей; программа должна вывести количество точек (с заданными координатами), попадающих в закрашенную область. (Очевидно, такая задача во многом родственна прежним заданиям **С1** ЕГЭ.)

**С4** — работа со строками (последовательностями символов — букв, цифр и других знаков) и словами (фрагментами этих строк, ограниченными некоторыми разделителями); предполагается умение грамотно вводить такие строки, осуществлять поиск в них местоположения заданных разделителей, выделять на основе этого “слова” (не забывая также, что “слова” могут быть ограничены разделителем только с одной стороны, если они располагаются в начале или в конце строки) и производить поочередный анализ и обработку этих слов. В этом вам также может помочь статья *Усенков Д.Ю., Богомолова О.Б.* “О знаках и строках замолвите слово..., или Несколько занимательных задач на работу со строками и множествами” // Информатика, 2011, № 16. С. 32–47.

В целом такая переориентация на практическое программирование означает (в случае, если эту “традицию” подхватят разработчики реального ЕГЭ), что в рамках изучения содержательной линии алгоритмики и программирования уже не удастся ограничиваться только лишь теоретическим рассмотрением основ того или иного языка. Потребуется изрядная доля практических занятий по решению задач по программированию (к чему, например, издательства учебно-методической литературы сегодня не очень готовы), а возможно — и подготовка, аналогичная подготовке к участию в олимпиадах по программированию не ниже чем городского уровня. Что, впрочем, школьникам наверняка пойдет только на пользу...

<sup>2</sup> В языке программирования Паскаль версии Pascal-ABC есть стандартные функции, возвращающие, соответственно, максимальный и минимальный из двух заданных аргументов. С их использованием подобная задача может быть решена буквально одной программной строкой, например: `maximum := Max(Max(a,b),Max(c,d));`

Однако, судя по всему, разработчики ЕГЭ ждут от учащегося не такого “тривиального” решения, а в случае автоматизированной проверки листингов программ используемая системой проверки версия языка программирования может попросту не содержать таких функций. Поэтому решение (в виде программного кода) лучше строить средствами, предусмотренными в общепринятом стандарте языка программирования, а не в его какой-то конкретной версии, но разработчики заданий ЕГЭ должны будут тогда явно оговаривать этот факт в условии заданий. — *Прим. авт.*