

# ИНФОРМАТИКА

4

**Плавали – знаем,**  
или “Неожиданности”  
в ЕГЭ-2014

40

**Е.А. Еремин –**  
**школьной**  
**информатике:**

сохраниться в школе  
в виде отдельного  
предмета  
и определиться  
наконец  
с содержанием

48

**Двоится в глазах?**  
Значит, вы всё  
склеили правильно!





НА ОБЛОЖКЕ

► “Тетрис” — одна из самых известных компьютерных игр в мире, была придумана в июне 1984 года советским программистом Алексеем Пажитновым, который работал в Вычислительном центре Академии наук. Первая версия “Тетриса” была реализована на компьютере “Электроника-60”. История, связанная с правами на игру, сложна и драматична. На некотором этапе в ней лично принимал участие Президент СССР Михаил Горбачев. “Тетрис” реализован практически на всех компьютерах и платформах. Имеется реализация “Тетриса” даже для осциллографа (!).

В НОМЕРЕ

- 3** ПЕДАГОГИЧЕСКИЙ МАРАФОН-2014
  - День учителя информатики
- 4** ЕГЭ
  - ЕГЭ: возможные особенности заданий 2014 года
  - Задачи с исполнителем: теперь на троих
- 23** СЕМИНАР
  - Оригинальные методы перевода чисел
- 24** ТЕХНОЛОГИИ
  - Как создать свою электронную книгу
- 40** ИНФОРМАТИКА В ЛИЦАХ
  - Е.А. Еремин
- 44** ВНЕКЛАСНАЯ РАБОТА
  - Вопросы по информатике для проведения школьных конкурсов “Что? Где? Когда?” и “Брейн-ринг”
- 48** ЗАНИМАТЕЛЬНЫЕ МАТЕРИАЛЫ ДЛЯ ПЫТЛИВЫХ УЧЕНИКОВ И ИХ ТАЛАНТЛИВЫХ УЧИТЕЛЕЙ
  - “В мир информатики” № 198

В ЛИЧНОМ КАБИНЕТЕ

Облачные технологии от Издательского дома “Первое сентября”

Уважаемые подписчики бумажной версии журнала!

Дополнительные материалы к номеру и электронная версия журнала находятся в вашем Личном кабинете на сайте [www.1september.ru](http://www.1september.ru).

Для доступа к материалам воспользуйтесь, пожалуйста, кодом доступа, вложенным в январский номер.

Срок действия кода: с 1 января по 30 июня 2014 года.

Для активации кода:

- зайдите на сайт [www.1september.ru](http://www.1september.ru);
- откройте Личный кабинет (создайте, если у вас его еще нет);
- введите код доступа и выберите свое издание.

Справки: [podpiska@1september.ru](mailto:podpiska@1september.ru) или через службу поддержки на портале “Первого сентября”.



ЭЛЕКТРОННЫЕ МАТЕРИАЛЫ  
Презентации к статьям номера

# ИНФОРМАТИКА

ПОДПИСНЫЕ ИНДЕКСЫ по каталогу “Почта России”: 79066 — бумажная версия, 12684 — электронная версия

<http://inf.1september.ru>

Учебно-методический журнал для учителей информатики  
Основан в 1995 г.  
Выходит один раз в месяц

РЕДАКЦИЯ:  
гл. редактор С.Л. Островский  
редакторы

Е.В. Андреева,  
Д.М. Златопольский  
(редактор вкладки  
“В мир информатики”)

Дизайн макета И.Е. Лукьянов  
верстка Н.И. Пронская  
корректор Е.Л. Володина  
секретарь Н.П. Медведева  
Фото: фотобанк Shutterstock  
Журнал распространяется по подписке  
Цена свободная  
Тираж 27 507 экз.  
Тел. редакции: (499) 249-48-96  
E-mail: [inf@1september.ru](mailto:inf@1september.ru)  
<http://inf.1september.ru>

ИЗДАТЕЛЬСКИЙ ДОМ  
“ПЕРВОЕ СЕНТЯБРЯ”

Главный редактор:  
Артем Соловейчик  
(генеральный директор)

Коммерческая деятельность:  
Константин Шмарковский  
(финансовый директор)

Развитие, IT  
и координация проектов:  
Сергей Островский  
(исполнительный директор)

Реклама, конференции  
и техническое обеспечение  
Издательского дома:  
Павел Кузнецов

Производство:  
Станислав Савельев

Административно-  
хозяйственное обеспечение:  
Андрей Ушков

Педагогический университет:  
Валерия Арсланьян (ректор)

ГАЗЕТА ИЗДАТЕЛЬСКОГО ДОМА  
Первое сентября – Е.Бирюкова

ЖУРНАЛЫ ИЗДАТЕЛЬСКОГО ДОМА  
Английский язык – А.Громушкина  
Библиотека в школе – О.Громова

Биология – Н.Иванова

География – О.Коротова

Дошкольное

образование – Д.Тюттерин

Здоровье детей – Н.Сёмина

Информатика – С.Островский

Искусство – О.Волкова

История – А.Савельев

Классное руководство

и воспитание школьников –

М.Битанова

Литература – С.Волков

Математика – Л.Рослова

Начальная школа – М.Соловейчик

Немецкий язык – М.Бузоева

ОБЖ – А.Митрофанов

Русский язык – Л.Гончар

Спорт в школе – О.Леонтьева

Технология – А.Митрофанов

Управление школой – Е.Рачевский

Физика – Н.Козлова

Французский язык – Г.Чесновицкая

Химия – О.Блохина

Школа для родителей – Д.Тюттерин

Школьный психолог – М.Чибисова

УЧРЕДИТЕЛЬ:  
ООО “ЧИСТЫЕ ПРУДЫ”

Зарегистрировано  
ПИ № ФС77-44341  
от 22.03.2011

в Министерстве РФ  
по делам печати  
Подписано в печать:  
по графику 15.04.2014,  
фактически 15.04.2014  
Заказ №  
Отпечатано в ОАО “Первая  
Образцовая типография”  
Филиал “Чеховский Печатный Двор”  
ул. Полиграфистов, д. 1,  
Московская область,  
г. Чехов, 142300  
Сайт: [www.chpk.ru](http://www.chpk.ru)  
E-mail: [sales@chpk.ru](mailto:sales@chpk.ru)  
Факс: 8 (495) 988-63-76

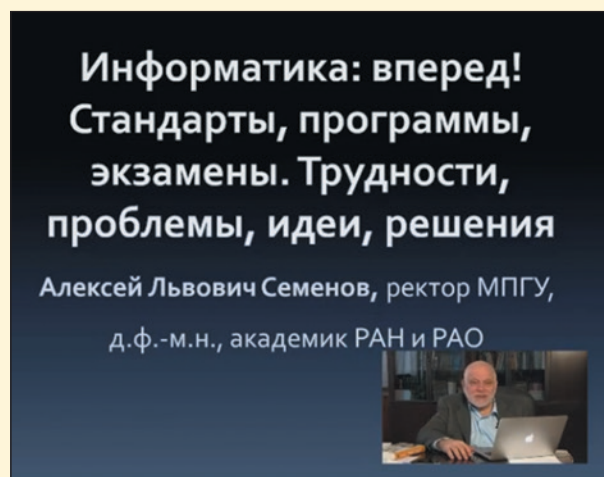
АДРЕС ИЗДАТЕЛЯ:  
ул. Киевская, д. 24,  
Москва, 121165  
Тел./факс: (499) 249-31-38

Отдел рекламы:  
(499) 249-98-70  
<http://1september.ru>

ИЗДАТЕЛЬСКАЯ ПОДПИСКА:  
Телефон: (499) 249-47-58  
E-mail: [podpiska@1september.ru](mailto:podpiska@1september.ru)

# День учителя информатики на XIII Всероссийском педагогическом марафоне в МПГУ

Репортаж о Дне учителя информатики, который прошел 4 апреля на гостеприимной площадке в Московском педагогическом государственном университете, можно прочитать на сайте [марафон.1сентября.рф](http://марафон.1сентября.рф). Там же имеется ссылка на запись видеолекции А.Л. Семенова.



▲ Видеолекция А.Л. Семенова собрала полный зал



▲ Без границ: вебинар М.Н. Семионенкова (Сан-Франциско)



▲ Надежда Николаевна Самылкина: экспресс-интервью после семинара



▲ Александр Георгиевич Гейн и Татьяна Антоновна Бурмистрова



▲ Живой выпуск "В мир информатики": Дмитрий Михайлович Златопольский



▲ На открытии. В кадр попали Людмила Леонидовна Босова и Сергей Дмитриевич Каракозов



ЕГЭ

## ЕГЭ: возможные особенности заданий 2014 года

**О.Б. Богомолова,**  
д. п. н., преподаватель  
Московского  
государственного  
гуманитарного  
университета (МГГУ)  
им. М.А. Шолохова,  
Москва

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

► Новый учебный год — новые проверочные работы в формате ЕГЭ. И — как это теперь довольно часто бывает — новые задания, иногда незначительно, а иногда довольно существенно отличающиеся от заданий Единого госэкзамена предыдущих лет. Правда, как показывает практика, проверочные задания от МИОО и СтатГрад обычно бывают сложнее, чем предлагаемые ФИПИ задания настоящего ЕГЭ, отраженные в демонстрационном варианте. Однако же, “чем черт не шутит”, — вполне возможно, что на ЕГЭ вдруг решат дать и усложненные задания, которых в демоварианте не было (и прецеденты уже, к сожалению, бывали). Поэтому желательно разобрать с учащимися и решение таких усложненных задач,

чтобы более полноценно подготовить их к экзамену.

Ниже приводится ряд заданий, которые были представлены в проверочной работе СтатГрад и оказались немного непривычными для школьников. Отметим, правда, что поскольку разработчики этих заданий запрещают их публикацию<sup>1</sup>, ниже приводятся не задания из проверочной работы, а однотипные с ними задания, придуманные авторами статьи.

### 1. “Старая знакомая” — двоичная арифметика

Даны числа, записанные в разных системах счисления, и в том числе в виде арифметических выражений. Укажите среди них то, в двоичной

<sup>1</sup> Чего авторы данной статьи совершенно не одобряют. Да, — не должно быть публикации этих заданий перед собственно проверочной работой. Но после того как такая работа уже проведена, запрет на публикацию ее заданий вызывает только недоумение: как же иначе (не размещая задания в Интернете или не публикуя их условия в печатных изданиях) реализовать дистанционное обучение или давать рекомендации по решению задач ученикам и учителям?

записи которого имеется ровно четыре единицы. Если таких чисел несколько, то укажите наибольшее из них.

- 1)  $В8_{16}$       2)  $110111_2$       3)  $216_8 + 35_{16}$       4)  $29_{10} * 10_8$

*Решение*

В принципе эта задача изменилась ненамного. Просто если раньше требовалось только переводить числа в двоичную систему счисления, то теперь понадобится еще и выполнять над ними соответствующие арифметические операции. А кроме того, в предыдущие годы правильный вариант ответа был обычно только один, и, найдя его, можно было не проверять остальные варианты, — а теперь придется по-честному выполнять проверку всех вариантов ответа, да еще и сравнивать получившиеся числа с четырьмя единицами по величине.

Решать такую задачу удобнее всего так:

- просто числа сразу переводить в двоичную систему;
- числа, задействованные в арифметических вычислениях, сначала переводить в десятичные, выполнять вычисления с ними, а потом переводить результат в двоичную систему;
- подсчитать во всех полученных числах количество единиц;
- числа, в которых есть по четыре единицы, сравнить между собой и найти наибольшее по величине; можно выполнять такое сравнение в двоичной системе счисления либо (большинству учащихся так проще) перевести эти числа в десятичные и сравнивать уже их.

В нашем случае (запишем все варианты ответа в виде таблицы):

№ варианта	Вариант ответа	Десятичные вычисления	Двоичное представление результата	Кол-во единиц	Десятичное представление результата
1	$В8_{16}$	–	10111000	4	184
2	$110111_2$	–	110111	5	55
3	$216_8 + 35_{16}$	$142 + 53 = 195$	11000011	4	195
4	$29_{10} * 10_8$	$29 * 8 = 232$	11101000	4	232

Итак, у нас имеются три числа, содержащих ровно по четыре единицы (выделены в таблице зеленой закраской).

Сравнить их можно как в десятичном представлении (очевидно, что наибольшее — число 232), так и в двоичном:

10111000      11000011      11101000

— больше то число, в котором больше единиц “кучкуются” возле левого края числа ☺.

*Ответ:*  $29_{10} * 10_8$  (ответ № 4).

**Комментарий.** В подобной задаче может потребоваться и подсчет значащих нулей. В этом случае нужно помнить, что значащими являются нули, перед которыми в числе есть хотя бы одна единица.

## 2. Родственники и прочие реляции: сериал, новый сезон<sup>2</sup>

Похоже, задачи на работу с базами данных, сюжет которых связан с поиском различных родственников, так понравился разработчикам заданий ЕГЭ, что подобные задачи стали еще более разнообразными. И то правда — разных родственников же много, а значит, можно придумать и много вариаций задачи. Сначала от мам и пап, сыновей и дочерей уже перешли к поискам племянников и племянниц, а сегодня “на повестке дня” — дяди и тети. Итак...

В фрагменте базы данных представлены сведения о родственных связях. На основании этих данных определите фамилию и инициалы дяди Кусто М.И.

ID	Фамилия_И.О.	Пол
101	Кусто И.Ф.	М
102	Кусто Л.И.	Ж
103	Кусто М.И.	М
104	Лунина А.В.	Ж
105	Никитина К.Т.	Ж
106	Симонов З.Т.	М
107	Туринец И.Н.	М
108	Туринец К.И.	М
109	Туринец А.И.	Ж
110	Туринец И.И.	Ж
111	Янов Ю.К.	М

ID родителя	ID ребенка
101	102
101	103
104	111
105	108
105	109
105	110
107	108
107	109
107	110
109	102
109	103

<sup>2</sup> См.: Богомолова О.Б., Усенков Д.Ю. Родственники и прочие реляции (новые задачи с базами данных) // Информатика, 2012, № 3. С. 38–47.

- 1) Кусто Л.И.                      3) Туринец К.И.  
 2) Туринец И.И.                4) Туринец А.И.

*Решение*

Прежде всего обратим внимание: таблицы базы данных содержат информацию только об одном виде родственных связей — “родитель — ребенок”. А нам нужно найти дядю заданного персонажа. Значит, нужно эту родственную связь выразить через связи “родители — дети”.

Дядя — это родной брат отца или матери. Но братьев и сестер данная БД тоже “напрямую” искать не позволяет. Поэтому выстраиваем следующую цепь рассуждений:

- для заданного человека ищутся отец и мать;
- для них, в свою очередь, также ищутся родители (т.е. бабушка и дедушка заданного персонажа);
- для найденных бабушки и дедушки ищутся все другие их дети, кроме родителей заданного персонажа. Это и есть его дяди и тети.

Теперь решение можно коротко записать так:

- 1) Кусто М.И. — ID = 103 (из левой таблицы);
- 2) родители ID 103 → ID = 101 и 109 (по правой таблице);
- 3) родители ID 109 → ID = 105 и 107 (по правой таблице; для ID 101 в ней сведений о родителях нет);
- 4) дети ID 105 и 107 → ID = 108, 109, 110 (также по правой таблице);
- 5) ID 109 — это уже ранее рассмотренный один из родителей Кусто М.И., поэтому данный номер из рассмотрения мы исключаем;
- 6) ID 110 — это Туринец И.И., женского пола (см. левую таблицу), значит, это тетя Кусто М.И., которая в данном случае нас не интересует;
- 7) ID 108 — это Туринец К.И., мужского пола (см. левую таблицу). Следовательно, это и есть искомый дядя Кусто М.И.

Задача решена.

*Ответ:* Туринец К.И. (ответ № 3).

**Комментарий.** “Дерево родственных связей” для данной задачи можно представить в виде схемы, представленной ниже.

А учащимся можно порекомендовать подучить на всякий случай, какие еще бывают названия родственников (например, шурин, деверь, сноха, невестка и пр.), чтобы не “сесть в лужу”, когда создатели заданий ЕГЭ решат еще немного усложнить задание... ☺

### 3. От БД — к ЭТ

Следующая задача — уже не про базы данных, а про не менее любимые разработчиками ЕГЭ электронные таблицы. Раньше в ЕГЭ встречались два типа заданий с ЭТ: на проверку знаний про формулы, абсолютные, относительные и смешанные ссылки и на знание принципов построения диаграмм. Собственно, в проверочной работе СтатГрад и остались эти два типа задач, но если задание с диаграммой почти не изменилось, то задание про ссылки поменялось самым коренным образом. Взгляните сами...

В ячейке **K12** электронной таблицы была записана некая формула. Затем ее скопировали в ячейку **H15**. Тогда в соответствии с формулой в ячейке **H15** ее значение стало равно произведению значений в ячейках **C10** и **F7**. Укажите, сколько из приведенных ниже высказываний не противоречат сказанному.

1. Значение в ячейке **K12** равно  $a*b$ , где  $a$  — значение ячейки **C10**, а  $b$  — значение ячейки **F7**.
2. Значение в ячейке **K12** равно  $a*b$ , где  $a$  — значение ячейки **C10**, а  $b$  — значение ячейки **I4**.
3. Значение в ячейке **K12** равно  $a*b$ , где  $a$  — значение ячейки **F10**, а  $b$  — значение ячейки **F4**.
4. Значение в ячейке **K12** равно  $a^2$ , где  $a$  — значение ячейки **F7**.

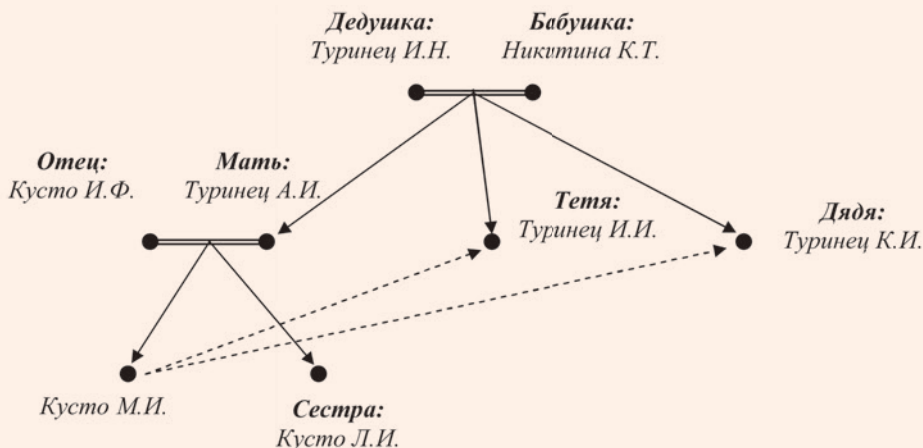
- 1) 1                      3) 3  
 2) 2                      4) 4

*Решение*

Главный секрет здесь в том, что довольно “обтекаемые” формулировки в приведенных выше четырех высказываниях не позволяют однозначно определить, о каких именно ссылках в них идет речь (так как говорится в них не о формулах, а только об используемых ячейках). И потому в каждом высказывании речь может идти как об абсолютных, так и об относительных или смешанных ссылках!

Разберем теперь каждое высказывание и посмотрим, можно ли подобрать такой тип ссылок в формуле, чтобы это высказывание оказалось верным.

**Высказывание 1.** Очевидно, что при копировании формулы из одной ячейки в другую используемые в формуле ячейки не изменяются только в одном случае — если это **абсолютные** ссылки. Так что если формула в ячейке **K12** имела вид  $=\$C\$10*\$F\$7$ , то в ячейке **H15** формула такой и



останется, — что нам и требовалось (произведение значений ячеек C10 и F7). Значит, первое высказывание условию задачи не противоречит.

**Высказывание 2.** Здесь при копировании формулы первый сомножитель не изменился (а мы только что показали, что это может быть, если ссылка на данную ячейку — абсолютная), а второй сомножитель при переносе формулы из ячейки K12 в ячейку H15 должен был измениться с I4 на F7. Все верно: если второй сомножитель был выражен в формуле обычной, относительной ссылкой, то при таком копировании формулы буква должна “уменьшиться” на три позиции латинского алфавита, а цифра — наоборот, увеличиться на 3. Что мы и наблюдаем. Значит, формула в ячейке K12 имела вид  $=\$C\$10*I4$ , а в ячейке H15 стала такой:  $=\$C\$10*F7$ . Опять мы получили искомое произведение значений ячеек, следовательно, второе высказывание условию задачи тоже не противоречит.

**Высказывание 3.** После копирования формулы здесь изменились оба сомножителя: первый из них в ячейке K12 соответствовал ячейке F10, а после копирования формулы в H15 предположительно изменился на C10. А второй сомножитель должен был соответственно измениться с F4 на F7. Может ли такое быть? Очевидно, да, — если в первом сомножителе было относительное имя столбца и абсолютный номер строки, а во втором — наоборот, относительный номер строки и абсолютное имя столбца. То есть, если формула в ячейке K12 была записана как  $=F\$10*F4$ , то в ячейке H15 она совершенно закономерно примет вид:  $=C\$10*F7$  (в относительных частях этих смешанных ссылок буква “уменьшается” на три позиции латинского алфавита, а цифра увеличивается на 3). В любом случае цель достигнута, а третье высказывание тоже не противоречиво.

**Высказывание 4.** Наконец, может ли быть такое, что произведение до копирования формулы было квадратом? Вспомним, что возведение в квадрат — это и есть умножение значения само на себя. Значит, в четвертом высказывании речь идет о следующем изменении ссылок на ячейки-сомножители при копировании формулы из K12 в H15: произведение F7 на F7 превратилось в произведение C10 на F7. Нетрудно догадаться, что в этом случае одна из “исходных” ссылок на F7 была абсолютной, а вторая — относительной. Или, говоря иначе, если формула в ячейке K12 имела вид, например,  $=F\$7*F7$ , то в ячейке H15 она станет такой:  $=F\$7*C10$  (во второй ссылке опять же буква “уменьшается” на три позиции латинского алфавита, а цифра увеличивается на 3). Поэтому четвертое высказывание тоже не содержит никаких противоречий с условием задачи.

Делаем вывод: не противоречащих высказываний в данном случае — четыре (т.е. все имеющиеся).

**Ответ:** 4 (ответ №4).

**Комментарий.** Задача оказалась не такой уж сложной (как казалось сначала). Только нужно правильно понять условие и не забывать перебирать все возможные варианты ссылок (абсолютных, смешанных и относительных).

## 4. Пишем звук

Достаточно “традиционная” уже задача все же не менее “традиционно” вызывает у многих школьников трудности, особенно после того как точные расчеты в ней заменили на своего рода экспертную оценку.

Четырехканальная (квадро) звукозапись производилась с частотой дискретизации 32 кГц и с 16-битным разрешением. Длительность записанного звука составила две минуты. Сжатие данных при оцифровке не применялось. Какая из приведенных ниже величин ближе всего к размеру полученного файла?

- 1) 5 Мб
- 2) 15 Мб
- 3) 25 Мб
- 4) 35 Мб

**Решение**

$$\begin{aligned} & \text{Сами вычисления производятся как обычно:} \\ & 16 \text{ (бит)} \times 32 \text{ (кГц)} \times 2 \text{ (мин.)} \times 4 \text{ (канала записи)} = \\ & = 16 \text{ (бит)} \times 32\,000 \text{ (Гц)} \times (2 \times 60) \text{ (с)} \times 4 \text{ (канала записи)} = \\ & = 2^4 \times (2^7 \times 5^3) \times (2^3 \times 5 \times 3) \times 2^2 = \\ & = 2^{16} \times 5^4 \times 3 = 1875 \times 2^{16} \text{ (бит)} = \frac{1875 \times 2^{16}}{2^{23}} \text{ (Мб)} = \\ & = \frac{1875}{2^7} = 14,6484 \approx 15 \text{ (Мб)}. \end{aligned}$$

Как видим, полученное значение (14,6484) округленно равно 15 Мб. Значит, это и есть правильный вариант ответа.

**Ответ:** 15 Мб (ответ № 2).

**Комментарий.** В последнее время в таких задачах разрешение сразу дается в битах. Надеемся, читатели не забыли, как вычислять значение разрешения в количестве бит, если разрешение будет задано количеством возможных градаций громкости? Напомним: если, например, сказано, что обеспечивается 128 градаций, то такое разрешение соответствует 7 битам ( $2^7 = 128$ ), а при количестве градаций, не кратном степени двойки, округление производится в большую сторону.

## 5. Кодирование

Задачи на кодирование (и, соответственно, на знание правил Фано) были и раньше. Но там предлагалось или выбрать для буквы такой код, который не дает неоднозначности кодирования “в комплекте” с уже известными кодами других букв, или найти, какой код декодируется при известном наборе кодов букв, а какие — нет. Теперь же требуется определить такое слово (из числа заданных), которое декодируется (в отличие от прочих) однозначно.

Передаются сообщения, состоящие только из пяти возможных букв: А, Б, В, Е, И. При этом буквы кодируются следующими числовыми последовательностями:

А — 000, Б — 001, В — 01, Е — 11, И — 1.

Среди приведенных ниже слов укажите то, которое можно после кодирования декодировать однозначно (т.е. только одним-единственным способом). А если таких слов несколько, то укажите первое из них по алфавиту:

- 1) ВАБА
- 2) БИИБ
- 3) ВЕИА
- 4) ни одно из сообщений не декодируется однозначно

*Решение*

Прежде всего напомним сами правила (или условия) Фано:

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

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

При этом обратное условие Фано обычно применяют, если не выполняется “прямое” условие Фано.

В нашем случае сразу видно, что оба условия Фано для данного набора кодов букв не выполняются, и “виновны” в этом коды букв Е и И. Именно их в закодированном сообщении легко спутать друг с другом. Поэтому скорее всего слова, содержащие эти “неправильные” буквы, и будут декодироваться неоднозначно, а слово, которое этих букв не содержит, будет являться правильным ответом. Но давайте проверим это:

$$1) \text{ ВАБА} \xrightarrow{\text{кодируем}} \begin{pmatrix} B & A & B & A \\ 01 & 000 & 001 & 000 \end{pmatrix} \rightarrow$$

$$\rightarrow 01000001000 \xrightarrow{\text{декодируем}} \begin{pmatrix} 01 & 000 & 001 & 000 \\ B & A & B & A \end{pmatrix}.$$

И иначе декодирование выполнить не удастся.

$$2) \text{ БИИБ} \xrightarrow{\text{кодируем}} \begin{pmatrix} B & И & И & B \\ 001 & 1 & 1 & 001 \end{pmatrix} \rightarrow$$

$$\rightarrow 00111001 \xrightarrow{\text{декодируем}} \begin{pmatrix} 001 & 11 & 001 \\ B & E & B \end{pmatrix}.$$

Нетрудно видеть, что можно получить совсем другое слово, если перегруппировать единицы в середине кода по-другому. Значит, в этом случае декодирование неоднозначно.

$$3) \text{ ВЕИА} \xrightarrow{\text{кодируем}} \begin{pmatrix} B & E & И & A \\ 01 & 11 & 1 & 000 \end{pmatrix} \rightarrow$$

$$\rightarrow 01111000 \xrightarrow{\text{декодируем}} \begin{pmatrix} 01 & 1 & 11 & 000 \\ B & И & E & A \end{pmatrix}.$$

Опять можно получить другое слово, перегруппировав единицы в середине кода. Значит, и это слово кодируется и декодируется неоднозначно.

Четвертый же вариант ответа можно уже не рассматривать, так как мы уже нашли хотя бы одно “правильное” слово и оно стоит в алфавите раньше.

*Ответ:* ВАБА (ответ №1).

**Комментарий.** Здесь задача немного упрощалась очевидностью возникающей путаницы, когда две “неправильные” буквы в слове стоят рядом. А вот если они будут “разбавлены” правильными буквами, то декодирование может оказаться и однозначным! Тут уже

нужно будет “честно” проверять каждое слово, а не делать сразу окончательный вывод только на основании наличия или отсутствия “неправильных” букв в слове.

## 6. Игра с отрезками

А вот это, пожалуй, одна из самых сложных задач во всей современной группе А! Пожалуй, можно было бы заявить протест разработчикам ЕГЭ: такая задача вполне могла бы стоить не одного балла, а хотя бы двух!

На числовой прямой даны два отрезка:  $X = [7, 11]$  и  $Y = [9, 13]$ . Выберите отрезок  $Z$  такой, что обе приведенные ниже формулы истинны при любом значении переменной  $t$ :

$$\begin{aligned} & \neg(t \in X) \rightarrow \neg(t \in Z) \\ & (t \in Z) \rightarrow (t \in Y) \end{aligned}$$

Если таких отрезков окажется несколько, то выберите тот, который имеет наибольшую длину.

- 1) [8,10]
- 2) [6,12]
- 3) [6,14]
- 4) [8,18]

*Решение (способ 1)*

На первый взгляд эта задача напоминает прежние типа “Какое имя соответствует истинности выражения:

(Первая буква гласная)  $\wedge$  (Вторая буква согласная)  $\rightarrow$  (Последняя буква гласная) и т.д.”, которые решались путем формального заполнения соответствующей таблицы истинности. Можно попытаться действовать аналогичным образом и здесь.

Сначала соединим обе формулы в одну через логическую операцию И, поскольку они по условию должны быть истинными одновременно:

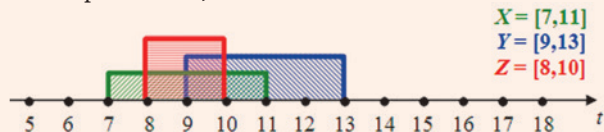
$$\begin{aligned} & \neg(t \in X) \rightarrow \neg(t \in Z) \\ & (t \in Z) \rightarrow (t \in Y) \end{aligned} \Rightarrow$$

$$\Rightarrow \neg(t \in X) \rightarrow \neg(t \in Z) \wedge ((t \in Z) \rightarrow (t \in Y))$$

Теперь начнем проверять каждый из отрезков  $Z$ , перечисленных в качестве вариантов ответа.

$$1) Z = [8, 10]$$

Вычертим числовую прямую с размеченными на ней отрезками  $X$ ,  $Y$  и  $Z$ :



Будем называть “неизменным интервалом” такой отрезок числовой прямой, на котором сохраняется одна и та же ситуация его принадлежности одному, двум или трем заданным отрезкам  $X$ ,  $Y$  и  $Z$  (либо не принадлежности ни одному из них). Тогда в данном случае можно выделить следующие “неизменные интервалы” значений  $t$ :  $[-\infty, 7]$ ,  $[7, 8]$ ,  $[8, 9]$ ,  $[9, 10]$ ,  $[10, 11]$ ,  $[11, 13]$ ,  $[13, \infty]$  (на граничные точки пока внимание не обращаем, поэтому они включены во все соответствующие отрезки).

Составим таблицу, строки которой соответствуют каждому из получившихся “неизменных интер-



валов”, и для каждого такого интервала определим результат выполнения заданного (и немного преобразованного нами) логического выражения (обозначим его как  $T$ ):

$$T = (\neg(t \notin X) \rightarrow \neg(t \notin Z)) \wedge ((t \notin Z) \rightarrow (t \notin Y))$$

При этом:

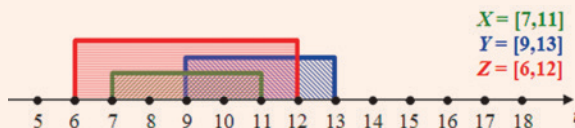
- не забываем, что операция следования дает нулевой результат в единственном случае  $1 \rightarrow 0$  и единичный результат — во всех остальных трех случаях ( $1 \rightarrow 1, 0 \rightarrow 0, 0 \rightarrow 1$ );
- исходные логические значения, подставляемые в это логическое выражение, равны 1 при истинности соответствующего факта либо 0 — при его ложности (например, факт  $t \notin X$  соответствует 1, если данный интервал действительно не “накрыт” отрезком  $X$ , либо 0, если данный интервал входит в отрезок  $X$ ).

Интервал	Факты			Значение выражения $T$
	$t \notin X$	$t \notin Y$	$t \notin Z$	
$[-\infty, 7]$	1	1	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (0 \rightarrow 0) \wedge (1 \rightarrow 1) = 1 \wedge 1 = 1$
$[7, 8]$	0	1	1	$(\neg 0 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (1 \rightarrow 0) \wedge (1 \rightarrow 1) = 0 \wedge 1 = 0$
$[8, 9]$	0	1	0	$(\neg 0 \rightarrow \neg 0) \wedge (0 \rightarrow 1) = (1 \rightarrow 1) \wedge (0 \rightarrow 1) = 1 \wedge 1 = 1$
$[9, 10]$	0	0	0	$(\neg 0 \rightarrow \neg 0) \wedge (0 \rightarrow 0) = (1 \rightarrow 1) \wedge (0 \rightarrow 0) = 1 \wedge 1 = 1$
$[10, 11]$	0	0	1	$(\neg 0 \rightarrow \neg 1) \wedge (1 \rightarrow 0) = (1 \rightarrow 0) \wedge (1 \rightarrow 0) = 0 \wedge 0 = 0$
$[11, 13]$	1	0	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 0) = (0 \rightarrow 0) \wedge (1 \rightarrow 0) = 1 \wedge 0 = 0$
$[13, \infty]$	1	1	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (0 \rightarrow 0) \wedge (1 \rightarrow 1) = 1 \wedge 1 = 1$

Как нетрудно видеть, для трех интервалов ( $[7, 8]$ ,  $[10, 11]$  и  $[11, 13]$ ) заданное логическое выражение обращается в нуль, следовательно, отрезок  $Z = [8, 10]$  не является правильным ответом.

2)  $Z = [6, 12]$

Снова вычертим числовую прямую с размеченными на ней отрезками  $X$ ,  $Y$  и  $Z$ :



В данном случае можно выделить следующие “неизменные интервалы” значений  $t$ :

$[-\infty, 6], [6, 7], [7, 9], [9, 11], [11, 12], [12, 13], [13, \infty]$

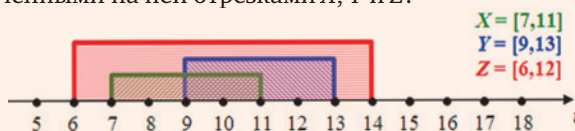
Вновь составим таблицу и для каждого интервала определим результат выполнения заданного логического выражения  $T$ .

Интервал	Факты			Значение выражения $T$
	$t \notin X$	$t \notin Y$	$t \notin Z$	
$[-\infty, 6]$	1	1	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (0 \rightarrow 0) \wedge (1 \rightarrow 1) = 1 \wedge 1 = 1$
$[6, 7]$	1	1	0	$(\neg 1 \rightarrow \neg 0) \wedge (0 \rightarrow 1) = (0 \rightarrow 1) \wedge (0 \rightarrow 1) = 1 \wedge 1 = 1$
$[7, 9]$	0	1	0	$(\neg 0 \rightarrow \neg 0) \wedge (0 \rightarrow 1) = (1 \rightarrow 1) \wedge (0 \rightarrow 1) = 1 \wedge 1 = 1$
$[9, 11]$	0	0	0	$(\neg 0 \rightarrow \neg 0) \wedge (0 \rightarrow 0) = (1 \rightarrow 1) \wedge (0 \rightarrow 0) = 1 \wedge 1 = 1$
$[11, 12]$	1	0	0	$(\neg 1 \rightarrow \neg 0) \wedge (0 \rightarrow 0) = (0 \rightarrow 1) \wedge (0 \rightarrow 0) = 1 \wedge 1 = 1$
$[12, 13]$	1	0	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 0) = (0 \rightarrow 0) \wedge (1 \rightarrow 0) = 1 \wedge 0 = 0$
$[13, \infty]$	1	1	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (0 \rightarrow 0) \wedge (1 \rightarrow 1) = 1 \wedge 1 = 1$

Все же есть один интервал ( $[12, 13]$ ), где заданное логическое выражение обращается в нуль. Значит, отрезок  $Z = [6, 12]$  тоже не может быть правильным ответом.

3)  $Z = [6, 14]$

Числовая прямая с размеченными на ней отрезками  $X$ ,  $Y$  и  $Z$ :



Здесь можно выделить следующие “неизменные интервалы” значений  $t$ :  
 $[-\infty, 6]$ ,  $[6, 7]$ ,  $[7, 9]$ ,  $[9, 11]$ ,  $[11, 13]$ ,  $[13, 14]$ ,  $[14, \infty]$

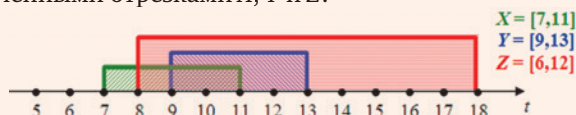
Таблица с результатами выполнения заданного логического выражения  $T$ .

Интервал	Факты			Значение выражения $T$
	$t \notin X$	$t \notin Y$	$t \notin Z$	
$[-\infty, 6]$	1	1	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (0 \rightarrow 0) \wedge (1 \rightarrow 1) = 1 \wedge 1 = 1$
$[6, 7]$	1	1	0	$(\neg 1 \rightarrow \neg 0) \wedge (0 \rightarrow 1) = (0 \rightarrow 1) \wedge (0 \rightarrow 1) = 1 \wedge 1 = 1$
$[7, 9]$	0	1	0	$(\neg 0 \rightarrow \neg 0) \wedge (0 \rightarrow 1) = (1 \rightarrow 1) \wedge (0 \rightarrow 1) = 1 \wedge 1 = 1$
$[9, 11]$	0	0	0	$(\neg 0 \rightarrow \neg 0) \wedge (0 \rightarrow 0) = (1 \rightarrow 1) \wedge (0 \rightarrow 0) = 1 \wedge 1 = 1$
$[11, 13]$	1	0	0	$(\neg 1 \rightarrow \neg 0) \wedge (0 \rightarrow 0) = (0 \rightarrow 1) \wedge (0 \rightarrow 0) = 1 \wedge 1 = 1$
$[13, 14]$	1	1	0	$(\neg 1 \rightarrow \neg 0) \wedge (0 \rightarrow 1) = (0 \rightarrow 1) \wedge (0 \rightarrow 1) = 1 \wedge 1 = 1$
$[14, \infty]$	1	1	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (0 \rightarrow 0) \wedge (1 \rightarrow 1) = 1 \wedge 1 = 1$

Обратим внимание: на всех без исключения интервалах заданное логическое выражение равно единице. Значит, отрезок  $Z = [6, 14]$  — это и есть правильный ответ. Однако проверим и оставшийся, четвертый вариант ответа.

4)  $Z = [8, 18]$

Числовая прямая с размеченными отрезками  $X$ ,  $Y$  и  $Z$ :



“Неизменные интервалы” значений  $t$ :

$[-\infty, 7]$ ,  $[7, 8]$ ,  $[8, 9]$ ,  $[9, 11]$ ,  $[11, 13]$ ,  $[13, 18]$ ,  $[18, \infty]$

Таблица результатов выполнения заданного логического выражения  $T$ .

$$T = (\neg(t \notin X) \rightarrow \neg(t \notin Z)) \wedge ((t \notin Z) \rightarrow (t \notin Y))$$

Интервал	Факты			Значение выражения $T$
	$t \notin X$	$t \notin Y$	$t \notin Z$	
$[-\infty, 7]$	1	1	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (0 \rightarrow 0) \wedge (1 \rightarrow 1) = 1 \wedge 1 = 1$
$[7, 8]$	0	1	1	$(\neg 0 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (1 \rightarrow 0) \wedge (1 \rightarrow 1) = 0 \wedge 1 = 0$
$[8, 9]$	0	1	0	$(\neg 0 \rightarrow \neg 0) \wedge (0 \rightarrow 1) = (1 \rightarrow 1) \wedge (0 \rightarrow 1) = 1 \wedge 1 = 1$
$[9, 11]$	0	0	0	$(\neg 0 \rightarrow \neg 0) \wedge (0 \rightarrow 0) = (1 \rightarrow 1) \wedge (0 \rightarrow 0) = 1 \wedge 1 = 1$
$[11, 13]$	1	0	0	$(\neg 1 \rightarrow \neg 0) \wedge (0 \rightarrow 0) = (0 \rightarrow 1) \wedge (0 \rightarrow 0) = 1 \wedge 1 = 1$
$[13, 18]$	1	1	0	$(\neg 1 \rightarrow \neg 0) \wedge (0 \rightarrow 1) = (0 \rightarrow 1) \wedge (0 \rightarrow 1) = 1 \wedge 1 = 1$
$[18, \infty]$	1	1	1	$(\neg 1 \rightarrow \neg 1) \wedge (1 \rightarrow 1) = (0 \rightarrow 0) \wedge (1 \rightarrow 1) = 1 \wedge 1 = 1$

Здесь тоже есть один интервал ( $[7, 8]$ ), где заданное логическое выражение обращается в нуль. Значит, отрезок  $Z = [8, 18]$  не является правильным ответом.

Ответ:  $[6, 14]$  (ответ №3).

**Комментарий.** Данный способ решения очень громоздок. Правда, можно сократить “рутинную” работу по вычислению значения логического выражения для каждого интервала, если замечать, что в очередной таблице для другого отрезка  $Z$  меняются не все наборы значений логических переменных, соответствующих используемым фактам. Кроме того, даже для изменившихся наборов значений этих логических переменных можно поискать в других, ранее составленных таблицах “готовое” значение логического выражения. Но все же расписывание подобных таблиц для каждого интервала займет немало времени, а вероятность ошибки в вычислениях достаточно высока.

Существует и другой способ решения, основанный на логических рассуждениях, который может оказаться для кого-то из учащихся сложноватым, но “сильным” ученикам поможет довольно существенно сократить время решения.

*Решение (способ 2)*

Сначала преобразуем заданные формулы, чтобы избавиться от логической операции НЕ. Сделать это нетрудно — нужно лишь заменить знак принадлежности отрезку на “обратный” знак непринадлежности и наоборот:

$$\begin{aligned} \neg(t \in X) \rightarrow \neg(t \in Z) &\Rightarrow (t \in X) \rightarrow (t \in Z) \\ (t \in Z) \rightarrow (t \in Y) &\Rightarrow (t \in Z) \rightarrow (t \in Y) \end{aligned}$$

А теперь немного порассуждаем, что означает такая запись, учитывая, что по условию оба заданных логических выражения должны быть истинными одновременно.

Первое выражение:  $(t \in X \rightarrow t \in Z)$ , если вспомнить таблицу истинности операции следования, можно понимать так:

- если  $t$  принадлежит отрезку  $X$ , то  $t$  обязательно должно принадлежать и отрезку  $Z$  ( $1 \rightarrow 1 = 1$ , а  $1 \rightarrow 0 = 0$ ); если же  $t$  не принадлежит отрезку  $X$ , то принадлежность  $t$  отрезку  $Z$  нам совершенно безразлична (независимо ни от чего результатом операции следования будет  $1: 0 \rightarrow 0 = 1$  и  $0 \rightarrow 1 = 1$ ).

Второе выражение:  $(t \in Z) \rightarrow (t \in Y)$  аналогичным образом можно понимать так:

- если  $t$  не принадлежит отрезку  $Z$ , то  $t$  обязательно не должно принадлежать и отрезку  $Y$  ( $1 \rightarrow 1 = 1$ , а  $1 \rightarrow 0 = 0$ ); если же  $t$  принадлежит отрезку  $Z$ , то принадлежность  $t$  отрезку  $Y$  нам безразлична ( $0 \rightarrow 0 = 1$  и  $0 \rightarrow 1 = 1$ ).

Очевидно, из этих рассуждений можно выделить два “ключевых” критерия, которые определяют возможность для того или иного отрезка быть правильным ответом:

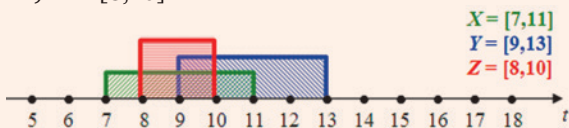
- если  $t$  принадлежит отрезку  $X$ , то  $t$  обязательно должно принадлежать и отрезку  $Z$ ;
- если  $t$  не принадлежит отрезку  $Z$ , то  $t$  обязательно не должно принадлежать и отрезку  $Y$ .

Тем самым мы существенно упростили себе задачу анализа заданных отрезков: теперь нам уже не нужно будет просчитывать значение логических выражений для каждого “неизменного интервала” (а их, вспомним, мы выделяли на числовой прямой семь штук!). Достаточно только быстро, “на глазок”, по полученным словесным описаниям проверить: есть ли в получившейся картине расположения отрезков на числовой прямой такие участки, где нарушаются указанные выше “ключевые” правила, т.е.:

- если  $t$  принадлежит отрезку  $X$  и  $t$  не принадлежит отрезку  $Z$
- или
- если  $t$  не принадлежит отрезку  $Z$  и  $t$  принадлежит отрезку  $Y$ .

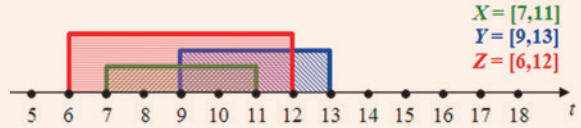
В нашей “цветовой гамме” при рисовании числовых прямых это означает, что мы ищем участки, где зеленая штриховка не пересекается с красной или где синяя штриховка не пересекается с красной.

1)  $Z = [8,10]$



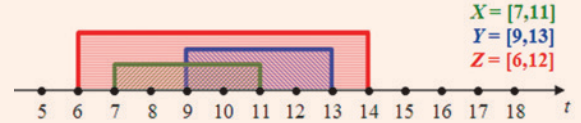
По крайней мере интервалы  $[7,8]$  и  $[11,13]$  нарушают правила (это сразу видно); кроме того, нарушение правил есть и на интервалах  $[10,11]$  и  $[10,13]$ . Значит, отрезок  $Z = [8,10]$  не является правильным ответом.

2)  $Z = [6,12]$



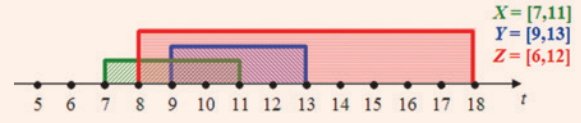
Здесь второе правило нарушено для интервала  $[12,13]$ , и этого уже достаточно, чтобы отрезок  $Z = [6,12]$  не мог быть правильным ответом.

3)  $Z = [6,14]$



В этом случае нарушений правил нигде нет: красная штриховка везде “с лихвой” закрывает собой и синюю, и зеленую. Значит, отрезок  $Z = [6,14]$  — это правильный ответ.

4)  $Z = [8,18]$



А здесь уже первое правило нарушено для интервала  $[7,8]$ , и этого тоже достаточно, чтобы отрезок  $Z = [8,18]$  не был правильным ответом.

Итак, благодаря предварительно сделанным логическим рассуждениям мы определили “правильный” отрезок, вообще не выполняя никаких логических вычислений!

Ответ:  $[6,14]$  (ответ №3).

**Комментарий.** Каким из двух приведенных способов воспользоваться, лучше всего если каждый ученик решит для себя сам. А если в результате выйдут несколько “правильных” отрезков, то определить среди них самый длинный будет уже нетрудно...

## 7. Еще один Робот

Ну, нравятся разработчикам заданий ЕГЭ работы, “ползающие” в клетчатых лабиринтах ☺. Тем более что достаточно лишь немного изменить алгоритм работы такого Робота, и решение задачи может существенно измениться. Покажем это.

Система команд Робота, “живущего” в прямоугольном<sup>3</sup> лабиринте на клетчатой плоскости, состоит из восьми команд.

Четыре команды — это приказы: **вверх**, **вниз**, **влево**, **вправо**. При выполнении любой из таких

<sup>3</sup> На самом деле все лабиринты, которые встречались авторам в таких задачах, — квадратные. Но, во-первых, как мы помним из геометрии, квадрат — это частный случай прямоугольника, а во-вторых, никто не гарантирует, что в будущем в задачах про Робота его “поле деятельности” действительно не окажется неравносторонним...

команд Робот перемещается на одну клетку в соответствующем направлении (вверх — ↑, вниз — ↓, влево — ←, вправо — →).

Еще четыре команды проверяют наличие стенок у каждой из сторон той клетки, в которой находится Робот<sup>4</sup>: **сверху свободно**, **снизу свободно**, **слева свободно**, **справа свободно**. При наличии стенки с соответствующей стороны клетки подобная команда возвращает значение **ЛОЖЬ**, а при отсутствии стенки — **ИСТИНА**.

В алгоритмической конструкции:

**ЕСЛИ условие**

**ТО команда 1**

**ИНАЧЕ команда 2**

**КОНЕЦ ЕСЛИ**

выполняется команда 1, если условие истинно, и команда 2, если условие ложно.

Конструкция

**ПОКА условие**

...

**КОНЕЦ ПОКА**

означает циклическое выполнение вложенных в нее команд, пока условие является истинным.

В конструкциях **ПОКА** и **ЕСЛИ условие** может содержать команды проверки, а также слова **И**, **ИЛИ**, **НЕ**, которые обозначают соответствующие логические операции.

Если Робот будет двигаться на находящуюся рядом с ним стенку, то он разрушится, и программа прервется.

Сколько клеток лабиринта соответствуют требованию, что, начав движение из данной клетки и выполнив предложенную программу, Робот уцелеет и остановится в заштрихованной клетке F6?

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

НАЧАЛО

**ПОКА снизу свободно ИЛИ справа свободно**

**ПОКА справа свободно**

**вправо**

**КОНЕЦ ПОКА**

**вниз**

**КОНЕЦ ПОКА**

**КОНЕЦ**

1) 7    2) 15    3) 29    4) 32

<sup>4</sup> Важно понимать, что такая проверка производится именно для клетки с Роботом, а не относительно текущей ориентации самого Робота в клетке (т.е. в данной задаче не учитываются повороты Робота вокруг своей оси — он просто смещается при движении на одну клетку в соответствующем направлении, и все).

*Решение*

Проанализировав программу, можно определить, что все перемещение Робота по лабиринту состоит из ряда одинаковых “этапов”, где каждому этапу соответствует один проход внешнего цикла **ПОКА**. При этом каждый такой “этап” начинается с проверки отсутствия стенок снизу ИЛИ справа, — значит, Робот остановится (и корректно завершит программу) в клетке, имеющей две стенки — И внизу, И справа (┘).

А что происходит в пределах каждого такого “этапа”? Внутри внешнего цикла **ПОКА** вложены:

- еще один цикл **ПОКА**, в котором Робот двигается вправо, пока не встретит справа стенку;
- единственную команду **вниз**, выполняемую один раз без каких-либо проверок (т.е. при этом, если снизу есть стенка, то Робот может разбиться!).

Итак, получается, что наш Робот “ходит буквой Г”: от текущей клетки, если в ней нет стенок снизу и справа, Робот идет вправо до первой встреченной стенки, а потом пытается сделать шаг на одну клетку вниз. Если Робот при этом не разбился, то затем он повторяет “букву Г” снова, и т.д. При этом очевидно, что все клетки, в которых Робот временно приостанавливается после очередного хода “буквой Г”, имеют одинаковый “статус”: если хотя бы одна из таких клеток удовлетворяет условию задачи, то удовлетворяют ему и все остальные такие клетки, и наоборот. Будем называть такие клетки “опорными”.

Ну что же, — начнем анализ.

1) Проверяем клетку A1. Она “опорная”, так как с нее начинается выполнение программы, а значит, и первого “этапа”. Так как в ней нет стенок ни снизу, ни справа, Робот начнет очередной “этап”: сначала пойдет вправо “до упора” (т.е. до клетки F1), а затем делает шаг вниз и попадает в клетку F2.

Эта клетка — тоже “опорная”. С нее начинается новый “этап” движения робота, так как в клетке F2 по крайней мере нет стенки снизу. Но движения вправо не будет: при проверке выяснится, что справа уже есть стенка. Поэтому Робот только сделает свой единственный шаг вниз и попадет в клетку F3 — она тоже станет “опорной”.

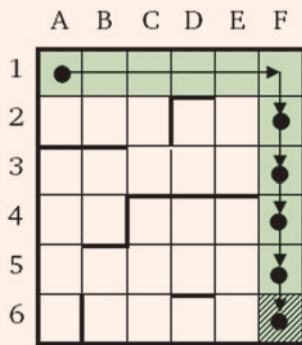
Очевидно, что далее Робот так и будет по одной клетке переходить вдоль столбца F вниз, при этом все клетки этого столбца становятся “опорными”. А когда Робот окажется в “целевой” клетке F6, программа завершится, так как у этой клетки есть стенки и снизу, и справа.

Кстати, и сама клетка F6 тоже нам годится: если Робот изначально уже в ней находился, то он просто никуда не пойдет (программа завершится сразу же после ее начала) и останется в F6, что тоже удовлетворяет условию задачи.

Таким образом, Робот выполнил условия задачи. Следовательно, все пройденные “опорные” клетки удовлетворяют условию задачи. И более того — годятся нам и все клетки строки 1 (не опорные): ведь

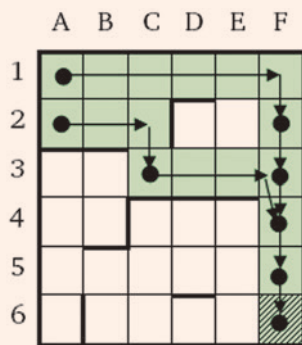
понятно, что, начиная движение с любой из них, Робот так или иначе выйдет на тот же самый путь (вплоть до клетки F1: из нее Робот не пойдет вправо из-за стенки, а сразу перейдет на один шаг вниз и попадет в ту же “опорную” точку F2).

Отметим эти клетки зеленым цветом.

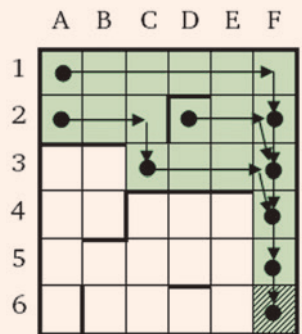


2) Начинаем проверку с клетки A2. В ней нет стенки справа (хотя снизу есть). Поэтому Робот начнет движение. Он пойдет вправо до стенки (т.е. до клетки C2), а потом сделает шаг вниз и попадет в клетку C3. Это — “опорная” точка.

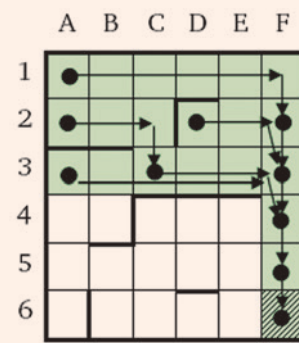
Из нее на следующем “этапе” Робот пойдет вправо до упора, сделает шаг вниз и... попадет в уже проверенную нами “опорную” клетку F4. Значит, “опорные” клетки A2, C3, а также все пройденные Роботом “не опорные” клетки в строках 2 и 3 нам тоже годятся. Отмечаем их тоже зеленым цветом.



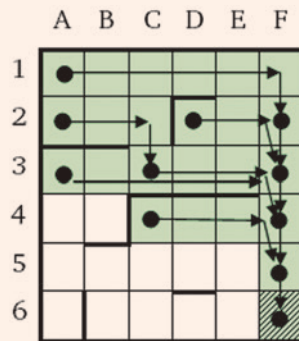
3) Не менее очевидно, что годятся и клетки D2 и E2: если Робот начинает движение с них, то тоже попадет в уже проверенную “опорную” точку F3.



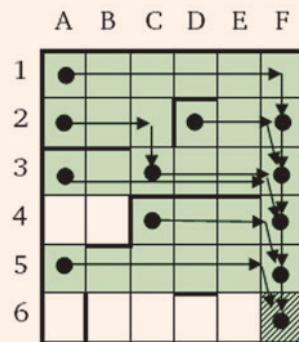
4) Если Робот начинает движение из клеток A3 или B3, то он первым ходом пойдет вправо до столбца F, а сделав шаг вниз, попадет в уже проверенную “опорную” точку F4. Вывод: указанные клетки тоже годятся.



5) То же касается клеток C4 — E4: при начале движения из них Робот дойдет до столбца F, сделает шаг вниз и попадет в проверенную “опорную точку” F5.

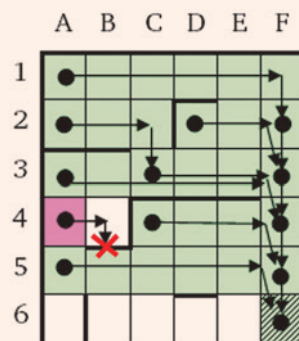


6) Аналогично, все клетки строки 5 тоже годятся: из любой из них Робот пройдет вправо до столбца F и, сделав шаг вниз, попадет в искомую клетку F6, где и останется.



7) А если Робот начнет двигаться с клетки A4? Очевидно, он пройдет вправо до стенки и окажется в клетке B4. Но затем Робот попытается без всяких проверок сделать шаг вниз — и разобьется о стенку! Значит, клетка A4 не удовлетворяет условиям задачи!

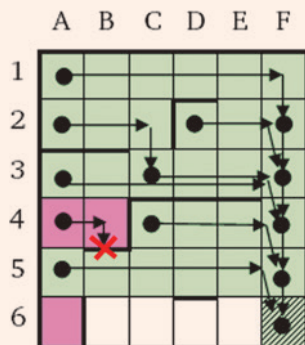
Поэтому отметим ее розовым цветом.



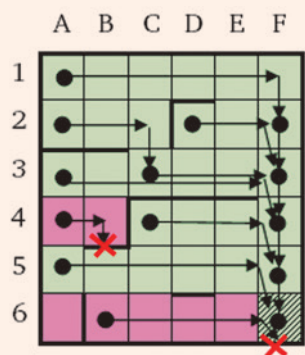
8) Если Робот изначально находится в клетке В4, то он никуда не пойдет: в этой клетке есть стенки И справа, И снизу. Но это — не та целевая клетка, которая нам нужна по условию (F6). Поэтому клетка В4 не удовлетворяет условию задачи.

И то же самое касается клетки А6.

Помечаем обе эти клетки розовым цветом.



9) Наконец, рассмотрим клетки нижней строки В6 — Е6. Если Робот начнет движение с любой из них, то дойдет вправо до клетки F6. Но в ней он не остановится! Он должен будет сделать еще один шаг вниз — и разобьется. Значит, ни одна из этих клеток нам не годится.



Подведем итоги.

Розовым цветом у нас оказалось помечено 7 клеток. А зеленым —  $36 - 7 = 29$  клеток.

Это и есть правильный ответ.

**Ответ:** 29 (ответ №3).

**Комментарий.** Решая задачу про Робота в лабиринте, внимательно анализируйте программу Робота и определяйте, как движение Робота разбивается на “этапы” по внешнему циклу или условному оператору (это поможет определять “опорные” точки), какие клетки он проходит при движении и меняется ли что-то, если он будет начинать движение с них, а также есть ли в программе Робота потенциально опасные команды, выполняемые без проверки наличия стенок по ходу движения.

## 8. “Отнять и поделить”

Перейдем теперь к задачам группы В. И первая из них — в общем-то хорошо нам знакомая задача на исполнителей.

У исполнителя есть две команды, которые кодируются своими номерами:

1. вычесть 3

2. делить на 2

Выполняя первую из них, исполнитель вычитает из текущего числа число 3. Выполняя вторую команду, исполнитель делит текущее число пополам (но если деление нацело невозможно, то выполнение программы прекращается).

Требуется записать программу, состоящую из номеров соответствующих команд, содержащую не более пяти команд и преобразующую число 186 в число 39. При этом номера команд записываются в нужном порядке подряд, без пробелов. Если возможно несколько таких программ, то запишите любую из них.

Например, для программы:

**вычесть 3**

**разделить на 2**

**вычесть 3**

надо записать программу в виде: **121**. Такая программа, например, преобразует число 37 в число 14.

*Решение*

Подобные задачи мы уже рассматривали в одном из предыдущих номеров журнала “Информатика” (см.: Богомолова О.Б., Усенков Д.Ю. “Каверзная” задача про автомат // Информатика, 2013, № 7–8. С. 40–41). И там мы рекомендовали начинать решение “с конца” — сначала заменить заданные команды на обратные им и найти программу, которая приводит от требуемого числа к исходному, — а затем записать номера “прямых” команд в обратном порядке по отношению к найденному.

Так было во всех предыдущих таких задачах. Но вот теперь нам встретилась первая задача, которую *нужно решать сразу напрямую, не заменяя команды на обратные!*

Почему мы ранее использовали “обратные” команды и почему этого не нужно делать сейчас?

Ранее в подобных задачах на исполнителя его команды были такими: “сложить”, “вычесть”, “умножить”, а иногда — “возвести в квадрат”. Работать с такими командами сложно: имея некоторое “промежуточное” число, трудно предугадать, какую команду лучше всего к нему применить, чтобы дойти до результата (разве что, зная, что программа должна быть короткой, можно догадаться, что надо стараться использовать умножение и возведение в квадрат, а не сложение, чтобы быстрее дойти до цели). А вот если заменить такие команды на “обратные”, то получившиеся команды деления и вычисления квадратного корня сразу становились подсказкой: если текущее число делится нацело или из него можно извлечь целый корень, то это означает, что надо использовать именно такую команду, а если деление или извлечение корня нацело не удастся, то использовать “альтернативное” сложение или вычитание.

А что в данной задаче? Здесь у нас уже имеется команда деления. А значит, она станет такой же подсказкой, если мы будем решать задачу напрямую.

186	делить на 2	(число делится нацело)
93	вычесть 3	(не делится нацело на 2)
90	делить на 2	(число делится нацело)
45	вычесть 3	(не делится нацело на 2)
42	вычесть 3	(а тут уже нетрудно догадаться, что нужно не делить на 2, а вычесть 3)
39		

Остается записать в том же порядке номера команд.

Ответ: 21211.

**Комментарий.** Итак, прежде всего нужно смотреть на то, какие у исполнителя команды. Если среди них есть такие команды, которые могут быть выполнены только при определенных условиях (деление нацело, извлечение целого квадратного корня), то задачу надо решать “напрямую”. А если все команды такие, что могут быть выполнены всегда (сложение, умножение, вычитание, возведение в квадрат), то лучше сначала решить задачу с “обратными” командами.

## 9. “Тарабарский язык”

Вновь появилась забытая было задача на списки “слов”, составленных из нескольких возможных букв по определенному правилу, которое и нужно понять для решения такой задачи. “Слова” при этом, правда, выглядят как какая-то “тарабарщина”, но для смысла самой задачи это не важно.

Все шестибуквенные слова, составленные из букв А, Б, В и Г, записаны в алфавитном порядке и пронумерованы. Вот начало такого списка:

1. АААААА
2. АААААБ
3. АААААВ
4. АААААГ
5. ААААБА
- ...

Под каким номером в этом списке располагается первое слово, начинающееся с буквы Г?

*Решение*

Посмотрим на начало списка “слов”. Эти “слова” ничего вам не напоминают?

А если сравнить их со следующими числами?

1. 000000
2. 000001
3. 000002
4. 000003
5. 000010
- ...

Так и есть! Имеется полная аналогия, если буквы А заменять нулем, Б — единицей, В — двойкой, а Г — тройкой. А “слова” в этом случае превращаются в возрастающую с шагом 1 последовательность чисел — в данном случае шестизначных, записанных в четверичной системе (потому как различных букв в “алфавите” используется именно четыре).

Ну а теперь уже решить задачу проще простого.

Какое первое по порядку возрастания шестизначное четверичное число будет начинать-

ся с тройки (т.е. с аналога буквы Г)? Очевидно,  $300000_4$ .

Теперь заметим: в приведенном списке под номером 1 идет число  $000000_4$ , т.е.  $0_{10}$ . Под номером 2 — число  $000001_4$ , т.е.  $1_{10}$ . И так далее. Значит, чтобы найти номер в списке искомого числа  $300000_4$  (а значит, и искомого первого слова, начинающегося с Г), надо преобразовать его в десятичную систему счисления и к полученному десятичному числу прибавить 1:

$$300000_4 = 3 \times 4^5 = 3072_{10};$$

$$3072 + 1 = 3073.$$

Ответ: 3073.

## 10. Шаг за шагом

Задача очень простая. Но, как показала практика, при ее решении очень многие школьники делают ошибки.

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

$$F(1) = 2; F(2) = 3;$$

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

Чему равно значение функции  $F(5)$ ?

*Решение*

Казалось бы, чего проще: мы знаем два первых значения некоторой последовательности чисел, а все остальные вычисляются по заданной рекуррентной формуле (в которой нетрудно узнать формулу последовательности чисел Фибоначчи). То есть нужно поочередно расписывать значения  $F$  для каждого очередного  $n$  и при этом подставлять в формулу известные или ранее вычисленные значения:

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

$$F(4) = F(4-1) + F(4-2) = F(3) + F(2) = 5 + 3 = 8;$$

$$F(5) = F(5-1) + F(5-2) = F(4) + F(3) = 8 + 5 = 13.$$

Ответ: 13.

Однако очень многие учащиеся просто сразу записывают формулу  $F(5) = F(5-1) + F(5-2) = F(4) + F(3)$ , а потом пытаются подставить вместо  $F(4)$  и  $F(3)$ ... числа 4 и 3 соответственно, получая в качестве ответа число 7. Именно на это и требуется обратить внимание ребят при подготовке к ЕГЭ!

А вот еще одна интересная задачка такого же типа:

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

$$F(1) = 3; F(2) = 3;$$

$$F(n) = 8 \cdot F(n-1) - 7 \cdot F(n-2) \text{ при } n > 2.$$

Чему равно значение функции  $F(2014)$ ?

*Решение*

На первый взгляд кажется, что на решение этой задачки придется потратить чуть ли не всю свою жизнь, ☺ — попробуй-ка, распиши по очереди целых 2012 строк для вычисления каждого очередного значения  $F(n)$ ! (Кстати, в аналогичной задаче из комплекта заданий, предложенных СтатГрад, работчики были не столь “кровожадны” и требо-

вали вычислить только значение  $F(13)$ .) Так ли это сложно на самом деле? А вот проверим!

$$F(3) = 8 * F(3-1) - 7 * F(3-2) = 8 * F(2) - 7 * F(1) = 8 * 3 - 7 * 3 = 3;$$

$$F(4) = 8 * F(4-1) - 7 * F(4-2) = 8 * F(3) - 7 * F(2) = 8 * 3 - 7 * 3 = 3; \dots$$

Впрочем, уже здесь можно легко догадаться, что формула для вычисления очередного члена последовательности составлена так, что при одинаковых значениях  $F(1)$  и  $F(2)$  по этой формуле *всегда* будет получаться число 3.

А значит, предложенная авторами задача — во все не “кровожадная” и не сложная, а очень-очень простая. И ответ к ней можно дать сразу же: это (независимо от того, про  $F$  от какого  $n$  в ней спрашивается) всегда будет число 3.

*Ответ: 3.*

**Комментарий.** При подготовке к ЕГЭ для успешного решения таких задач нужно раз и навсегда запомнить: подобные вычисления  $F(n)$  через предыдущие значения ( $n-1$ ,  $n-2$  и т.д.) нужно именно так и вычислять — пошагово, постепенно переходя от меньших значений  $n$  ко все большим. И, конечно, следить за получаемыми на первых же строках результатами: вдруг там отыщется какая-то закономерность, которая или даст возможность вычислить требуемое значение  $F$  более быстро (по формуле, а не рекуррентным способом), или вообще позволит дать ответ сразу.

## 11. Что печатает программа

И с такими задачами учащиеся (и учителя, которые их готовят к сдаче ЕГЭ) вроде бы уже знакомы. Вот, например, типичное задание из демоварианта ЕГЭ (для экономии места здесь дается только текст программы на языке Паскаль; в исходном задании ЕГЭ программа также продублирована на языках Бейсик, Си и “алгоритмическом языке”, чтобы каждый школьник мог выбрать тот язык, который он изучал).

Ниже записан алгоритм. Получив на вход число  $x$ , этот алгоритм печатает два числа:  $a$  и  $b$ . Укажите наименьшее из таких чисел  $x$ , при вводе которых алгоритм печатает сначала 15, а потом 2.

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

Здесь, анализируя текст программы, можно заметить, что “ключевым” в ней является цикл `while`. Он выполняется, пока введенное (неизвестное нам) число  $x$  еще остается больше нуля, а в теле этого цикла производятся следующие действия:

- значение  $b$  увеличивается на 1 (т.е., по сути,  $b$  — это счетчик количества проходов цикла);
- остаток от деления  $x$  на 10 (т.е. последняя цифра числа!) записывается в  $c$ ;
- выделенная из числа последняя цифра умножается на значение переменной  $a$  и результат снова заносится в  $a$ ;
- число  $x$  нацело делится на 10 и результат снова заносится в переменную  $x$  (что, по сути, означает отбрасывание в числе  $x$  его последней цифры).

А после завершения цикла выводится сначала значение переменной  $a$  (накопленное произведение цифр исходного числа  $x$ ), а потом значение переменной  $b$ , которое указывает, сколько раз был выполнен цикл (то есть, очевидно, указывает количество цифр в исходном числе).

А дальше — уже несложно. Мы знаем, что  $a = 15$ , а  $b = 2$ . Значит, число  $x$  было двузначное, а произведение его цифр равно 15. Какие это могут быть две цифры? Разложив число 15 на простые множители, получаем: 3 и 5. А какое двузначное число, состоящее из цифр 3 и 5, является наименьшим? Очевидно, 35! Это и есть ответ.

*Ответ: 35.*

Однако дальше пошли усложнения. Уже в демоварианте ЕГЭ-2014 задание стало таким (изменения в тексте программы на Паскале в задании выделены жирным шрифтом).

Ниже записан алгоритм. Получив на вход число  $x$ , этот алгоритм печатает два числа:  $a$  и  $b$ . Укажите наименьшее из таких чисел  $x$ , при вводе которых алгоритм печатает сначала 13, а потом 5.

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

*Решение*

Очевидно, здесь тоже из числа  $x$  выделяется последняя цифра, а в переменной  $a$ , таким образом, накапливается сумма цифр исходного числа, после чего в числе  $x$  очередная последняя цифра отбрасывается. То есть и здесь программа выполняет “разборку” заданного числа на отдельные цифры. Но, как нетрудно видеть, здесь нет “обычного” счетчика проходов цикла. Зато появился “какой-то” оператор `if`. Что он делает?

В этом условном операторе выполняется проверка: если только что выделенная цифра меньше текущего значения  $b$  (а вначале так и будет: исходное  $b = 10$  и больше, чем любая возможная десятичная цифра), то эта цифра перезапоминается в  $b$ . А дальше такое перезапоминание будет производиться, только если очередная (т.е. более “старшая” по разрядам) цифра числа  $x$  будет



меньше, чем более “младшие” (запомненные в  $b$  ранее).

Теперь смотрим, что программа выводит на экран. Сначала это число  $a$  — сумма цифр числа  $x$ , равная 13. А потом — число  $b$ , указывающее, что одна из этих цифр — цифра 5, причем она — наименьшая из всех цифр в числе  $x$  (однако количество этих цифр нам неизвестно).

Можно ли по этим данным определить число  $x$ ? Да, можно! Если одна из цифр равна 5, то, вычтя ее из суммы (13), мы получаем число 8. Это — сумма всех остальных цифр числа  $x$ . Какими же они могут быть и сколько их может быть? Перебираем варианты:

- $1 + 7$  — невозможно: тогда в  $b$  была бы запомнена наименьшая по величине цифра 1, а не 5; то же касается и вариантов  $2 + 6$ ,  $3 + 5$ ,  $4 + 4$  и всех им “симметричных” (например,  $5 + 3$ ) — во всех них есть хотя бы одна цифра, меньшая 5;

- $1 + 1 + 6$ ,  $1 + 2 + 5$  и все прочие возможные варианты с тремя цифрами тоже невозможны по той же причине;

- очевидно, точно так же невозможны и любые другие варианты, в которых остаток суммы, равный 8, раскладывается на несколько цифр.

Следовательно, остается одна-единственная возможность: число  $x$  — двузначное и содержит цифры 5 и 8.

А какое из подобных чисел — наименьшее? Очевидно, 58 (а не 85)!

*Ответ:* 58.

А теперь посмотрим, что нам заготовили в тренировочной работе СтатГрад (напомним, что здесь рассматриваются не в точности задания из указанной тренировочной работы, а однотипные с ними).

Ниже записан алгоритм. После выполнения алгоритма было напечатано 3 числа. Первые два напечатанных числа — это числа 7 и 35. Какое наибольшее число может быть напечатано третьим?

```
var x, y, z: integer;
    r, a, b: integer;
begin
  readln(x, y);
  if y > x then begin
    z := x; x := y; y := z; end;
  a := x; b := y;
  while b > 0 do begin
    r := a mod b;
    a := b;
    b := r;
  end;
  writeln(a); writeln(x); write(y);
end.
```

*Решение*

Как видим, алгоритм здесь совершенно другой: хотя в теле цикла и остается оператор вычисления остатка от деления, но это уже вовсе не разбиение числа на цифры! А что же?

Анализируем текст программы с самого начала.

Первый ключевой момент здесь — это условный оператор:

```
if y > x then begin z := x; x := y; end;
```

Не очень трудно догадаться, что он сортирует заданные числа  $x$  и  $y$  по убыванию: какие бы два числа ни вводились, после выполнения этого оператора всегда в  $x$  будет большее из двух введенных чисел, а в  $y$  — меньшее. Для чего это нужно? Смотрим дальше.

Далее числа  $x$  и  $y$  дублируются в переменные  $a$  и  $b$ . Благодаря этому можно затем произвольно изменять значения  $a$  и  $b$ , но сохранить их исходные значения (в  $x$  и  $y$ ), которые нужны в конце программы для их вывода на экран.

И наконец, самое интересное — цикл `while`:

```
while b > 0 do begin
  r := a mod b;
  a := b;
  b := r;
end;
```

Пока понять его смысл достаточно трудно. Но попробуем записать запрограммированный здесь алгоритм словесно:

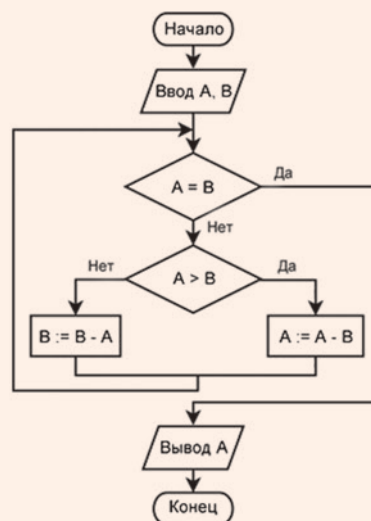
- вычисляется остаток от деления большего числа ( $a$ ) на меньшее ( $b$ ), этот остаток запоминается в  $r$ ;

- большее из бывших двух чисел заменяется на меньшее, а бывшее меньшее — на вычисленный остаток от деления;

- и так — до тех пор, пока получаемое при этом меньшее из двух чисел (т.е., по сути, последний вычисленный остаток от деления большего числа на меньшее) не станет равным нулю (меньше нуля остаток от деления, очевидно, быть не может).

По завершении цикла выводится значение  $a$ , которое равно бывшему на предыдущем шаге цикла меньшему числу.

А теперь рассмотрим следующий алгоритм (в виде блок-схемы):



Этот алгоритм должен знать каждый школьник! Это знаменитый алгоритм вычисления НОД (наибольшего общего делителя) двух чисел — алгоритм Евклида, описанный этим греческим ученым в его книге “Начала”.

Как он работает?

Берутся два числа, для которых требуется вычислить НОД.

Если они равны, то любое из них и есть НОД.

Иначе из большего числа вычитается меньшее. Такое вычитание будет производиться до тех пор, пока полученная разность не окажется меньше, чем меньшее из имевшихся чисел, тогда далее эти числа меняются местами: бывшее меньшее становится новым большим, а полученная разность становится меньшим числом. И так, пока числа не сравняются.

Это — “классический” алгоритм Евклида для нахождения НОД, основанный на вычитаниях. Но можно его и ускорить, реализовав на основе делений с остатком.

Действительно: вместо целой цепочки вычитаний из большего числа меньшего до тех пор, пока разность не станет меньше меньшего из чисел, можно заменить делением большего числа на меньшее с остатком. Например, цепочка вычитаний:

$$14 - 3 = 11; 11 - 3 = 8; 8 - 3 = 5; 5 - 3 = 2$$

дает тот же результат, что и деление с остатком:

$$14 \bmod 3 = 2.$$

А сравнение двух обрабатываемых чисел между собой вполне можно заменить на проверку равенства нулю остатка от их деления друг на друга.

На этих рассуждениях можно сформулировать следующий вариант алгоритма Евклида для нахождения НОД двух чисел:

- берется два натуральных числа  $A$  и  $B$ , где  $A > B$ ;
- вычисляется остаток от их деления друг на друга;
- большее число заменяется остатком;
- эти действия повторяются, пока не получится нулевой остаток; тогда предыдущий ненулевой остаток и будет являться НОД исходных двух чисел.

*Пример:*

$$\text{НОД}(14,35): 35 \bmod 14 = 7; \Rightarrow$$

$$\text{НОД}(14,7): 14 \bmod 7 = 0; \Rightarrow \text{НОД}(14,35) = 7.$$

У этого алгоритма есть только один недостаток: очевидно, что получаемый остаток будет меньше, чем меньшее из двух делящихся чисел (делитель), так что каждый раз надо сравнивать числа и менять их местами, чтобы первое (делимое) было больше, чем второе (делитель). То есть эту операцию проверки и обмена чисел местами надо включать в цикл. А можно ли обойтись без этого?

Будем после вычисления остатка от деления большего числа на меньшее записывать бывшее меньшее взамен большего, а остаток — взамен бывшего меньшего. Смысла алгоритма это, очевидно, не изменит: все равно на очередном шаге цикла будет выполняться деление оставшегося меньшего числа предыдущей пары на вычисленный остаток. Но зато мы будем уверены, что у нас всегда первое число будет больше второго!

Получаем следующий “модифицированный” алгоритм Евклида с делением.

Берется два натуральных числа  $A$  и  $B$ , где  $A > B$ . Затем в цикле:

- вычисляется остаток от деления  $A$  на  $B$ ;
- большее число заменяется меньшим;
- меньшее число заменяется остатком;

• эти действия повторяются, пока не получится нулевой остаток; тогда предыдущий ненулевой остаток и будет являться НОД исходных двух чисел.

А теперь вернемся к той словесной записи алгоритма заданной программы, которую мы сделали ранее:

- вычисляется остаток от деления большего числа ( $a$ ) на меньшее ( $b$ ), этот остаток запоминается в  $r$ ;
- большее из бывших двух чисел заменяется на меньшее, а бывшее меньшее — на вычисленный остаток от деления;

• и так — до тех пор, пока получаемое при этом меньшее из двух чисел (т.е., по сути, последний вычисленный остаток от деления большего числа на меньшее) не станет равным нулю.

По завершении цикла выводится значение  $a$ , которое равно меньшему числу на данном шаге цикла и которое, в свою очередь, равно остатку от деления на предыдущем шаге цикла.

Легко видеть: здесь не просто аналогия, а практически совпадение один в один!

Таким образом, в заданном тексте программы фрагмент:

```
while b > 0 do begin
  r := a mod b;
  a := b;
  b := r;
end;
```

— это вычисление НОД двух чисел  $a$  и  $b$  (т.е., по сути, чисел  $x$  и  $y$ ), а в конце программы сначала выводится найденное значение НОД, а потом — сами исходные числа.

Программа, согласно условию, выводила числа 7 и 35 плюс какое-то неизвестное третье число, которое предлагается определить. Значит, в данном случае НОД равен 7, а одно из чисел равно 35:  $\text{НОД}(35, x) = 7$ . Тогда искомое второе число  $x$  тоже должно делиться на 7, но не делиться на 35 его наибольший из существующих делителей. Конечно, таких чисел много (например, то же число 7). Но нам по условию требуется наибольшее из таких чисел, меньшее 35. Очевидно, это число 28.

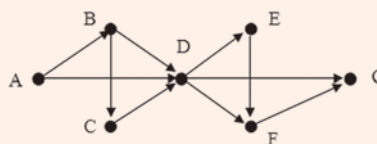
*Ответ: 28.*

**Комментарий.** Очень желательно на уроках информатики познакомиться хотя бы с основными вычислительными алгоритмами и понять принципы их работы. Не раз пригодится!

## 12. Мы едем, едем, едем

Задача на граф в работе СтатГрад оказалась с небольшой “хитринкой” (впрочем, принцип решения не меняющей).

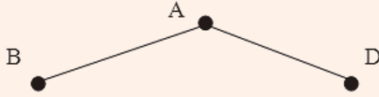
На рисунке изображена схема дорог между городами. По каждой дороге можно двигаться только по стрелке. Сколько существует различных путей из города  $A$  в город  $G$ ?



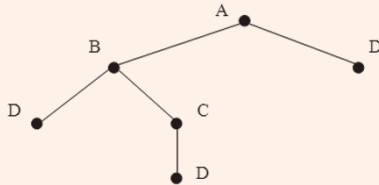
### Решение

Подобные задачи проще всего решать, выстраивая “дерево путей”.

1. Из А можно попасть в В или в D:

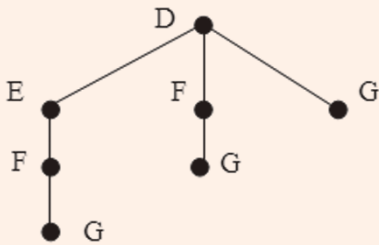


2. Из В можно попасть в D или в C, а из C, в свою очередь, опять-таки в D:



3. Дальше из каждой вершины D надо вычертить три пути — к E, F и G, потом везде прочертить от E пути к F, а затем от F везде прочертить пути к G. Тогда останется только подсчитать в полученном дереве количество отмеченных вершин G, и это количество как раз и дает нам искомое количество различных путей из А в G.

Единственный недостаток такого способа — то, что дерево получается довольно “раскидистым”, рисовать его неудобно. А кроме того, легко заметить, что для каждой вершины D в дереве, построенном нами на шаге 2, мы далее строим совершенно одинаковые продолжения. А значит, можно существенно сократить себе затраты времени и сил, вычертив поддерево путей от D к G только один раз, подсчитав в нем количество вхождений вершины G, а потом (что вполне очевидно) умножив это количество на количество вхождений в ранее построенную часть дерева путей вершины D:



В данном случае в первом поддереве (см. шаг 2) вершина D повторяется 3 раза, а во втором поддереве вершина G повторяется тоже 3 раза. Значит, общее количество путей из А в G равно  $3 \times 3 = 9$ .

Ответ: 9.

**Комментарий.** Такую хитрость с разделением дерева путей на два отдельных поддерева можно применять всегда, когда в исходной схеме дорог где-то в середине есть точка, к которой сходятся все пути левой части схемы.

## 13. Плотнo? Еще плотнее!

Задачи на передачу информации, когда файл можно передавать по каналу связи с заданной скоростью или напрямую, или в архиве (затратив, соответственно, некоторое время на сжатие и распаковку, но вы-

играв за счет уменьшения объема передаваемой “порции” информации), и надо указать, какой способ быстрее и насколько, уже нам знакомы (см., например: Богомолова О.Б., Усенков Д.Ю. Сжимать или не сжимать — вот в чем вопрос // Информатика, 2012, № 2. С. 20–21). Но в заданиях проверочной работы эта задача получила “новое звучание”.

Документ объемом 50 Мб можно передать с одного компьютера на другой двумя способами:

А) сжать архиватором, передать по каналу связи архив, распаковать его;

Б) сжать суперархиватором, передать суперархив по каналу связи, распаковать его.

При этом:

- скорость передачи данных по каналу связи составляет  $2^{22}$  бит/с;

- объем архива составляет 75% от исходного файла;

- объем суперархива составляет 50% от исходного файла;

- сжатие документа архиватором занимает 15 с, а распаковка — 10 с;

- сжатие документа суперархиватором занимает 30 с, а распаковка — 15 с.

Какой способ быстрее и насколько?

В ответе надо записать сначала букву, обозначающую соответствующий способ, а затем сразу после нее записать количество секунд, на сколько этот способ быстрее другого. Например, если способ А быстрее способа Б на 30 секунд, то надо записать ответ в виде А30.

Решение

Собственно, решается данная задача аналогично ее более ранним вариациям, — только здесь проще всего честно вычислить время, затрачиваемое на сжатие, передачу и распаковку информации в каждом из двух способов, а потом сравнить результаты.

А) Тратим 25 секунд на сжатие/распаковку, а затем передаем 75% от 50 Мб со скоростью  $2^{22}$  бит/с:

$$\begin{aligned} & 25 \text{ (с)} + 0,75 \cdot 50 \text{ (Мб)} / 2^{22} \text{ (бит/с)} = \\ & = 25 \text{ (с)} + \frac{3}{4} \cdot 50 \cdot 2^{23} \text{ (бит)} / 2^{22} \text{ (бит/с)} = \\ & = 25 + \frac{3}{2^2} \cdot 25 \cdot 2 \cdot 2^{23} / 2^{22} = 25 + \frac{3 \cdot 25 \cdot 2^{24}}{2^{24}} = \\ & = 25 + 3 \cdot 25 = 25 + 75 = 100 \text{ (с)}. \end{aligned}$$

Б) Тратим 45 секунд на сжатие/распаковку, а затем передаем 50% от 50 Мб со скоростью  $2^{22}$  бит/с:

$$\begin{aligned} & 45 \text{ (с)} + 0,5 \cdot 50 \text{ (Мб)} / 2^{22} \text{ (бит/с)} = \\ & 45 \text{ (с)} + \frac{1}{2} \cdot 50 \cdot 2^{23} \text{ (бит)} / 2^{22} \text{ (бит/с)} = \\ & = 45 + \frac{1}{2} \cdot 25 \cdot 2 \cdot 2^{23} / 2^{22} = 45 + \frac{25 \cdot 2^{24}}{2^{23}} = \\ & = 45 + 2 \cdot 25 = 45 + 50 = 95 \text{ (с)}. \end{aligned}$$

Сравнивая способы А и Б, получаем, что быстрее способ Б на 5 секунд.

Ответ: Б5.

**Комментарий.** Как всегда при подобных расчетах, лучше всего стараться все числа представлять как степени двойки или произведение некоторых

нечетных чисел на степени двойки. Это позволит упростить расчеты и снизить вероятность ошибок благодаря тому, что умножения и деления отчасти заменяются на сложения и вычитания степеней двойки.

## 14. В поисках поиска

Задача, встретившаяся в тренировочной работе СтатГрад, предполагала определение количества найденных сайтов для некоторого запроса, притом, что известно количество сайтов, найденных по другим запросам. При этом усложнение задачи состояло в том, что в запросах участвовало не два, а три ключевых слова. Впрочем, это усложнение было только кажущимся: в условии явно было указано, что для одного из составных запросов не было найдено ни одной страницы, и это резко упрощало решение составленных уравнений. В остальном же принципы решения этой задачи идентичны применявшимся для ее прежних вариантов, поэтому подробно рассматривать ее здесь мы не будем.

**Комментарий.** О решении таких задач см. в статье: Богомолова О.Б., Усенков Д.Ю. “В поисках поиска”: задачи ЕГЭ, посвященные поиску информации на сайтах // Информатика, 2011, № 11. С. 17–23.

## 15. Сколько программ?

Это тоже уже ранее встречавшаяся задача. В ней для исполнителя (такого же, как в задаче 8 выше) требовалось найти, сколько существует программ для такого исполнителя, которые преобразуют одно заданное число в другое. Усложнение же заключалось в том, что исполнитель имел уже не две команды, а три.

Впрочем, и это усложнение — только кажущееся. Просто вместо двух слагаемых в записи цепочек, вычисляющих количества программ для получения из исходного значения другого числа, возрастающего каждый раз на 1, будет включаться три. А далее — принцип тот же:  $R$  от первого числа всегда равен единице, а последующие значения  $R$  вычисляются через команды, обратные заданным, причем слагаемые, дающие не целые значения, из рассмотрения сразу исключаются.

**Комментарий.** Принцип решения подобных задач (для исполнителей с двумя командами) подробно рассмотрен в статье: Богомолова О.Б., Усенков Д.Ю. Исполнитель-вычислитель: сложная задача с простым решением // Информатика, 2012, № 8. С. 20–25.

## 16. Опять В15

Завершим статью рассмотрением обязательной для любого ЕГЭ задачи на определение количества решений системы логических уравнений, — знаменитой “задачи В15”. Сложность таких задач заключается в том, что в зависимости от заданных

логических уравнений принцип их решения может сильно различаться. Поэтому остается либо разбить решение всех “типовых” задач, какие попадутся (в расчете, что на ЕГЭ встретится что-то аналогичное), или... учиться рассуждать логически, чтобы суметь решить *любую* подобную задачу (с чем, увы, не все учащиеся справляются).

Поскольку разработчики заданий СтатГрад запретили публиковать их задания, но решение задачи В15 можно рассмотреть только на примере конкретной системы логических уравнений, мы вынуждены выбрать компромисс: рассмотреть упрощенную версию задачи, подобную приводившейся в СтатГрад.

Сколько существует различных наборов значений логических переменных  $x_1 \dots x_6$ , которые удовлетворяют перечисленным ниже условиям:

$$(x_1 \rightarrow x_2) \rightarrow (x_3 \rightarrow x_4) = 1$$

$$(x_3 \rightarrow x_4) \rightarrow (x_5 \rightarrow x_6) = 1$$

В ответе не требуется перечислять сами наборы значений переменных, а нужно указать только количество таких наборов.

*Решение*

Ранее авторы уже писали о решении подобных задач:

- Богомолова О.Б., Усенков Д.Ю. Логические задачи на ЕГЭ: имена и логические выражения // Информатика, 2011, №8. С. 4–12;

- Богомолова О.Б., Усенков Д.Ю. По следам ЕГЭ-2011: новые задачи // Информатика, 2011, № 15. С. 14–19;

- Богомолова О.Б., Усенков Д.Ю. В15: новые задачи и новое решение // Информатика, 2012, № 6. С. 35–40.

Посмотрим теперь, как решается эта.

1. Для упрощения решения выполним подстановку:

$$a_1 = x_1 \rightarrow x_2; a_2 = x_3 \rightarrow x_4; a_3 = x_5 \rightarrow x_6.$$

Тогда исходная система уравнений примет вид:

$$a_1 \rightarrow a_2 = 1$$

$$a_2 \rightarrow a_3 = 1$$

2. Рассмотрим первое уравнение:  $a_1 \rightarrow a_2 = 1$ .

Вспомним, что логическая операция следования дает единицу в трех случаях:  $0 \rightarrow 0$ ,  $0 \rightarrow 1$  и  $1 \rightarrow 1$  и только в одном случае дает нуль:  $1 \rightarrow 0$ . Поэтому данное логическое уравнение дает нам три варианта решения:  $(a_1 a_2) = (00), (01), (11)$ .

3. Рассмотрим теперь второе уравнение:

$$a_2 \rightarrow a_3 = 1.$$

Обратим внимание, что здесь первый операнд — тот же самый, который был вторым операндом в предыдущем уравнении. А значит, он нам уже известен (в трех вариантах) и теперь мы должны к нему добавить возможные значения второго операнда.

При этом если операнд  $a_2$  равен 0, то для него возможно два значения второго операнда  $a_3$ , дающих в результате следования единицу, — это может быть как 0, так и 1. А вот если  $a_2$  равно 1, то для него возможно только одно-единственное значение  $a_3$ , равное 1, — только в этом случае результат следования будет тоже равен 1.

Какой из этого можно сделать вывод? То, что при формировании наборов значений переменных  $a$  мы должны к последней единице дописывать только единицу, а если последней цифрой был нуль, то писать взамен целых два варианта, в одном из которых дописывается 0, а в другом — 1.

В нашем случае это можно выразить схемой:



и т.д. Поскольку все уравнения в системе — типовые, эту схему можно расширять на любое количество уравнений и, соответственно, используемых переменных  $a$ .

В нашем же случае мы получаем следующие наборы значений переменных  $a$ :

$$(a_1 a_2 a_3) = (000), (001), (011), (111).$$

4. А теперь вспомним, что каждая переменная  $a$  — это тоже операция следования:

$$a_1 = x_1 \rightarrow x_2; a_2 = x_3 \rightarrow x_4; a_3 = x_5 \rightarrow x_6.$$

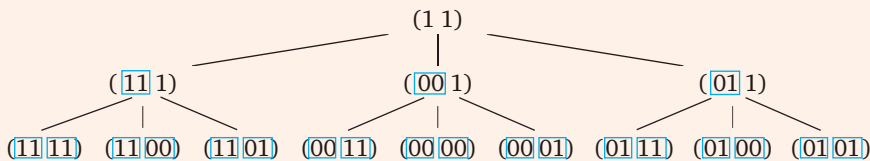
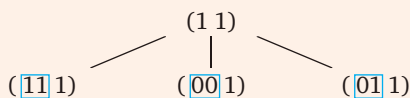
Как теперь перейти от значений  $a$  к значениям  $x$ ?

Это нетрудно, если опять-таки вспомнить, в каких случаях операция следования дает 0, а в каких — 1. Причем сейчас нам даже не важны сами получающиеся при этом значения переменных  $x_1, x_2, x_3, x_4, x_5$  и  $x_6$ . Нам важно лишь то, что для каждого нуля в полученных ранее вариантах значений  $(a_1 a_2 a_3)$  будет только один вариант значений соответствующих переменных  $x$ , а для каждой единицы получается по три разных набора “иксов”. И если единиц в наборе  $(a_1 a_2 a_3)$  несколько, то тройки наборов “иксов” для каждой такой единицы перемножаются.

Понять, почему это именно так, достаточно легко. Предположим, что единиц — две.

$$(1\ 1)$$

Тогда первой соответствует 3 варианта “иксов” и в каждом из них будет еще оставаться по одной “нераспакованной” в “иксы” единице:



А далее в каждом из полученных трех вариантов оставшаяся единица тоже дает по три варианта:

Итого действительно получается  $3 \times 3 = 9$  вариантов.

А если в исходном наборе переменных  $a$  есть и единицы, и нули? Тогда надо для каждой единицы выполнять умножение количества вариантов на 3, а для нулей такое умножение не требуется (разве что еще надо учесть, что для варианта из одних нулей будет один-единственный набор “иксов”).

Можно вывести следующее формальное правило, позволяющее вычислить окончательное количество вариантов “иксов” для того или иного набора значений переменных  $a$ :

- берем набор значений переменных  $a$  (например: (0011) );
- заменяем в нем каждую цифру 0 на 1, а каждую цифру 1 — на 3 (например: (0011) заменяется на (1133) );

• вставляем между каждыми двумя полученными цифрами знак умножения (пример:  $(1 \times 1 \times 3 \times 3)$  );

• выполняем все получившиеся перемножения (и в приведенном примере получаем 9).

Ну а для нашей задачи можно подсчитать, что:

- $(a_1 a_2 a_3) = (000)$  дает 1 набор “иксов”;
- $(a_1 a_2 a_3) = (001)$  дает 3 набора “иксов”;
- $(a_1 a_2 a_3) = (011)$  дает 9 наборов “иксов”;
- $(a_1 a_2 a_3) = (111)$  дает 27 наборов “иксов”.

А всего в сумме получается  $1 + 3 + 9 + 27 = 40$  наборов переменных  $x_1, x_2, x_3, x_4, x_5$  и  $x_6$ .

Ответ: 40.

Итак, мы разобрали все усложненные (или просто интересные) задачи, аналогичные которым встретились в тренировочной работе СтатГрад. Остались не затронутыми только задания группы С. Но они заслуживают отдельного рассмотрения каждая. И, соответственно, отдельной статьи. А пока авторы могут порекомендовать для освоения решения задач группы С воспользоваться книгами:

• Богомолова О.Б. Информатика. Полный справочник для подготовки к ЕГЭ. М.: АСТ: Астрель, 2014;

• Богомолова О.Б. Информатика. ЕГЭ за 30 дней: экспресс-репетитор. М.: АСТ: Астрель, 2014

и статей:

• Богомолова О.Б., Усенков Д.Ю. Задачи С3: “быстрый путь к выигрышу” // Информатика, 2013, № 7–8. С. 42–46.

# Задачи с исполнителем: теперь на троих

ЕГЭ

**О.Б. Богомолова,**

*д. п. н., преподаватель Московского государственного гуманитарного университета (МГУ) им. М.А. Шолохова, Москва*

**Д.Ю. Усенков,**

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

► Среди задач Единого государственного экзамена задания В1 вряд ли можно считать особенно сложными. Речь идет о задачах с исполнителем: даются две имеющиеся в нем команды и требуется на их основе составить программу, преобразующую одно заданное число в другое.

Школьники уже хорошо знакомы с такими задачами и умеют решать даже задачи “с хитростью” — те, в которых используется команда возведения в квадрат (см.: *Богомолова О.Б., Усенков Д.Ю. “Каверзная” задача про автомат // Информатика, 2013, № 7–8. С. 40–41.* Однако в последнее время стали появляться задачи типа В1 с усложнением: в них вместо двух имеется целых три команды.

Это несколько нарушает обычные рассуждения при решении подобных задач. Если имеется всего две команды, то выбор требуемых команд бывает практически очевидным. Эту очевидность дает одна из команд — деление либо вычисление квадратного корня, которые могут быть выполнены с получением целого результата не в любом случае. (А если таких команд нет, то обычно можно их получить, решая задачу вначале “с обратными командами” и заменяя умножение или возведение в квадрат на требуемые деление и вычисление квадратного корня.) В результате при решении остается лишь применять команду деления или корня, которая быстрее всего приводит нас к искомому числу-результату, всегда, когда она возможна, и брать другую команду там, где невозможна первая.

Однако если исполнитель имеет три команды, одна из которых — деление или квадратный корень, а две другие — сложение и/или вычитание, то возникает неоднозначность: если деление или корень для текущего числа неприменимы, то какую из двух оставшихся команд брать?

Можно попробовать решать задачу “в лоб”: там, где нельзя использовать деление или корень квадратный, “разветвлять” решение соответственно двум возможным вариантам и смотреть, что получится в каждом случае. Но такое решение, конечно, получится достаточно громоздким и долгим. Можно ли найти идею, “эвристику”, которая позволит выбирать на каждом шаге верную команду?

В качестве такой эвристики может быть выбрано следующее рассуждение. Если команда деления (либо корня квадратного) для текущего числа невыполнима, то мы выбираем из оставшихся команд такую, которая быстрее всего приведет наше число к числу, делящемуся нацело (либо являющемуся точным квадратом).

Как работает эта эвристика, рассмотрим на примере следующей задачи.

**Задача.** Исполнитель Вычислитель имеет три команды:

1. Прибавить 2
2. Прибавить 3
3. Умножить на 10

Требуется записать порядок команд в программе получения из числа 3 числа 626, содержащей не более 6 команд (записываются только номера команд).

**Решение**

Как видим, среди команд исполнителя нет ни одной команды, которая выполнялась бы лишь в некоторых случаях. Поэтому сначала нужно решить обратную задачу — определить программу, которая позволяет получить из числа 626 число 3, используя обратные команды:

1. Вычесть 2
2. Вычесть 3
3. Делить на 10

Вот теперь у нас есть команда, которая будет “ключевой” в выборе решения, — это деление. А одно из вычитаний мы будем выбирать так, чтобы как можно быстрее привести текущее число к делящемуся нацело на 10:

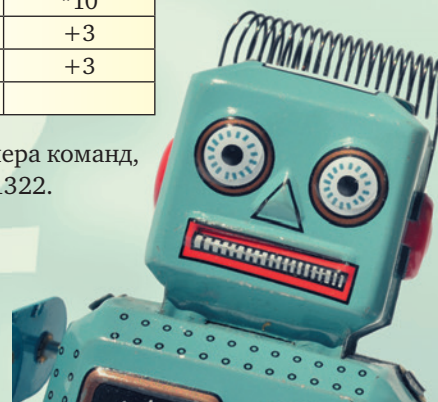
626	-3	Не делится нацело на 10
623	-3	Стремимся как можно скорее привести к числу, делящемуся на 10
620	/10	Делится на 10
62	-2	Не делится нацело на 10 Стремимся привести к числу, делящемуся на 10
60	/10	Делится на 10
6	-3	
3		

Итак, мы получили программу, которая за не более чем 6 команд приводит число 626 к числу 3. Теперь остается только лишь перейти к исходной задаче и записать “прямые” команды в обратном порядке:

3	+3
6	*10
60	+2
62	*10
620	+3
623	+3
626	

Или, если записать номера команд, получаем программу: 231322.

**Ответ:** 231322.



# Оригинальные методы перевода чисел

Д.М. Златопольский,  
Москва

► В вышедшей в 2012 году книге [1] описаны два оригинальных метода (названных “быстрыми”) перевода целых чисел из двоичной системы в десятичную и обратно.

Первый метод был предложен в 1953 году Соденом. Он заключается в следующем.

Сначала надо перевести число из двоичной системы в восьмеричную (это можно сделать “в уме”, разбив двоичное число на триады [2]).

Далее для полученного  $n$ -значного восьмеричного числа выполняются  $(n - 1)$  шагов по переводу его в десятичную систему.

Проиллюстрируем все на примере.

Пусть надо перевести в десятичную систему двоичное число 1111110000.

$$1111110000_2 = 1\ 111\ 110\ 000_2 = 1760_8.$$

Число 1760 — 4-значное ( $n = 4$ ).

На первом шаге надо из числа 1760, рассматривая его как десятичное, вычесть удвоенное произведение его первой цифры на 100 (в общем случае — на  $10^{n-2}$ ):

$$\begin{array}{r} 1760 \\ - 200 \\ \hline 1560 \end{array}$$

(при большом количестве конечных нулей их с целью экономии времени можно не писать)

На втором шаге из полученной разности (также рассматривая ее как десятичную) надо вычесть удвоенное произведение двузначного числа, образованного 1-й и 2-й цифрами разности, на 10 (на  $10^{n-3}$ ):

$$\begin{array}{r} 1560 \\ - 300 \\ \hline 1260 \end{array}$$

На третьем, в данном случае последнем, шаге вычитается удвоенное произведение трехзначного числа, образованного тремя первыми цифрами последней разности, на 1 (на  $10^0$ , или на  $10^{n-4}$ ):

$$\begin{array}{r} 1260 \\ - 252 \\ \hline 1008 \end{array}$$

Полученное число и будет искомым десятичным (проверьте!).

Видно, что при переводе используется только умножение на 2 и вычитание.

Чтобы оценить преимущества описанного способа, переведите им в десятичную систему двоичное число 11101110010101011.

Алгоритм быстрого перевода чисел из десятичной системы в двоичную, предложенный Ш.Розье в 1962 году, почти такой же (только, можно сказать, “обратный описанному”).

Сначала переводим заданное число в восьмеричную систему по методике, которую проиллюстрируем на примере десятичного числа 1945.

Надо выполнить три шага (число 1945 — 4-значное):

1-й шаг:

$$\begin{array}{r} 1945 \\ + 200 \\ \hline 2345 \end{array}$$

(здесь и ниже выполняются в восьмеричной системе)

2-й шаг:

$$\begin{array}{r} 2345 \\ + 460 \\ \hline 3025 \end{array}$$

3-й шаг:

$$\begin{array}{r} 3025 \\ + 604 \\ \hline 3631 \end{array}$$

Итак,  $1945_{10} = 3631_8$ .

После этого осталось перевести полученное восьмеричное число в двоичную систему:

$$3631_8 = 11110011001_2.$$

Предложите двум своим ученикам перевести в двоичную систему десятичное число 1234567890987654321, одному — методом Розье, другому — методом последовательного деления на основание, после чего сравните время перевода (и, конечно, результат ☺).

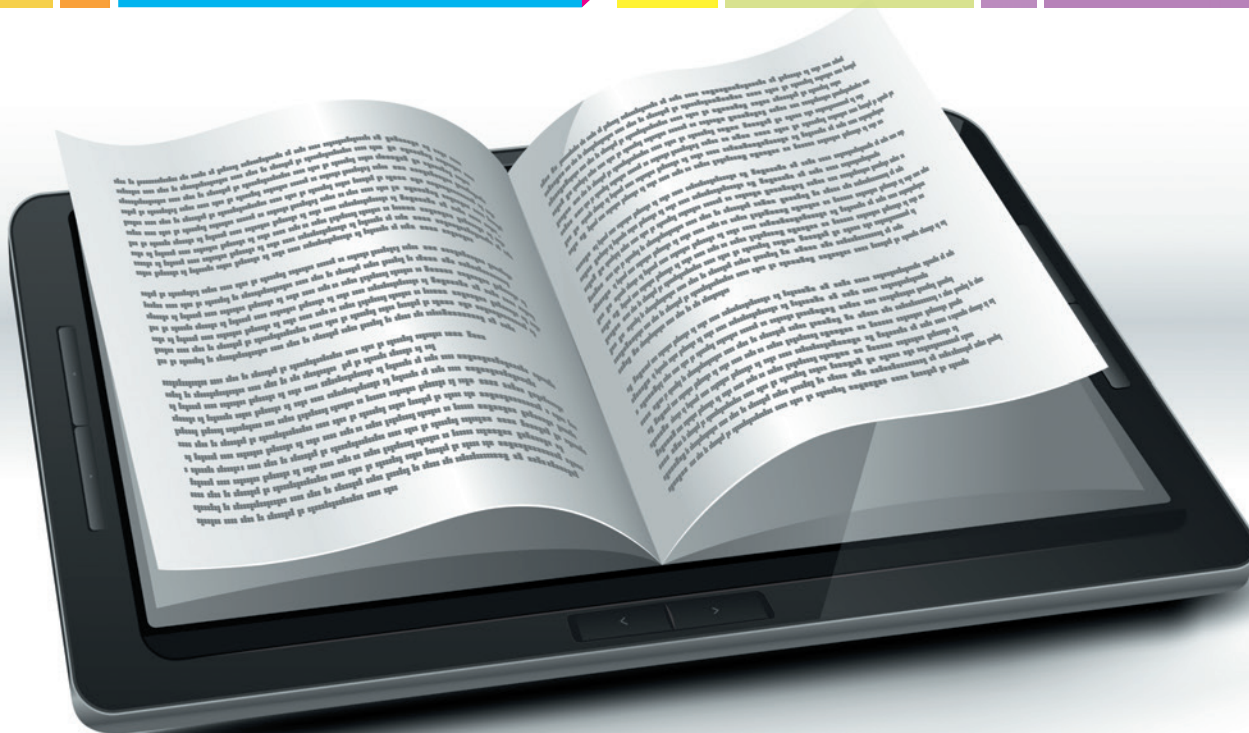
Конечно, методами Содена и Розье\* можно осуществлять и только взаимный перевод десятичных и восьмеричных чисел.

## Литература

1. Гашков С.Б. Занимательная компьютерная арифметика. Математика и искусство счета на компьютерах и без них. М.: URSS, 2012.

2. Андреева Е.В., Босова Л.Л., Фалина И.Н. Математические основы информатики. М.: Бином. Лаборатория знаний, 2007.

\* К сожалению, в книге [1] не указаны имена авторов методов и страны, в которых они проживали. Нам не удалось найти эту информацию и в Интернете.



## Как создать свою электронную книгу

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

► Когда-то, “на заре” компьютеризации, было популярным такое название — “безбумажные технологии”. Казалось, вот еще чуть-чуть, и вся информация будет передаваться, читаться и т.д. только в электронном виде — в виде файлов, с которыми можно будет работать на экранах компьютеров. Этого, правда, так и не случилось (и даже наоборот, широкое распространение компьютеров только увеличило потребление бумаги, — ведь так легко напечатать лишнюю копию “на всякий случай”, даже если она скорее всего не понадобится).

Но вот книги все увереннее становятся электронными. Конечно, вряд ли это означает полное исчезновение обычных, “бумажных” книг, — слишком много еще есть приверженцев “классики”, которым просто нравится держать книгу в руках, листать страницы, ощущать тот неповторимый запах бумаги, клея и типографской краски, который так хорошо знаком любому библиофилу... Так что “бумажные” фолианты еще долго будут

оставаться в книжных шкафах — во всяком случае, как подарочные, “элитные” издания. А вот молодежь с удовольствием “ломает традиции”, и сегодня в метро гораздо чаще можно увидеть людей, читающих текст с экранов различных электронных устройств, чем с обычными книгами в руках.

Да, электронные книги обладают целым рядом неоспоримых преимуществ.

Во-первых, компактность. На одной микро-SD карточке памяти размером с ноготь мизинца можно уместить книг больше, чем на обычной книжной полке. Это очень важно в малогабаритной квартире или общежитии и еще важнее — в поездке.

Во-вторых, доступность. Это раньше приходилось обегать все окрестные книжные и букинистические магазины в поисках книги, которую хочется прочесть, а то и собирать макулатуру и копить полученные за нее квиточки, чтобы получить долгожданную книгу за абонемент. А сегодня можно купить



почти любую электронную книгу (причем гораздо дешевле бумажной — ведь издательство в этом случае не платит за бумагу, за услуги типографии и за аренду склада, да и магазину уже не нужны обширные площади для хранения товара), их продажей занимаются как специальные сайты (например, ЛитРес), так и обычные электронные магазины (скажем, раздел электронных книг есть на ОЗОНе). Более того, электронные книги можно и совершенно бесплатно скачивать из Интернета, причем не только “пиратские”: большинство классической прозы и поэзии уже является общим достоянием. Так что, посидев часок-другой в Интернете, можно составить себе такую библиотеку, на создание которой у наших родителей уходило не один десяток лет.

В-третьих, преимуществом электронных книг является оперативность. Это, правда, радует в основном издателей: в случае обнаружения ошибки заменить прежний файл электронной книги на исправленный очень легко, так же как и пользователям совсем нетрудно скачать себе исправленную книгу взамен прежней.

В-четвертых, электронная книга, как и любой компьютерный документ, обладает гораздо большими возможностями для повышения наглядности материала. Это — не только свободное использование цвета в тексте и иллюстрациях (причем практически без затрат по сравнению с черно-белым вариантом), но и вообще возможность включения в текст практически любых медиаматериалов — анимаций, звука, видео... и даже запахов и вкуса, если когда-нибудь появится соответствующее аппаратное обеспечение ☺. Можно также включать в текст интерактивные компоненты (например, тесты в учебнике) или создавать “нелинейные” тексты при помощи гиперссылок.

Есть у электронной книги, конечно, и недостатки. Главный из них состоит в том, что электронную книгу невозможно просто взять в руки и начать читать — для этого нужен компьютер или какое-либо компьютерное устройство — специальный “ридер”, планшет, смартфон и пр., а при повреждении носителя, на котором записаны электронные книги, их легко потерять безвозвратно (если не сделаны резервные копии). Кроме того, когда нужно многократно переходить из одной части текста в другую (например, к размещенному в конце глоссарию и обратно), листать бумажные страницы и пользоваться обычными закладками многим бывает удобнее, чем переключаться с одного раздела на другой при помощи гиперссылок, а тем более — указывая номера страниц (которые приходится тогда запоминать наизусть). Авторы и издательства недолюбливают электронное книгоиздание вообще по понятной причине: с его появлением “пиратские” копии книг стало делать и распространять гораздо легче. Ну и, кроме того, если электронная книга создана путем сканирова-

ния “бумажной” с последующим автоматическим распознаванием текста, то ошибок в этом тексте обычно бывает очень много (хотя этим грешат все же обычно “пиратские” копии, в “официально” изданных электронных книгах количество ошибок не больше, чем в “бумажных”).

Тем не менее в целом можно предполагать, что в самое ближайшее время вся научно-техническая и образовательная литература, а тем более — газетно-журнальная продукция и различные “бульварные” романы — станет исключительно электронной, а “классика”, энциклопедии и словари будут выпускаться в электронном варианте гораздо большими тиражами, чем в “бумажном”. Особенно — на фоне резкого снижения цен на устройства для чтения электронных книг (“ридеры”) и широкого распространения (и удешевления!) планшетов и смартфонов, которые тоже могут использоваться в качестве “ридеров”, если на них установить специальную программу — “читалку” для электронных книг. Например, совсем недавно один из интернет-магазинов устроил распродажу “ридеров” (с цветным экраном и с дополнительными возможностями записи и воспроизведения звука) по цене всего 600 руб., а с сенсорным экраном — за 700 руб., и это притом, что даже довольно простенько изданная и не очень объемистая “бумажная” книга сегодня стоит рублей 200–300, а цена подарочных изданий доходит и до двух-трех тысяч рублей. То есть “ридер” стоит всего лишь как 3–4 обычных книги, а читать на нем можно будет десятки и даже сотни книг, которые можно получить в том числе и бесплатно. Думается, выбор большинства будет очевидным. А в системе образования и вовсе готовятся значительные перемены: начиная с 2015 года любой учебник сможет получить гриф Министерства образования и науки и будет включен в Федеральный перечень рекомендованной для школ учебной литературы, только если его “бумажная” версия будет сопровождаться электронной.

## Своя электронная книга — это реально!

Да-да! Основная цель этой статьи — не просто поговорить о том, что такое электронные книги и какие они бывают (это — только лишь необходимый “ликбез”), но и научить читателей создавать свои собственные электронные книги. Причем — полноценные, ничуть не хуже тех, которые продаются, скажем, на ЛитРес!

В принципе создать электронную книгу — совсем просто. Ведь вообще любой компьютерный документ (например, учебный реферат, подготовленный в редакторе Word) — это уже электронная книга. Но, конечно, хотелось бы, чтобы электронная книга была в одном из общепринятых, широко распространенных форматов, чтобы она красиво выглядела и чтобы ее страницы всегда отображались так, как вы это задумали (т.е. чтобы текст не

## В каких форматах бывают электронные книги

Поскольку любая электронная книга — это прежде всего компьютерный документ, она представляет собой файл (иногда — группу файлов), в котором информация записана в некотором формате. И этот формат нужно знать, чтобы правильно выбрать программное обеспечение, позволяющее читать такую электронную книгу.

Общепринятых форматов электронных книг уже существует достаточно много, и каждый из них обладает своими особенностями.

**ТХТ** — это обычный текст, с которым мы привыкли иметь дело, например, в редакторе “Блокнот”. Такая электронная книга может быть прочитана на любом устройстве, работающем с текстами (главное, чтобы кодировка кириллицы была соответствующей), но в ней не будет никакого текстового оформления и никаких иллюстраций.

**DOC** и **RTF** — это тоже привычные всем форматы текстовых документов, с которыми работает, например, текстовый редактор Word. В такой электронной книге могут быть использованы практически любые элементы оформления и иллюстрирования, какие есть в Word. Однако далеко не все устройства для чтения электронных книг смогут “понять” и корректно воспроизвести такие тексты, поэтому читать их скорее всего вам придется на настольном компьютере или ноутбуке.

**СНМ** — знакомый любому пользователю формат Windows Help. Изначально в нем создавались файлы-“подсказки” для пользователей, а теперь нередко создаются и электронные книги. Удобством его является гарантированное наличие на любом компьютере с ОС Windows программных средств для чтения, а также возможности применения шрифтовых оформлений, использования графики, создания интерактивных оглавлений и переходов по гиперссылкам и даже поиска по заданным ключевым словам. Недостаток — то, что устройства на базе Android или “классического” Linux этот формат могут и не “понять”.

**PDF** — широко используемый формат документов от фирмы Adobe. Электронная книга в этом случае представляет собой точную копию “бумажной” книги (или “имитирует” ее): состоит из отдельных страниц с текстом и иллюстрациями. Формат PDF позволяет сохранить все оформление текста и все иллю-

страции, причем отображение всего этого не зависит от используемого устройства: если на нем есть программа для чтения PDF, то можно быть уверенным, что текст будет отображен аутентично (ничто в нем не “поплывет” и не пропадет). Недостатком формата PDF является его довольно большой объем, поскольку по своей природе этот формат является графическим (хранит графические изображения страниц с текстом).

**DjVu** — еще один интересный графический формат. Обладая всеми преимуществами PDF, он реализует большую степень сжатия информации, так что объем файлов DjVu получается меньше. Основным недостатком этого формата является его нестандартность: если программа для чтения PDF есть почти у каждого (хотя бы бесплатный Acrobat Reader), то программу, “понимающую” формат DjVu, еще надо найти и установить.

**FB2** (и его “продвинутый” вариант **FB3**) — формат, набирающий сегодня все большую популярность благодаря наилучшему соотношению “объем/функциональность”. По своей сути он является текстовым (родственен формату HTML, применяемому при создании интернет-страниц, и его “продвинутой версии” — XML, на котором формат FB2, собственно, и основан), но позволяет включать в документ элементы шрифтового оформления и любые графические изображения. Кроме того, в электронных книгах в формате FB2 могут быть созданы гиперссылки, удобно реализующие интерактивные оглавления, сноски и прочие переходы по тексту, а пользователь может (при помощи соответствующих функций программы — “читалки” или устройства — “ридера”) настраивать размер шрифта и цветовую гамму, создавать электронные закладки и т.д.

Существует и ряд других форматов, созданных отдельными фирмами, например, **LIT** (созданный компанией Microsoft и во многом подобный PDF), **ePUB** (основанный на XML), **mobi** и его защищенный от пиратского копирования вариант **AZW** и др. Есть даже “3D-форматы” для создания электронных книг с возможностью “перелистывания” страниц мышкой за уголок, создаваемые на основе шаблона, написанного на базе Macromedia Flash, — такие книги обычно представляют собой исполняемый файл формата EXE. Однако основными все же можно считать форматы TXT, PDF, FB2 и DjVu, на которых реализовано наибольшее число электронных книг.

## На чем читать электронную книгу

Чтобы прочитать электронную книгу, требуется компьютер или компьютерное устройство (планшет, смартфон либо специальный “ридер”) и программа-“читалка”, понимающая формат файла, в котором записана электронная книга. Такие программы, предназначенные для чтения большого числа разных форматов, в специальном “ридере” обычно “встроенные” (и чаще всего — неизменные), а при использовании компьютера, планшета или смартфона может потребоваться скачать и доустановить нужную “читалку”.

Начнем с “ридеров”.

“Ридер” — это, по сути, тот же планшетный компьютер, только его программное обеспечение (обычно на базе Linux или Android) заранее “заточено” под выполнение одной основной задачи — хранение и чтение электронных книг. Такой “псевдокомпьютер” не позволит вам устанавливать и запускать произвольные программы (для вычислений, игр и пр.), но зато он имеет удобное меню в виде каталога имеющихся на карте памяти книг и ограниченный набор функций, который не позволит запутаться в них даже человеку, совсем не умеющему работать с “настоящим” компьютером.

Современные “ридеры” бывают с обычным экраном (тогда все управление осуществляется при помощи расположенных на корпусе кнопок) или с сенсорным экраном, подобно обычным планшетами или смартфонам (тогда все операции выполняются просто пальцами на экране). Сам же экран “ридера” может быть жидкокристаллическим (такие экраны — цветные и чаще всего сенсорные) либо выполненным на основе “электронной бумаги” (eInk) — сравнительно недавно появившейся технологии отображения информации. Главная особенность “электронной бумаги” состоит в том, что такой экран требует подачи электроэнергии только для смены изображения на нем, а уже выведенное изображение может чуть ли не месяцами сохраняться вообще без подачи энергии, как обычное типографское! Так что одной зарядки аккумулятора может хватить на 2–3 недели активного чтения. (Желающие более подробно узнать, как работает эта технология, могут посмотреть в коллекции School-Collection цифровой образовательный ресурс (ЦОР) “Электронная бумага”



из комплекта ЦОР “Информатика, 8–9-е классы” (прямая ссылка: <http://school-collection.edu.ru/catalog/res/2fd168ba-19da-42f8-a787-0e3fe87a52ed/view>.) Однако экраны на базе электронной бумаги обычно не сенсорные и чаще всего монохромные (цветная “электронная бумага” появилась лишь буквально в последние 1–2 года), а “ридеры” на базе “электронной бумаги” пока еще дороже “ридеров” с ЖК-экранами.

Особо выделить можно, пожалуй, разновидность ридеров, предназначенных для использования в школах. Эксперимент по их внедрению уже проводится в нескольких “пилотных” регионах России. Такой “ридер” имеет два экрана и благодаря этому может отображать сразу весь разворот книги (например, учебника), что в ряде случаев бывает немаловажно.

Вместо специального “ридера” можно также использовать любой планшет, смартфон или обычный настольный компьютер либо ноутбук. Но в этом случае, возможно, потребуется доустановить на него программу-“читалку”. В “джентльменский набор”, пожалуй, стоит включить:

Для компьютеров на базе Windows	Для устройств на базе Android
Adobe Acrobat или Foxit — для чтения PDF-файлов	VuDroid — программа для чтения PDF и DjVu
WinDjVu — для чтения файлов формата DjVu	
FBReader — универсальная “читалка” для форматов TXT, CHM, RTF, HTML, ePub, FB2; умеет работать с файлами в архиве ZIP	

“плыл”, как это бывает иногда при переносе Word-документа на другой компьютер). А еще — хочется, чтобы в книге было интерактивное оглавление, позволяющее переходить сразу на нужную статью.

Вот всему этому нам и предстоит научиться.

Выберем наиболее простой вариант: будем создавать электронную книгу в формате PDF. (Конвертация в другие форматы в общем-то не сложнее, если есть соответствующие программы.)

Для этого нам понадобятся:

- текстовый редактор Word — для подготовки и верстки (оформления) исходного текста электронной книги;
- программа для создания (“печати”) документа Word в PDF-файл (например, ABBYY PDF Transformer или PDF Creator);
- программа для редактирования PDF-файла — лучше всего Acrobat Professional.

Создание электронной книги рассмотрим на примере учебного пособия, посвященного различным технологиям съемки и просмотра стереоизображений (материал для него взят из бесплатного электронного журнала по 3D-технологиям — сайт <http://mir-3d-world.w.pw>). Этот журнал выходит в свет (публикуется на сайте) раз в два месяца и содержит различные статьи о стереофотографии и стереокино. Причем в наше пособие будут включены материалы по каждой из существующих стереотехнологий, заготовленные в отдельных файлах, так что нашу электронную книгу нужно будет собрать из нескольких файлов. Однако при создании электронной книги на основе материала, представленного в одном файле, общие принципы будут те же.

## 1. Создаем текст

Начинать надо конечно же с подготовки текста будущей электронной книги. Он набирается и редактируется в редакторе Word, а затем выполняется его верстка с использованием шрифтового и абзацного форматирования, вставкой иллюстраций, созданием заголовка и колонтитулов и пр. При этом, в соответствии с общепринятыми правилами, для всех частей (глав, разделов) будущей книги необходимо выдерживать единый стиль оформления текста, заголовков, колонтитулов и т.д. Дизайн текста желательно продумать заранее и отработать на какой-либо одной главе книги, а затем уже применять в других главах, воспользовавшись механизмом *стилевой разметки* либо кнопкой копирования оформления (📄).

Более подробно процесс написания и форматирования текста мы здесь описывать не будем. Желающие освоить необходимый функционал текстового редактора Word могут обратиться к учебникам и учебным пособиям, из которых можно порекомендовать следующие:

- *Фиошин М.Е., Рессин А.А., Юнусов С.М.* Информатика. 11-й класс. Углубленный уровень. Учебник (издательство “ДРОФА”) — в заново переработанном варианте данного учебника, выпускаемом

начиная с 2013 года, очень подробно описаны возможности текстового редактора Word 2010 и принципы работы с ним (для тех, у кого имеется прежняя версия Word 2003, подойдет предыдущая версия того же учебника);

- *Богомолова О.Б., Васильев А.В.* Обработка текстовой информации: практикум. М.: БИНОМ. Лаборатория знаний, 2006 — в настоящее время эту книгу можно скорее найти в библиотеке, чем в продаже; вместе с методическим пособием *Богомоловой О.В.* “Преподавание информационных технологий в школе” (М.: БИНОМ. Лаборатория знаний, 2009), которое, по сути, представляет собой учебник по информационным технологиям, данный практикум позволит освоить работу с Word 2003;

- *Богомолова О.Б.* Создание документов в Open Office.org Writer: практикум. М.: БИНОМ. Лаборатория знаний, 2011 — аналогичное пособие, посвященное основам работы в текстовом редакторе OpenOffice.org Writer (аналог Word 2003).

В результате мы должны получить набор текстовых документов, соответствующих главам будущей книги (рис. 1). Желательно выделить для них на диске отдельную папку и пронумеровать файлы, чтобы с ними удобнее было работать (определять порядок их следования друг за другом). Нумерацию страниц внутри этих документов пока можно установить начиная с единицы; позже ее нужно будет соответственно изменить.

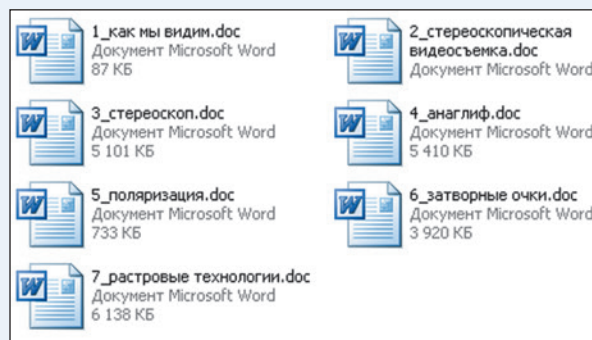


Рис. 1. Подборка файлов с главами будущей книги

## 2. Подготавливаем обложку, оглавление и выходные данные

Следующий шаг в подготовке исходных текстов — подготовка обложки и оглавления (в настоящей книге также присутствует отдельный титульный лист, страница оборота титульного листа с библиографическими данными и, обычно, отдельная страница в конце книги с так называемыми “выходными данными” — в том числе сведениями о тираже, типографии и пр.; мы же будем здесь рассматривать несколько упрощенный вариант).

Обложку, оборот титула и страницу с оглавлением можно подготовить в одном документе Word, а можно и в трех разных документах, — в зависимости от личных представлений об удобстве работы.

Желательно, чтобы обложка была достаточно красочной, привлекательной (можно воспользо-

ваться всеми имеющимися знаниями о работе в каких-либо графических редакторах). На обложке должны содержаться: фамилия и инициалы автора (или авторов) книги, ее название и год выпуска. Например, для нашей книги обложка может выглядеть так, как на *рис. 2*.



*Рис. 2.* Обложка книги

Второй лист нашей книги будет занимать информация, которая обычно дается на обороте титульного листа<sup>1</sup>:

- УДК и ББК — номера библиотечной классификации по темам изданий, от них зависит, в какие разделы различных фондов, картотек, библиотек, книжных магазинов и т.д. попадет ваша книга. Соответствующие номера можно отыскать в имеющихся каталогах (классификаторах) в Интернете. В нашем случае можно использовать номера **УДК 004.9** (раздел “Прикладные информационные (компьютерные) технологии”) и **ББК 32.973** (раздел “Компьютеры”);

- авторский знак — еще один классификационный номер, на этот раз по алфавиту, облегчающий поиск в библиотечном каталоге карточки по фамилии автора. Его также можно найти в Интернете. В данном случае фамилия автора соответствует авторский знак **У74**;

- библиографические данные — запись согласно требованиям соответствующего ГОСТ информации о книге (эта информация позже попадает в библиотечную карточку). В нашем случае библиографическая запись может быть такой:

**Усенков Д.Ю.** 3D-технологии: съемка и просмотр стереоизображений / Усенков Д.Ю. — М.: Самиздат, 2014. — <кол-во страниц> с., ил. (количество страниц нужно будет позже указать по факту, а сокращение “ил.” означает наличие иллюстраций);

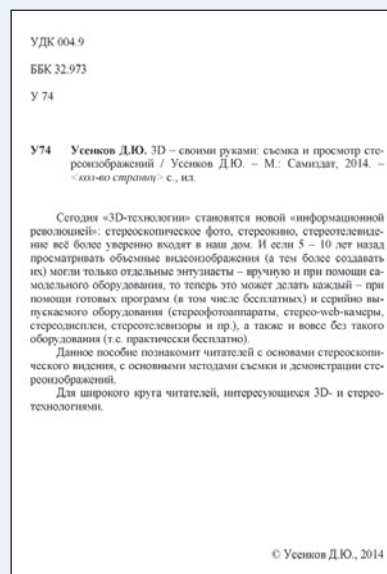
- аннотация — краткий текст, позволяющий читателю сразу определить, о чем говорится в книге;

- копирайты — сведения об авторских правах автора книги и/или выпустившего ее издательства, содержащие указание года выпуска и предваряемые знаком копирайта — ©. В нашем случае копирайт будет выглядеть так:

© Усенков Д.Ю., 2014

<sup>1</sup> Номер ISBN — универсальный номер книжного издания — обычно предоставляет издательство, приобретая его в уполномоченной организации (Национальном агентстве ISBN). Однако наличие ISBN в электронной книге пока не является обязательным, так что мы будем обходиться без него.

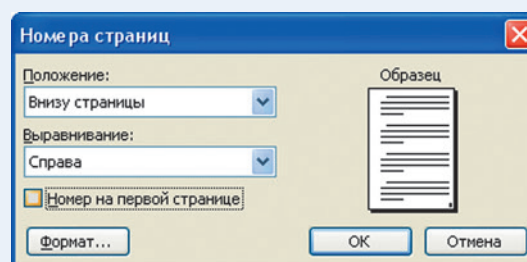
Пример страницы с указанными выше сведениями показан на *рис. 3*.



*Рис. 3.* Страница выходных сведений

Наконец, оглавление разместим на третьей по счету странице. Стиль его оформления выбирается таким же, как для остального текста книги. Под заголовком “Оглавление” перечисляются названия глав и подразделов с указанием номеров страниц. На этом этапе, когда мы определяем, сколько страниц занимает оглавление, и выполняем перенумерацию страниц во всех ранее подготовленных документах (главах), делая ее сквозной (непрерывной от начала и до конца). При этом считаем, что страница начала оглавления имеет номер 3 (на обложке и странице “оборота титула” номера не ставятся; Word позволяет отключить простановку номера на первой странице документа, а на второй странице проставленный номер можно закрыть, нарисовав поверх него прямоугольник с белой границей и белым фоном).

Напомним: для простановки номеров страниц в Word 2003 служит диалоговое окно (*рис. 4*), раскрываемое выбором пункта меню **Вставка, Номера страниц**; в нем, в частности, указывается, где именно надо проставлять номера страниц (вверху или внизу, слева, по центру или справа), а также нужно ли ставить номер на первой странице. Чтобы изменить нумерацию страниц, нужно повторно вызвать это же окно, нажать в нем кнопку **Формат** и в новом окне (*рис. 5*) в поле **начать с** указать номер, который должна иметь самая первая страница данного документа.



*Рис. 4.* Окно простановки номеров страниц в Word 2003

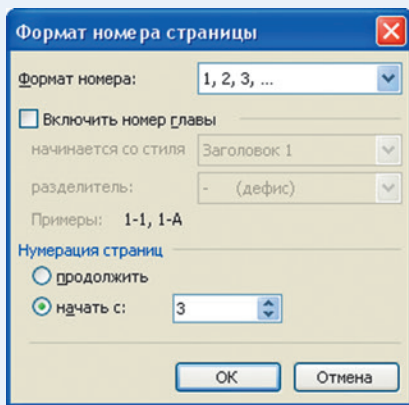


Рис. 5. Окно выбора начала нумерации страниц в Word 2003

После того как будет выполнена перенумерация страниц во всех главах, мы определим также общий объем нашей книги<sup>2</sup> и проставим его в библиографических сведениях на второй странице.

Чтобы красиво оформить оглавление, нужно воспользоваться механизмом табуляций. Для номеров страниц у правого края листа проставляется метка табуляции с выравнением справа и для нее включается отображение заполнения табуляционного промежутка точками.

Пример созданного оглавления показан на рис. 6.

Оглавление	
Как мы видим то, что видим	4
Как снимать 3D	10
Стереосъемка обычным фотоаппаратом	10
Стереосъемка через призматическую или зеркальную призму	12
Стереосъемка двумя фотоаппаратами («спаркой»)	16
3D-фотоаппараты	19
3D – в World Wide Web	21
Как смотреть 3D: со стереоскопом и без него	23
Просмотр нересрестной стереопары невооруженным глазом	24
Стереопары: «немного теории»	26
Стереоскоп	27
«Этот двухцветный объемный мир (аналогическая технология)»	35
Что такое «аналгиф»	35
Как это работает	37
Преимущества и недостатки анаглифа	39
Двухцветные очки	41
Программный инструментарий	41
На волне поляризации	53
Что такое поляризация света	53
Поляризационные технологии 3D	57
«А что в активе?» (затворный метод воспроизведения стерео)	60
Немного «физиобиологии»	60
Немного истории	62
«Активные» технологии сегодня	66
Компромиссный вариант	68
За решетками растра	69
Немного истории	70
Немного технологии	73
Немного про «много» ©	79
Немного GIFа	84

Рис. 6. Оглавление книги

Если у вас установлена программа-конвертер в формат PDF, то все очень просто: такая программа обычно при ее установке на компьютер создает еще один “виртуальный” принтер, который достаточно выбрать в стандартном окне печати. При этом в формат PDF преобразуется весь обрабатываемый Word-документ либо указанные (выводимые на печать) его отдельные страницы. Нам будет удобнее готовить и переводить в PDF каждую статью в отдельности, а затем собирать из них выпуск журнала уже в Acrobat Professional.

Итак:

1) проверяем еще раз, все ли документы подготовлены, правильно ли они отформатированы, правильно ли проставлены в них номера страниц;

2) выбираем в Word пункт меню **Файл, Печать** и в появившемся окне печати выбираем “виртуальный” принтер, установленный в системе программой-конвертером (в данном случае — **PDF-XChange for ABBYY PDF Transformer 2.0**);

3) выполняем печать страниц документа. Все опции печати при этом выставляются точно такими же, как при обычной распечатке на бумаге, но печать производится в файл. При этом будет открыто стандартное окно сохранения в файл, где можно выбрать папку для записи файлов и при желании изменить предложенное имя файла (по умолчанию оно совпадает с именем печатаемого документа Word; сохранять порядковую нумерацию файлов желательно и здесь).

Если файлов с исходными документами много, то можно воспользоваться *пакетным режимом* их обработки (если это реализовано в имеющемся PDF-конвертере). Для этого достаточно выделить все файлы формата Word, щелкнуть на них правой кнопкой мыши и выбрать в контекстном меню пункт **Convert to Adobe PDF** (или иной, аналогичный по смыслу) либо, наоборот, запустить саму программу-конвертер на исполнение и загрузить в нее требуемые документы Word.

Наконец, при отсутствии на компьютере программы-конвертера можно воспользоваться одним из онлайн-овых PDF-конвертеров, имеющих в Интернете, — например, универсальным конвертером на сайте <http://www.online-convert.com/ru?fl=ru>. Для преобразования в формат PDF нужно в окне “Конвертер документов” выбрать в списке формат конечного файла **Конвертируйте в формат PDF**, на раскрывшейся web-странице найти (чуть ниже рекламы) кнопку **Выберите файл**, нажать ее и в окне открытия файлов выбрать требуемый файл текстового документа, а затем выбрать в списке **Source language** язык **Русский** и нажать кнопку **Преобразовать файл**. Через некоторое время автоматически откроется окно сохранения файлов и вам будет предложено сохранить на диск полученный PDF-документ (скорее всего с именем **\_docx.pdf**, но его можно сразу же изменить по своему желанию).

### 3. Сохраняем текст в PDF

Теперь, когда у нас заготовлены все компоненты будущей книги в виде текстовых документов в формате Word, нужно преобразовать их в выбранный для электронной книги формат — в данном случае в формат PDF.

<sup>2</sup> Для книг, выпускаемых в “бумажном” виде, необходимо, чтобы количество страниц было кратно 8 или 16 (иначе требуется ужимать текст либо добавлять дополнительные страницы “Для заметок”); это связано с техническими особенностями типографской печати. В нашем же случае это требование не обязательно.

Таким образом, в результате мы получим подборку файлов-“заготовок” электронной книги, показанных на рис. 7.

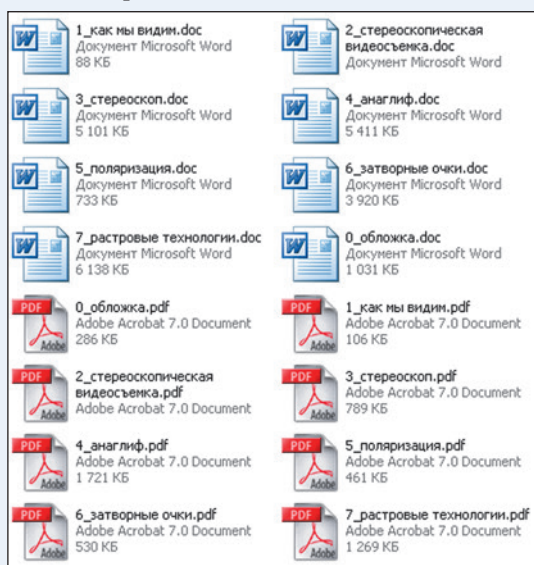


Рис. 7. Подборка файлов с главами будущей книги (в формате Word и PDF)

#### 4. Собираем электронную книгу

Следующий шаг — сборка единой электронной книги (единого PDF-документа) из заготовленных отдельных PDF-файлов. Эта операция (как и все последующие) выполняется средствами программы Adobe Acrobat Professional.

Общий алгоритм работы следующий:

1) копируем самый первый PDF-файл (в нашем случае — **0\_обложка.pdf**) и переименовываем его. Например: **Усенков Д.Ю. 3D-технологии.pdf**;

2) открываем этот файл в Adobe Acrobat Professional;

3) выбираем в меню пункт **Document, Insert Pages (Документ, Вставить страницы)**;

4) в появившемся окне открытия файлов выбираем файл PDF-документа, который нужно пристыковать к уже открытому (вот здесь-то нам и пригодится порядковая нумерация в именах файлов!);

5) в новом окне (рис. 8) нужно указать место вставки новых страниц в прежний документ. Для пристыковки очередной части книги в конце уже собранного фрагмента нужно в этом окне выбрать в списке **Location (Расположение)** пункт **After (После)**, а ниже выбрать переключатель **Last (Последняя)**;

6) затем аналогичным образом к уже собранному началу книги пристыковываются остальные PDF-документы, пока вся книга не будет собрана. Затем сохраняем полученный файл.

Результат работы — готовая электронная книга в формате PDF, которую в принципе уже можно распространять среди своих знакомых, сдавать учителю на проверку (если речь идет об учебном реферате) или даже выставить на продажу ☺. Од-

нако мы еще немного поработаем с нашим PDF-документом, чтобы повысить удобство работы с ним. В частности, сделаем оглавление интерактивным и добавим закладки.

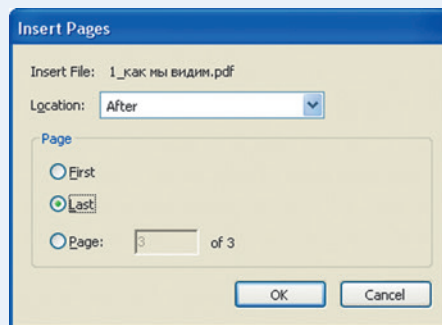



Рис. 8. Окно Adobe Acrobat Professional для добавления страниц в PDF-документ


#### 5. Создаем закладки

Одним из интерактивных средств, помогающих быстро перейти на нужную страницу электронной книги, являются закладки (bookmarks). В Adobe Acrobat Professional и Adobe Acrobat Reader (равно как и в большинстве их аналогов) закладки располагаются слева, и панель с ними можно раскрыть или спрятать щелчком по соответствующей кнопке.

Чтобы добавить новую закладку, нужно сначала прокрутить PDF-документ так, чтобы в окне программы отображалась нужная страница, а затем щелкнуть на кнопке  вверху панели закладок. (**Важно!** Последовательность действий должна быть именно такой — сначала отобразить страницу, на которую делается закладка, и только потом вставлять новую закладку, поскольку расположение закладки фиксируется сразу в момент ее создания.)

После того как закладка создана, в ней записано “условное” название **Untitled**. Оно подсвечено фоновой закраской, так что можно начать вводить вместо него текст закладки (для страницы обложки — название книги, для глав — их названия и т.д.). Если же потребуется позже изменить текст закладки, то нужно щелкнуть на ней мышью, чтобы выбрать ее (при этом закладка подсветится синим фоном, а в окне программы появится соответствующая ей страница), а затем повторно щелкнуть на ней же, чтобы войти в режим редактирования текста закладки (в целом же это не сложнее, чем, например, переименование файла в окне Windows).

Рекомендуется создать закладки по крайней мере для всей книги в целом (соответствует странице с обложкой), для оглавления и для каждой главы, но можно создать и дополнительные закладки для каждого отмеченного в оглавлении подраздела.

Ошибочно добавленную закладку можно удалить, нажав кнопку  вверху панели закладок. А чтобы реализовать в закладках “вложенную структуру” глав и подразделов, можно смещать закладки правее. Для этого нужно, нажав и удерживая

живая левую кнопку мыши, начать перетаскивать закладку чуть выше, под предыдущую, а потом попытаться смещать перетаскиваемую закладку чуть правее или левее. При этом появившаяся черная пунктирная полоска укажет, где будет после отпущения кнопки мыши располагаться верх перетаскиваемой закладки, а черный треугольничек на левом конце линии укажет положение ее левого края (рис. 9). Вложенность закладок друг в друга при этом формируется автоматически. Далее создаваемые закладки будут иметь тот же самый уровень вложенности, и при необходимости может потребоваться сместить очередную созданную закладку под последнюю из предыдущих закладок, имевших нужный уровень вложенности (рис. 10).

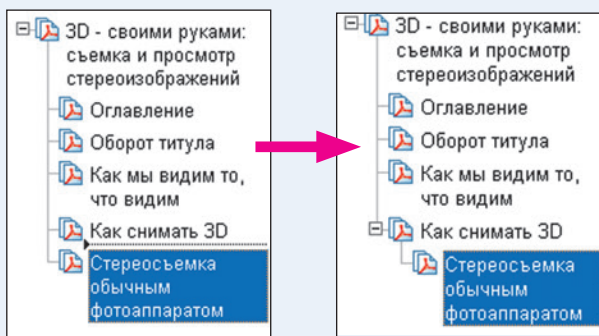


Рис. 9. Перетаскивание закладки, чтобы понизить ее уровень вложенности

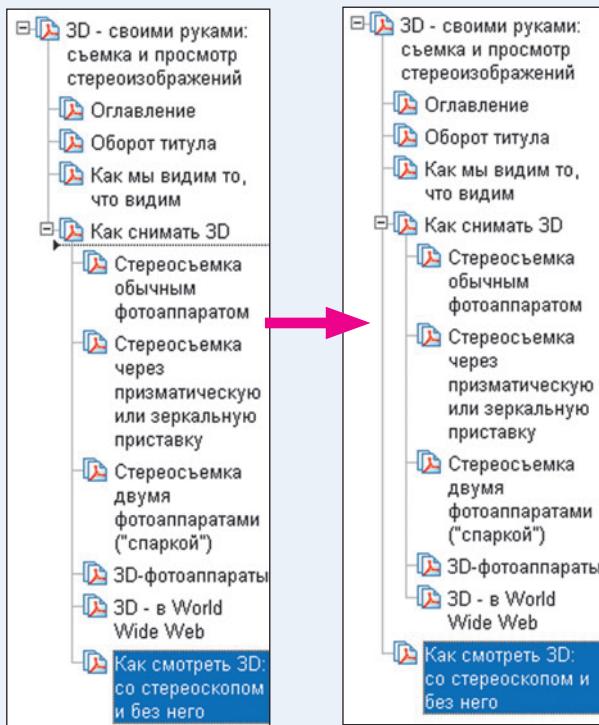


Рис. 10. Перетаскивание закладки, чтобы повысить ее уровень вложенности

После того как вся структура закладок создана, можно свернуть “ветви”, соответствующие подразделам, щелкая на мини-кнопки со знаком “[-]” (вновь развернуть их позволит щелчок мышью на появившейся взамен мини-кнопке со знаком “[+]”), — рис. 11.

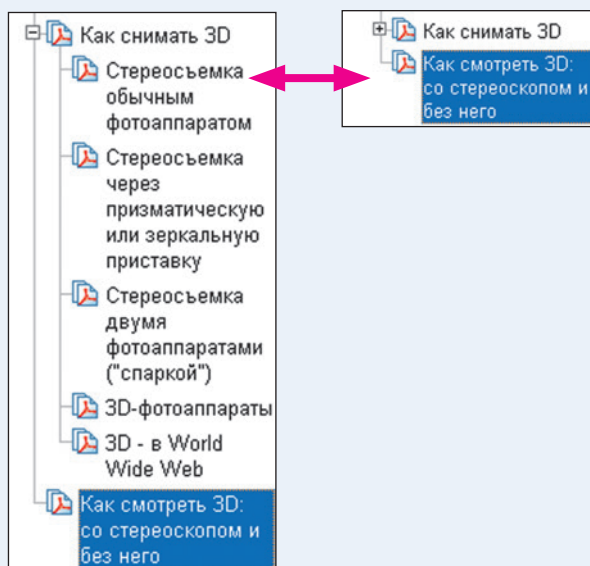


Рис. 11. Раскрытие/свертывание вложенных “ветвей” структуры закладок

При всем этом не нужно обращать внимание на некорректные переносы строк в тексте закладок, “вылезшие” на новую строку символы и пр., — к сожалению, Adobe Acrobat не позволяет форматировать тексты в закладках (и, кстати, вставлять в них нестандартные символы), а сам этот текст “плывет” при изменении ширины панели закладок.

Результат нашей работы — заполненная панель закладок, имеющая иерархическую структуру (вся книга в целом — главы — подразделы глав), где можно произвольно сворачивать/разворачивать “ветви” с вложенными закладками, а щелчок на закладке мышью вызывает переход на соответствующую страницу. Сохраняем полученные изменения в файл.

## 6. Создаем интерактивное оглавление

Чтобы сделать имеющееся оглавление интерактивным, нужно расставить в нем гиперссылки на соответствующие страницы.

- 1) Открываем страницу с оглавлением (при помощи соответствующей закладки).
- 2) Выбираем инструмент выделения текста (кнопка Select).
- 3) Выделяем очередной пункт оглавления (название вместе с многоточием и номером страницы).
- 4) Щелкаем на выделенном тексте правой кнопкой мыши и в контекстном меню выбираем пункт **Create Link (Создать ссылку)** — рис. 12.

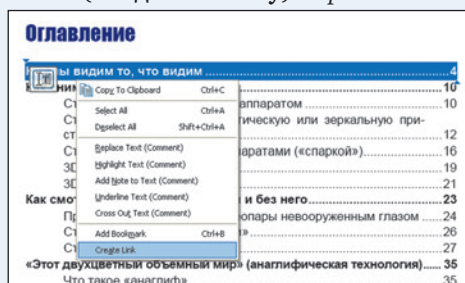


Рис. 12. Создание гиперссылки в оглавлении



5) В появившемся диалоговом окне (рис. 13) нужно установить следующие параметры оформления ссылки:

- в списке **Link Type (Тип ссылки)** выбрать пункт **Invisible Rectangle (Невидимый прямоугольник)**;
- в списке **Highlight Style (Стиль подсветки)** выбрать пункт **None (Нет)**;
- выбрать переключатель **Go to a page view (Перейти к странице)**.

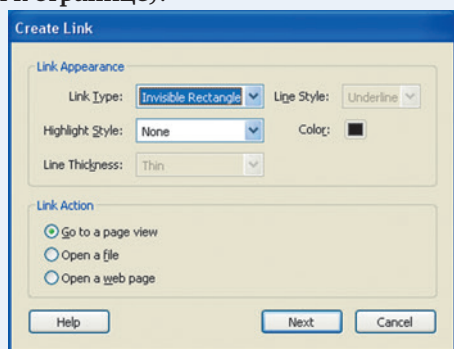



Рис. 13. Параметры оформления ссылки

6) После нажатия кнопки **Next (Далее)** на экране появится окно (рис. 14) с указанием дальнейших действий: необходимо прокрутить документ мышью (колесиком либо с помощью линейки прокрутки) или просто воспользоваться соответствующей закладкой так, чтобы в окне программы была отображена требуемая страница («цель» ссылки) именно в том виде, в каком она должна открываться при щелчке на данном пункте оглавления (рис. 14). После этого нужно в окне нажать кнопку **Set Link (Установить ссылку)**. Снова появится страница с содержанием, при этом в нем выбранный пункт меню будет обведен красным прямоугольником. Не обращайте на него внимания: это — условное обозначение, после того как вы начнете выделение следующего фрагмента текста (пункта оглавления) или выберете инструмент просмотра (кнопка )

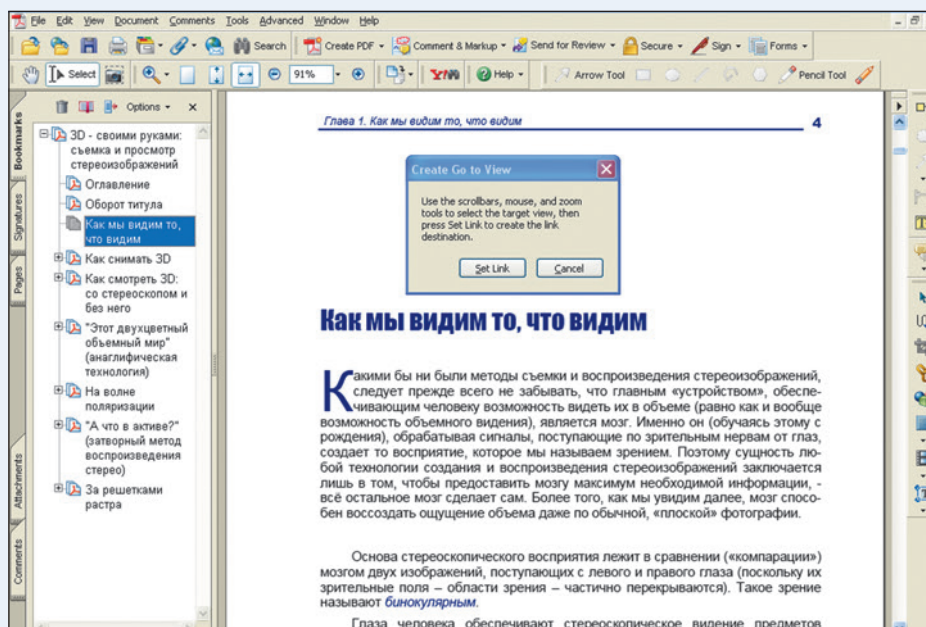




Рис. 14. Выбор целевой страницы

красная обводка исчезнет. Но ссылка (хотя и визуально не отображаемая) останется и будет действовать: нажав кнопку , можно щелкнуть мышью на только что обработанном пункте оглавления и убедиться, что откроется соответствующая страница книги.

Далее остается лишь аналогичным образом расставить все остальные ссылки на всех остальных пунктах оглавления. Закладки позволят сделать это максимально быстро. Тем более что установленные параметры оформления ссылки программа запомнила, и заново устанавливать их каждый раз уже не нужно.

Результатом работы будет интерактивное оглавление, щелкая мышью на пунктах которого (в режиме просмотра — кнопка ) можно переходить на соответствующие страницы книги. (Вернуться назад в оглавление проще всего при помощи закладки «Оглавление».)

## 7. Размечаем якоря

Еще одна полезная возможность — при размещении электронной книги на web-сайте можно из HTML-страниц переходить сразу на требуемые главы. Например, можно продублировать оглавление в HTML-формате или же обеспечить переходы на более подробное изложение материала из текста краткого реферата. Для этого используются размечаемые в тексте особые метки — **якоря**.

Те, кто умеет создавать на HTML собственные сайты, хорошо знают тег `<a>`, который используется для создания гиперссылок. В частности, этот тег может быть записан со следующими параметрами:

- `<a name="имя якоря">` — создает в этом месте HTML-документа невидимую метку с указанным именем (используются латинские буквы и цифры);

• `<a href="URL, путь и имя HTML-файла#имя якоря"> ... </a>` — создает гиперссылку не просто на указанный HTML-документ, но и на конкретное место в нем, которое было помечено якорем с указанным именем.

В PDF-документе (хотя эту информацию практически нигде не публикуют) тоже есть возможность создавать якоря, чтобы затем сослаться на них из HTML-документа при помощи ссылки типа `<a href="URL, путь и имя HTML-файла#имя якоря">`. Только в терминологии Adobe Acrobat эти якоря называются “местами назначения” (destinations).

Чтобы расставить якоря в PDF-документе, нужно выполнить следующие действия.

1) В меню программы Acrobat Professional выбрать пункт **View, Navigation Tabs, Destinations** (Просмотр, Навигационные панели, Места назначения). Откроется специальная панель для работы с “местами назначения” (рис. 15). Изначально список якорей в нем будет пустым.

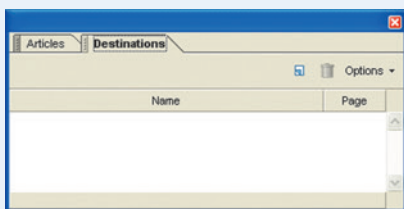


Рис. 15. Панель разметки якорей (“мест назначения”)

2) Чтобы установить якорь, нужно прокрутить PDF-документ так, чтобы была открыта нужная страница (т.е., например, если нужно поставить якорь на некоторый заголовок главы, то этот заголовок должен быть на экране в самом верху). После этого нужно в режиме выделения текста (кнопка **Select**) щелкнуть мышью где-либо на этом заголовке, поставив в него текстовый курсор.

3) Чтобы создать новый якорь, нужно в окне **Destinations** нажать кнопку . В списке якорей в

окне **Destinations** появится новый якорь с именем **Untitled**, а справа от него будет указан номер страницы с этим якорем.

Теперь можно щелкнуть мышью на этом названии якоря (два раза, как при переименовании файлов в Windows) и изменить имя якоря на любое желаемое.

Лучше всего, чтобы позже не запутаться и заодно упростить себе работу по созданию HTML-файла со ссылками на якоря, выбирать для якорей какие-то типовые имена. Например, для оглавления создадим отдельный якорь с именем **ogl**, а для глав — якоря с именами из букв **gl** и порядковых номеров глав (для подразделов отдельные якоря делать не будем). Результат показан на рис. 16.

Чтобы удалить ненужный якорь, можно воспользоваться кнопкой в окне **Destinations**. А расположенная рядом кнопка **Options** (Параметры) раскрывает меню, где имеются пункты: **New Destination** (создать новый якорь), **Sort by Name** (сортировка по именам) и **Sort by Page** (сортировка по страницам). Удобнее всего сортировать список якорей, конечно, по порядку номеров страниц (это можно сделать также и в “шапке” списка щелчком на кнопке **Page**).

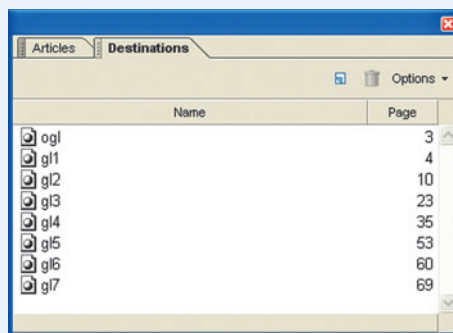


Рис. 16. Панель разметки с размеченными якорями оглавления и глав

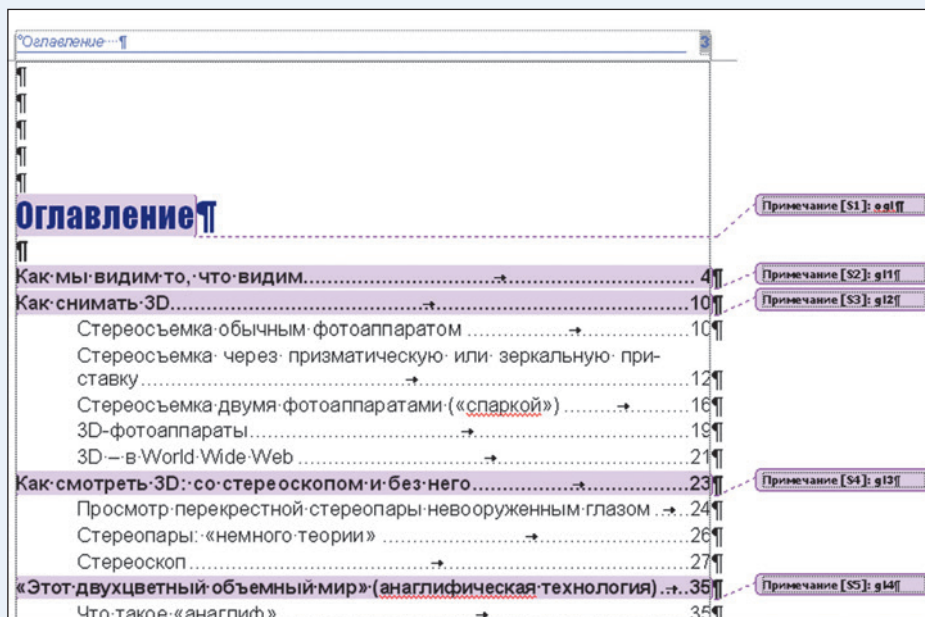


Рис. 17. Разметка в оглавлении названий якорей

Для удобства можно также составить для себя список соответствия пунктов оглавления расставленным якорям, например, в исходном файле оглавления в формате Word (рис. 17).

Вот и все. PDF-документ можно сохранить и начать создание HTML-страницы со ссылками на его якори. Например (параметр `target="_blank"` при этом указывает, что PDF-документ должен открываться в новой вкладке браузера):

```
<html>
<head>
  <title>Оглавление книги</title>
</head>
<body>
  <a href="Усенков Д.Ю._3D-технологии.pdf#ogl" target="_blank">
  <h2 align=center>Оглавление</h2></a>
  <a href="Усенков Д.Ю._3D-технологии.pdf#gl1" target="_blank">Как мы видим то,
    что видим</a><br>
  <a href="Усенков Д.Ю._3D-технологии.pdf#gl2" target="_blank">Как снимать 3D</a><br>
  <a href="Усенков Д.Ю._3D-технологии.pdf#gl3" target="_blank">Как смотреть 3D:
    со стереоскопом и без него</a><br>
  <a href="Усенков Д.Ю._3D-технологии.pdf#gl4" target="_blank">"Этот двухцветный
    объемный мир" (анаглифическая технология)</a><br>
  <a href="Усенков Д.Ю._3D-технологии.pdf#gl5" target="_blank">На волне поляризации</a><br>
  <a href="Усенков Д.Ю._3D-технологии.pdf#gl6" target="_blank">"А что в активе?"
    (затворный метод воспроизведения стерео)</a><br>
  <a href="Усенков Д.Ю._3D-технологии.pdf#gl7" target="_blank">За решетками растра</a>
</body>
</html>
```

Как выглядит такое HTML-оглавление, видно на рис. 18.

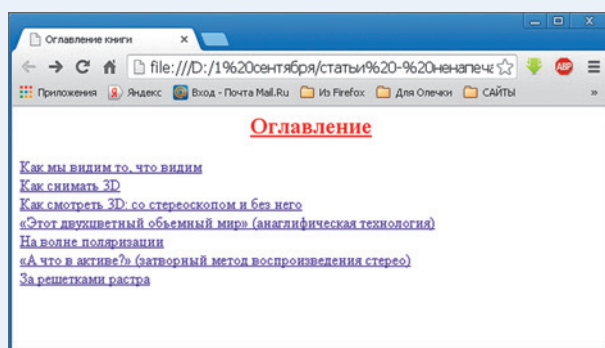


Рис. 18. HTML-оглавление со ссылками на якори в PDF-документе

## 8. А что, если ошибка?

А что делать, если вдруг уже после того, как электронная книга полностью подготовлена и размечена, вы в тексте заметите ошибку?

Вот здесь и проявляется то самое полезное свойство электронной книги по сравнению с «бумажной». Достаточно просто исправить ошибку в исходном тексте в формате Word, потом распечатать только исправленную страницу (страницы) в формат PDF, а затем заменить исправленными страницами соответствующую часть электронной книги. Причем, что важно, — поскольку все сделанные интерактивные разметки (закладки, ссылки, якори) в Adobe Acrobat связываются не с самим текстом в документе PDF, а только лишь с номерами соответствующих страниц, переделывать после такой замены страниц ничего не нужно!

Чтобы заменить в уже готовом PDF-документе электронной книги одну или несколько страниц, нужно:

- 1) выбрать в меню Adobe Acrobat Professional пункт **Document, Replace Pages (Документ, Замена страниц)**;

- 2) в появившемся окне открытия файлов выбрать файл PDF с исправленной страницей (страницами);

- 3) в открывшемся окне замены страниц (рис. 19) нужно в верхних полях (**Replace Pages** и **To**) указать номера заменяемых страниц (с какой по какую). После этого в нижней части окна надо проконтролировать автоматически выведенное указание — сколько страниц из нового документа будет использовано для замены, а также указать в поле **With Pages** номер страницы в новом документе, начиная с которой нужно брать из него заменяющие страницы.

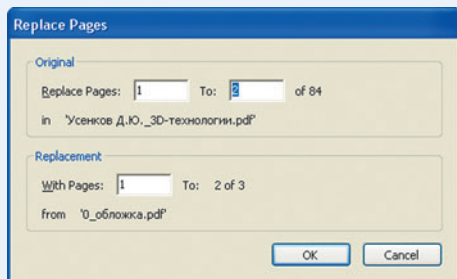


Рис. 19. Окно замены страниц в PDF-документе

После выполнения замены страниц остается лишь заново просмотреть электронную книгу и проверить правильность замены.

## 9. ЗАЩИТИМ СВОЮ ИНТЕЛЛЕКТУАЛЬНУЮ СОБСТВЕННОСТЬ

Когда электронная книга создана, может быть, вам захочется защитить ее от возможности копирования фрагментов текста из книги, печати на принтере (если вы хотите разрешить только чтение с экрана) и пр. Adobe Acrobat Professional позволяет установить пароль на PDF-документ и заблокировать те или иные операции с ним.

Чтобы установить пароль, нужно:

1) выбрать в меню пункт **File, Document Properties (Файл, Свойства документа)** и в открывшемся окне выбрать вкладку **Security (Защита)** — рис. 20;

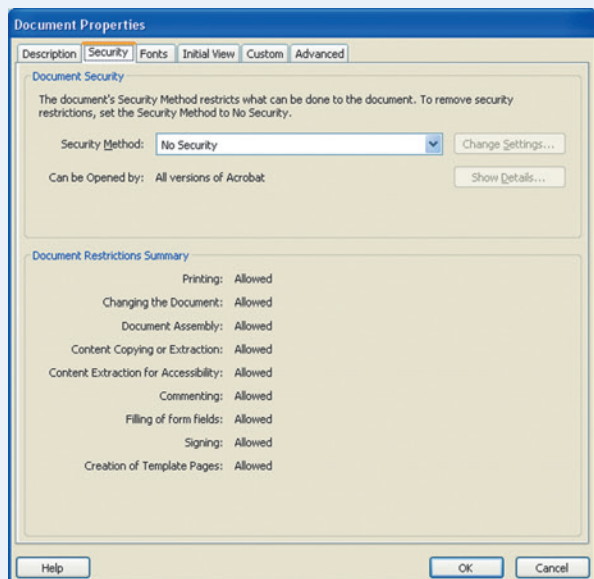


Рис. 20. Окно защиты PDF-документа паролем

2) изначально в списке **Security Method (Способ защиты)** установлен пункт **No Security (Нет защиты)**. Вместо него нужно выбрать пункт **Password Security (Защита паролем)** — откроется новое окно (рис. 21) для настройки пароля.

В этом окне можно сделать следующие установки:

- в списке **Compatibility (Совместимость)** оставляем предложенный по умолчанию пункт **Acrobat 5.0 and later** (версия 5.0 и старше);

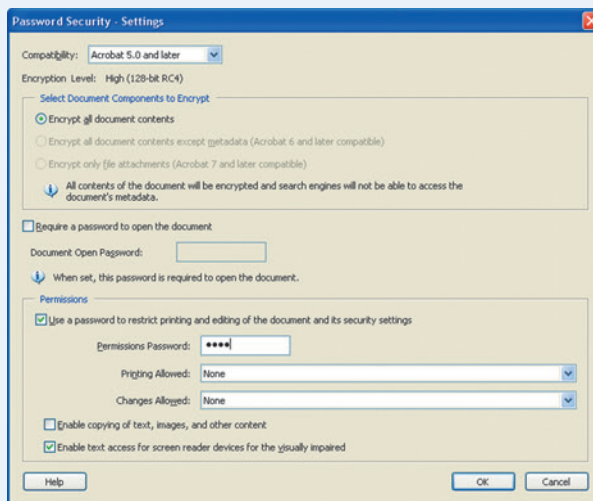


Рис. 21. Окно настройки защиты паролем

- оставляем помеченным переключатель **Encrypt all document contents (Шифрование всего содержимого документа)**;

- если вы хотите предоставлять возможность открывать и читать данную электронную книгу только тем, кому вы сообщите пароль, то можно пометить флажок **Require a password to open the document (Требовать пароль для открытия документа)** и ввести желаемый пароль в ставшем доступным поле **Document Open Password**;

- чтобы оставить возможность просмотра электронной книги, но выборочно заблокировать те или иные возможности работы с ней, нужно оставить флажок **Require a password to open the document** не помеченным, но пометить расположенный ниже флажок **Use a password to restrict printing and editing of the document and its security settings (Использовать пароль для запрета печати и редактирования документа и его установки защиты)**. После этого нужно:

- ✓ ввести в поле **Permissions Password** желаемый пароль;

- ✓ в списке **Printing Allowed (Разрешить печать)** можно выбрать один из пунктов: **None (Нет)** — полное запрещение распечатки на принтере, **Low Resolution (150 dpi) (Низкое разрешение)** — разрешить “черновую” печать или **High Resolution (Высокое разрешение)** — разрешить распечатку в высоком качестве;

- ✓ в списке **Changes Allowed (Разрешить изменения)** можно выбрать один из пунктов: **None (Нет)** — запрет любых изменений, **Inserting, deleting, and rotating pages (Вставка, удаление и поворот страниц)** — разрешение вставлять, удалять и поворачивать страницы, **Any except extracting pages (Все, кроме извлечения страниц)** — разрешение любых действий, кроме возможности выделять выбранные страницы PDF-документа как отдельные PDF-документы (еще два расположенных здесь пункта касаются разрешений/запретов при работе с интерактив-

ными формами в HTML-документе и нам не интересны);

▼ флажок **Enable copying of text, images, and other content** (**Разрешить копирование текста, изображений и другого содержимого документа**) нужно оставить не помеченным, если вы не хотите допускать копирование текста и рисунков из вашей электронной книги;

▼ флажок **Enable text access for screen reader devices for the visually impaired** (**Разрешить доступ к тексту для устройств экранного чтения для слабовидящих**), наоборот, лучше оставить помеченным;

• остается нажать кнопку **OK** и в появившемся окне (либо, возможно, нескольких окнах) подтвердить выполнение операции (в том числе повторно ввести пароль для контроля).

После этого окно настройки параметров защиты паролем закроется и на экране снова появится окно свойств документа (вкладка **Security**). Его можно сразу закрыть, после чего сохранить защищенный PDF-документ (желательно под другим именем, чтобы в случае, если вы забудете пароль, у вас остался исходный не запароленный документ).

Если теперь закрыть, а потом снова открыть файл защищенного PDF-документа, то можно увидеть, что в меню программы Adobe Acrobat Professional все пункты, соответствующие запрещенным операциям (печать на принтер, копирование, вставка/замена страниц и т.д.), стали серыми, недоступными для выбора.

Если же теперь вы захотите снять защиту с PDF-документа или изменить ее настройки, то нужно снова выбрать пункт меню **File, Document Properties** и на вкладке **Security** открывшегося окна свойств документа нажать кнопку **Change Settings** (**Изменить настройки**) справа от списка **Security Method** (рис. 22).

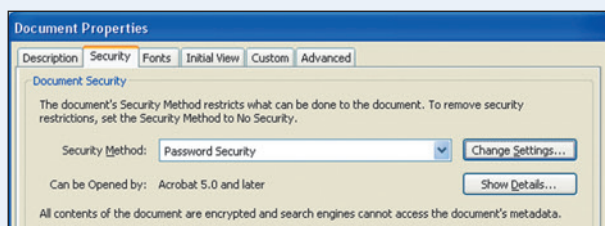


Рис. 22. Изменение настроек защиты паролем

Прежде всего программа в отдельном окне запросит у вас пароль — тот самый, который вы ранее установили. А после этого на экране вновь откроется окно настройки защиты паролем (см. рис. 21) со всеми ранее сделанными установками. Здесь можно поменять как эти установки, так и пароль и затем сохранить PDF-документ заново.

## 10. А теперь “сделаем красиво”

Пока что наша электронная книга — это только лишь обычный PDF-файл. Кроме того, если она

будет выставлена для всеобщего обозрения на сайте, то скачать ее на свой компьютер сможет любой желающий (просто так “устроен” доступ к любым материалам в World Wide Web). Однако можно (как мы отмечали в начале этой статьи) при помощи специальных программ преобразовать созданный PDF-документ в электронную книгу “формата 3D” — с красивым листанием страниц. Кроме того, такая электронная книга будет уже сохранена в формате Macromedia Flash, так что при ее размещении на web-странице будет обеспечено чтение книги, но не ее копирование (во всяком случае, не стандартными способами).

Из всех существующих программ конвертации PDF в “3D-книги” можно порекомендовать программу FlippingBook PDF Publisher.

После запуска программы открывается ее рабочее окно, пока пустое (рис. 23). Эта программа имеет достаточно большое число настроек (в том числе размеры страниц будущей электронной книги, используемая цветовая гамма, наборы кнопок и т.д., а также текст заголовка). Найти и правильно установить все нужные настройки в длинном их списке может быть довольно сложно (тем более на английском языке), но можно сохранить уже сделанные установки в файле и позже подгружать его. (Такой файл настроек прилагается в составе электронного приложения; потребуется только указать для параметра **Main Window Settings, Title Label** название электронной книги, которое должно отображаться в ее заголовке.)

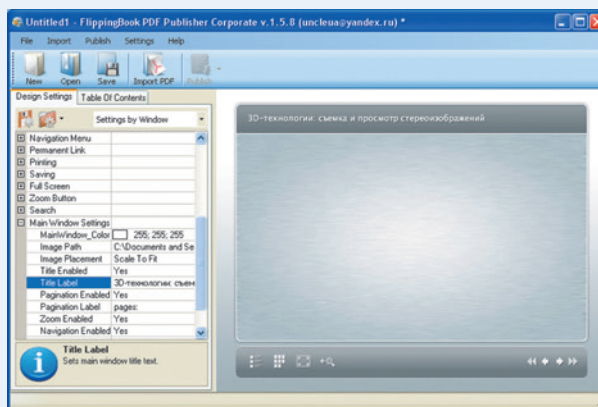


Рис. 23. Окно программы-конвертера в “3D”-электронную книгу

Следующий шаг — загрузка (“импорт”) в программу обрабатываемого PDF-файла (не запароленного). Для этого нужно выбрать в меню пункт **Import, Import PDF** либо нажать одноименную кнопку.

После выбора в окне открытия файлов желаемого PDF-документа открывается окно настройки (рис. 24). Здесь, в частности, указывается:

- переключатели группы **Import Range** — диапазон страниц, извлекаемых из PDF-документа: **All** (все) или **Custom range** (заданный интервал);
- группа **Conversion** — настройки формата, в который выполняется конверсия (здесь оставля-

ем выбор пункта **SWF** и опции, заданные по умолчанию);

- флажок **Import links** — его пометка предписывает сохранять все имеющиеся в PDF-документе гиперссылки;
- флажок **Import Table Of Contents** — его пометка предписывает загрузить из PDF-документа его содержание.

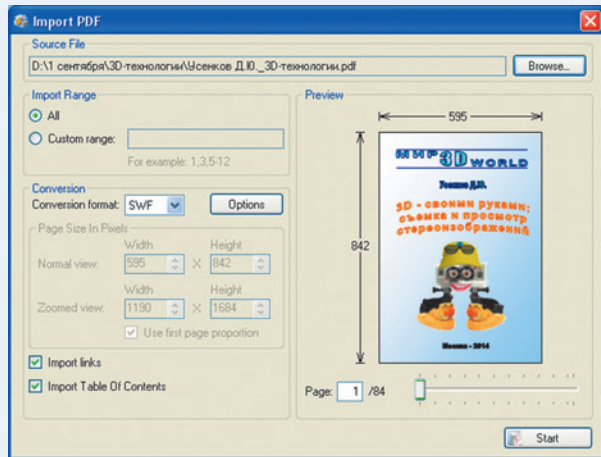








Рис. 24. Окно настройки импорта PDF-документа

Нажав кнопку **Start**, запускаем процесс импорта PDF-документа в программу-конвертер и ждем его завершения (процесс индицируется динамическим индикатором).

Как только конверсия будет завершена, окно импорта закроется, а в окне программы можно будет увидеть (в миниатюре) готовую “3D”-книгу (рис. 25). Можно полистать ее мышью, попробовать пользоваться всеми имеющимися кнопками. Например, можно раскрыть оглавление (кнопка ) и увидеть, что в открывшейся панели имеется то оглавление из закладок, которое мы ранее сделали в PDF-файле (только для раскрытия/закрытия вложенных “ветвей” надо щелкать мышью исключительно на значках в виде треугольников, а не на самих пунктах оглавления). Кнопка  позволяет работать с миниатюрами страниц, чтобы сразу перейти на нужную. Кнопка  раскрывает электронную книгу на весь экран (правда, эта функция будет работать только уже после размещения электронной книги на web-сайте). А кнопка  позволит увеличить изображение текущей страницы (или разворота) электронной книги (тогда повторный щелчок на этой кнопке, снабженной уже знаком минуса, восстанавливает исходный размер отображения), — то же можно делать, щелкая мышью прямо на изображении страниц электронной книги. Кнопки   внизу справа позволят листать страницы книги (то же можно сделать при помощи кнопок слева и справа от самой книги или перетаскивая мышью нижние уголки ее страниц). Вверху, там, где указаны через косую черту номер текущей страницы и общее число страниц, можно щелкнуть мышью на номере текущей страницы (он отображен белым цветом), — появится поле, в ко-

тором можно ввести номер нужной страницы и нажатием клавиши **Enter** перейти на нее. И, наконец, поле вверху справа позволяет осуществлять поиск по заданным словам — если в исходном PDF при его создании был сохранен “текстовый слой” — исходные тексты в виде самих символов, а не только их изображений.

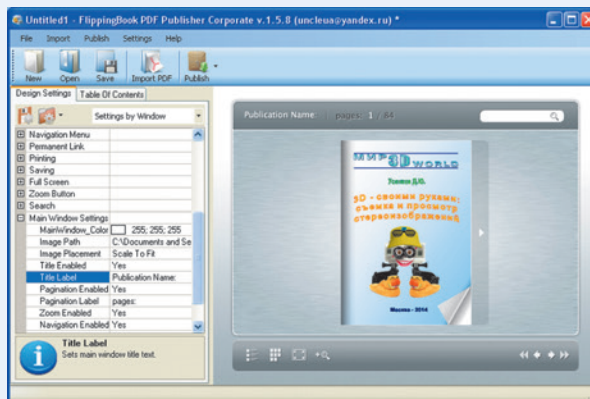


Рис. 25. “3D”-книга создана!

Остается сохранить созданную “3D”-книгу. Для этого нужно выбрать в меню пункт **Publish, To HTML** (или нажать кнопку **Publish** и выбрать пункт **To HTML** в открывшемся меню).

В появившемся окне (рис. 26) можно указать желаемую папку диска, в которую будет записан весь комплект файлов, соответствующих электронной книге. В поле справа вводится текст, который будет отображаться в заголовке браузера для страницы с опубликованной электронной книгой. А нажатие кнопки **Publish Project** запускает сохранение созданной книги на диск.

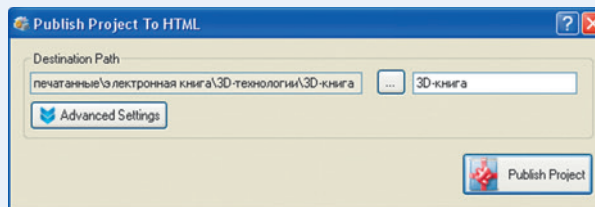


Рис. 26. Сохранение созданной “3D”-книги на диск

По завершении операции сохранения кнопка **Publish Project** заменяется на кнопку **View Result**, нажав которую можно загрузить страницу с электронной книгой для просмотра в браузер (рис. 27).

Интересно, кстати, посмотреть, как выглядит полученный комплект файлов. Открыв заданную при сохранении папку, можно увидеть, что в ней имеются две вложенные папки и два файла (рис. 28):

- файл **index.html** — это web-страница с электронной книгой, которую нужно открывать для чтения книги;
- файл **movie.swf** — “движок” “3D”-книги на Macromedia Flash;
- папка **files** — вспомогательные файлы (в частности, извлеченные из исходного PDF и сохраненные для каждой страницы в формате XML “ключевые слова” для реализации поиска по электронной книге);

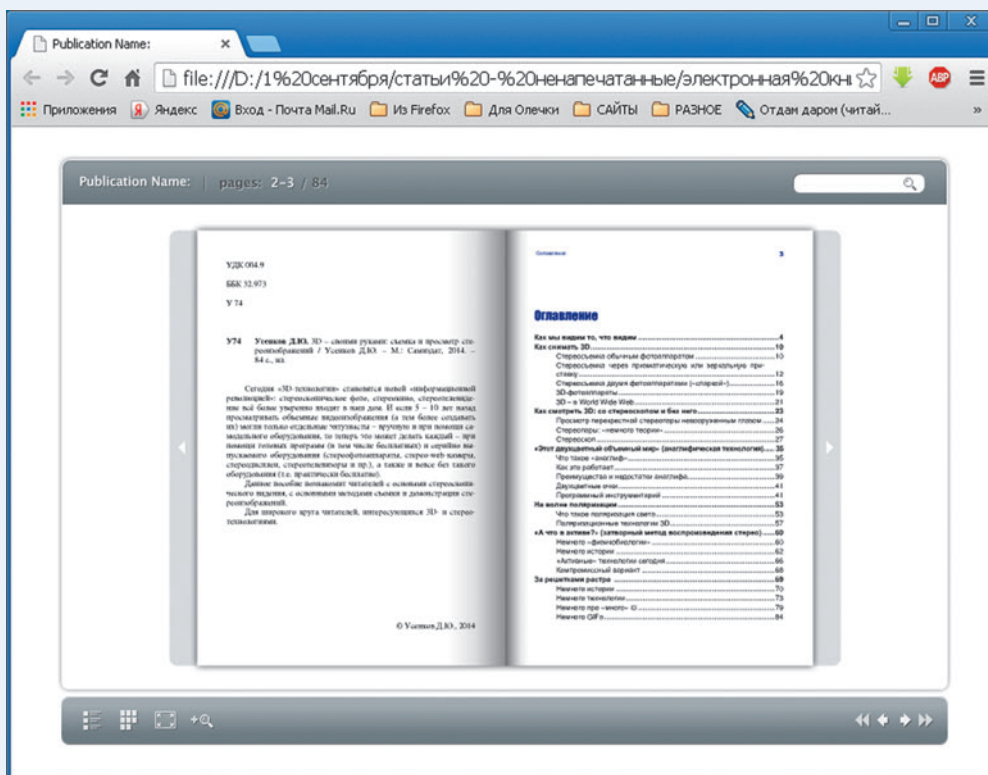


Рис. 27. “3D”-книга в окне браузера

• наконец, папка **images** — это то, во что при конвертации превратился наш PDF-файл. Открыв ее, можно увидеть, что это — изображения страниц электронной книги (каждая страница — в отдельном файле) в формате Macromedia Flash и их миниатюр в формате **jpg**.

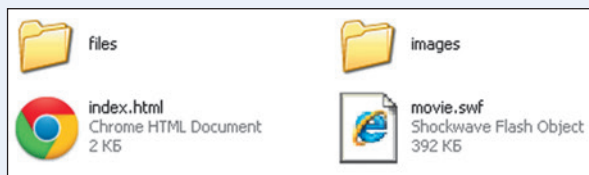


Рис. 28. Файлы “3D”-книги

Теперь остается лишь выложить весь этот комплект файлов на своем сайте и предусмотреть ссылку с него на данный файл **index.html** для открытия книги. И все! При этом очевидно, что, не зная точного пути к папке **images**, посетители web-страницы, читающие книгу, не смогут ее скачать на свой компьютер как целое. Причем эти данные прошиты в swf-файле, а не в HTML-коде, так что при всем желании извлечь их оттуда будет довольно непросто.

Вот мы и завершили создание нашей первой электронной книги. А далее — вся инициатива в ваших руках, господа читатели! И можно надеяться, что теперь вы сможете внести свой вклад в электронное книгоиздание.



## Еремин Евгений Александрович

Доцент кафедры мультимедийной дидактики и ИТО Пермского государственного гуманитарно-педагогического университета, кандидат физико-математических наук, Почетный работник высшего профессионального образования Российской Федерации. Работает с вычислительной техникой начиная с 1974 года, преподает связанные с информатикой курсы с момента появления этого предмета в школе в 1985 году. Общий педагогический стаж — более 30 лет. Всего имеет более 150 публикаций, в том числе несколько книг. Сотрудничает с редакцией нашего издания с 2001 года, за это время опубликовал в “Информатике” не менее 50 статей по различным вопросам: от основ вычислительной техники, языка VRML и среды Scratch до физических основ цвета и роли наглядности; а многие читатели, вероятно, вспомнят его с А.П. Шестаковым “примерные ответы на примерные билеты”. Область интересов: архитектура вычислительной техники, программирование, методика их преподавания; представление знаний.

*От редакции.  
Мы продолжаем публиковать интервью с нашими коллегами. Теми, кто здесь и сейчас работает в школьной информатике*

*Как Вы пришли в информатику? Первый компьютер, на котором Вы работали? Первый язык программирования, с которым Вы познакомились? Первые задачи, которые Вы решали?*

Пришел очень естественно. В аспирантуре занимался численным решением уравнений, описывающих поведение жидкости с химической реакцией, — значит, нужно было проводить сложные расчеты. А когда в 1985 году в школе ввели новый предмет, наша кафедра теоретической физики оказалась к ЭВМ “ближайшей по духу”. И кому, как не молодым (в то время) сотрудникам было заняться этим новым предметом? С тех пор и пошло...

Свои самые первые вычисления проводил на машине второго поколения “Наири-2”. Ее и компьютером-то нельзя назвать, тогда в ходу был термин “ЭВМ”; машина занимала целую комнату. У нее был специфический язык под названием АП (АвтоПрограммирование), его очень забавно сейчас вспоминать! Транслятор был не самый эффективный, задачи считались медленно. Поэтому почти сразу научился переписывать уравнения в машинные коды — это ускоряло расчеты в 1,5–2 раза. С тех пор полюбил низкоуровневое программирование.

Еще есть очень много воспоминаний о внедрении предмета и обучении учителей, но это отдельный разговор.

*Первые пять фамилий из мира информатики, которые приходят Вам в голову? Если можете, пожалуйста, коротко расскажите, почему именно они.*

Аль-Хорезми, который “придумал” алгоритмы, Чарльз Бэббидж, который предложил автоматическую обработку информации, Ада Лавлейс — первая программистка, Джобс и Возняк, которые сделали ПК. Не могу также не вспомнить А.П. Ершова, благодаря которому информатика пришла в школу (в развитие науки информатики он тоже внес вклад).



*Какая компьютерная техника и IT-сервисы окружают Вас сейчас? Насколько Вы “компьютерный” человек в обычной жизни? Это вопрос про все — про компьютеры и гаджеты, которые Вас окружают, про Ваш личный стиль социальной коммуникации в сети — насколько Вы живете в социальных сетях, насколько не можете жить без электронной почты и других средств коммуникации. Насколько бережете или не бережете свое личное пространство и время.*

Я не слишком “компьютерный” человек в быту. ПК у меня дома очень давно. Есть ноутбук, сравнительно недавно купил для поездок нетбук. Сотовый телефон довольно простой, Опера-мини на нем установлена, и 1–2 раза в месяц я ее даже запускаю (после ПК противно — экран маленький, кнопки убогие...). Модных гаджетов не держу. Недавно после вечера встречи с одноклассниками (35 лет выпуска) хотел завести аккаунт в “Одноклассниках”, но требование номера моего сотового телефона со стороны этого сервиса вызвало у меня внутренний протест (как вы понимаете, они его не дождались). С другой стороны, я до сих пор (с 1999 года) пользуюсь аккаунтом на *yahoo.com*, хотя в последнее время там много изменений в худшую сторону. Электронной почтой, конечно, пользуюсь — как без этого? Социальные сети не привлекают, зато “перебрал” множество бесплатных хостингов для своих учебных материалов.



*Если бы в Вашем распоряжении была лишь одна лекция (допустим — “пара”, полтора часа) и полная свобода рассказывать о том, что Вам интересно в информатике, чему бы Вы посвятили это время?*

Это простой для меня вопрос. Рассказал бы про устройство компьютера и про то, как он обрабатывает информацию.

*Есть ли какой-то факт из информатики — может быть, какой-то алгоритм, — который произвел на Вас сильное, запоминающееся впечатление?*

После некоторых раздумий вспомнил, что, пожалуй, очень глубокое впечатление на меня произвело первое знакомство с Delphi (тогда еще версии 1). После Турбо-Паскаля это был прорыв!

*Есть ли у Вас любимая задача или задачи?*

Не уверен, что я отвечаю на вопрос, но некоторое время назад я очень любил “ковыряться” во внутренностях ПО для “Ямахи”. В те годы это была очень передовая машина! Возможно, это и не совсем правильно, но я до сих пор с большой теплотой и даже некоторой гордостью вспоминаю свою статью всего на полстранички «Улучшение Турбо-Паскаля для “Ямахи»» (журнал “Информатика и образование”, 1990, № 6). Там показано, как после изменения трех байт внутри компилятора он начинает нормально выходить из программы ученика по нажатию  +  (в имевшемся варианте система напрочь зависала!).

*Какую тему/темы школьного курса (будем считать, что мы находимся в пространстве ФГОС) Вы не хотели бы рассказывать детям? Ну не нравится она Вам.*

Все, что связано с философским осмыслением информации. Я вообще очень не люблю, когда в ответ на несколько “трескучих” фраз в программе трудно найти что-либо конкретное в учебной литературе: как хочешь, так и рассказывай...

*Надо ли сейчас вообще учить предмету “информатика” в школе? Ведь навыки работы на компьютерах, в сетях, с прикладным ПО дети получают и без этого и гораздо раньше, чем начинают изучать предмет. А специалистов вполне можно готовить в вузах. В школе-то зачем?*

Ответ сильно зависит от того, что понимать под содержанием предмета. Если работа в браузере — то незачем, конечно. А если серьезные вещи, адаптированные под уровень школьников, — это совсем другое дело.

Чтобы пояснить, что я имею в виду, приведу две цитаты.

Первая. “Даже если не делается попытки показать картину того, что происходит “внутри” [компьютера], учащиеся создадут свою собственную” (*du Boulay B. Some difficulties of learning to program*).

Вторая. В замечательной обзорной статье М.Бен-Ари “Constructivism in Computer Science Education” убедительно показывается, что формирование у учеников представлений о работе компьютера в виде некоторой модели является обязательным условием успешности любой нетривиальной компьютерной деятельности. Причем добиваться формирования модели необходимо в начале курса до освоения каких-либо более сложных видов деятельности, в том числе пользовательских (М.Бен-Ари в своей статье неоднократно рассматривает примеры работы с текстовым процессором). По сравнению с естественными науками, например с физикой, где модели действительности формируются интуитивно из жизненного опыта, в компьютерных дисциплинах это не так. Ссылаясь на результаты экспериментальных исследований, М.Бен-Ари утверждает, что “интуитивные модели компьютера обречены быть нежизнеспособными. В лучшем случае такие модели ограничены образом грубого гигантского антропоморфного мозга, что едва ли является полезным при изучении computer science”.

*Какой Вы представляете себе информатику в школе через пять лет? А через десять лет?*

Очень трудно ответить. Я даже плохо представляю, что вообще сделают с нашим образованием за пять лет, если реформы будут продолжаться так, как они сейчас делаются! А какой бы я хотел видеть информатику? Глубокой, но, опять же повторю, не философского уровня. А о примерном направлении я уже написал в ответе на предыдущий вопрос. Возможно, к этому добавится материал, который сейчас условно называют “инженерией знаний”.

*Какой вопрос Вы хотели бы задать себе, чтобы на него ответить?*

Это, видимо, надо что-то придумать вроде того, как поется в известной песне Максима Леонидова: “Я оглянулся посмотреть, не оглянулась ли она, чтоб посмотреть, не оглянулся ли я”? Затрудняюсь, очень затрудняюсь... Давайте я спрошу себя: что вы желаете школьной информатике? И тут же сам отвечу: сохраниться в школе в виде отдельного предмета и определиться, наконец, с содержанием материала. Мое личное мнение — быть поближе к мировой computer science и не искать на уровне школьной информатики особого российского пути.



## ДИСТАНЦИОННЫЕ КУРСЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ

(с учетом требований ФГОС)

Открыт прием заявок на 2014/15 учебный год

### образовательные программы:

- НОРМАТИВНЫЙ СРОК ОСВОЕНИЯ – **108** УЧЕБНЫХ ЧАСОВ  
Стоимость – 3990 руб.

- НОРМАТИВНЫЙ СРОК ОСВОЕНИЯ – **72** УЧЕБНЫХ ЧАСА  
Стоимость – 3390 руб.

По окончании выдается удостоверение о повышении квалификации  
установленного образца

Перечень курсов и подробности – на сайте [edu.1september.ru](http://edu.1september.ru)

Пожалуйста, обратите внимание:

заявки на обучение подаются только из Личного кабинета,  
который можно открыть на любом сайте портала [www.1september.ru](http://www.1september.ru)



## Вопросы по информатике для проведения школьных конкурсов “Что? Где? Когда?” и “Брейн-ринг”

Д.М. Златопольский,  
Москва

1. Когда при делении одного на два получается два?

**Вариант вопроса**

Когда справедливо равенство:

$$1 : 2 = 2$$

2. Ведущий: “Вы знаете, что такое «русское лото»?”. Напомню, что это игра, в которой поочередно кто-то из играющих вынимает из мешочка небольшие бочонки, на которых изображены неповторяющиеся числа от 1 до 90, и объявляет очередное число. Первый из участников игры, у которого названными числами будет закрыта карта в виде таблицы, считается выигравшим. Иногда вынимающий бочонки называл не числа, а слова или выражения. Например, вместо числа 11 — “барабанные палочки”.

А вместо какого числа объявлялось слово “стульчики”?

**Вариант вопроса (отличается последней фразой)**

...А вместо числа 44 объявлялось слово “стульчики”. Почему?

3. Ведущий: “Вы, конечно, знаете старый русский счетный прибор — счеты. До их широкого распространения на Руси для расчетов использовали косточки от слив, которые выкладывались на линиях, вычерчиваемых мелом на столе и означавших единицы, десятки, сотни и т.д. Внимание — вопрос: «Почему именно от слив?»”

**Вариант вопроса (отличается последней фразой)**

...«Почему именно от слив, а не вишневых косточек?»”

4. Посмотрите, пожалуйста, на следующую таблицу:

Код	Контрольный бит
00000	0
00001	1
00010	1
00011	0
00101	0
01000	1
01001	0
01010	0
01011	1
01100	0
01101	1
01110	1
01111	0
10000	1
10010	0
10100	0

По какому принципу определяется число в правом столбце?

**Вариант вопроса**

Отличается отсутствием “шапки” в показываемой таблице:

00000	0
...	...

5. Пожалуйста, посмотрите на табличку:

0	000	
1	001	1
2	010	2
3	011	1
4	100	3
5	101	1
6	110	2
7	111	1
8	1000	?

Какое число, по вашему мнению, должно стоять на месте вопросительного знака?

6. Пожалуйста, посмотрите на табличку:

0000	
0001	1
0011	2
0010	1
0110	3
0111	1
0101	2
0100	1
1100	4
1101	1
1111	?

Какое число, по вашему мнению, должно стоять на месте вопросительного знака?

7. Если  $5 \text{ AND } 3 = 1$ , то чему равно  $5 \text{ OR } 3$ ?

**Примечание.** Выражения “ $5 \text{ AND } 3 = 1$ ” и “ $5 \text{ OR } 3$ ” представляются участникам конкурса в письменном виде или демонстрируются на экране монитора.

8. Один ученик на листе бумаги записал римскими цифрами следующее равенство:

$$I + IX = X$$

Его товарищ увидел это равенство в виде:

$$X = XI + I$$

Как это могло быть?

**Вариант вопроса (отличается числами)**

$$\begin{aligned} IX + II &= XI \\ IX &= II + XI \end{aligned}$$

**Второй вариант вопроса**

Один ученик на листе бумаги записал римскими цифрами следующие равенства:

$$\begin{aligned} I + IX &= X \\ IX + II &= XI \end{aligned}$$

Его товарищ увидел эти равенства в виде:

$$\begin{aligned} X &= XI + I \\ IX &= II + XI \end{aligned}$$

Как это могло быть?

9. Один человек увидел надпись:

R H A T

Что она означает?

**Вариант вопроса**

Один человек увидел надпись — имя девочки, причем не зашифрованное.

R H A T

Какое?

**Примечание.** Можно также использовать надпись:

R T N M

а в вопросе уточнить, что это — имя мальчика.

Кроме того, можно в вопросе привести определенную “подсказку” — сообщить, что речь идет о надписи на стекле.

10. Один человек увидел надпись:

H O V P

и понял, что это название десятичной цифры. Какой?

**Вариант вопроса**

Один человек увидел надпись:

H O V P

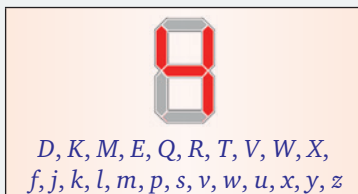
Что она означает?

11. Ведущий: “Вы знаете, что означает английское слово *love*?”. (После ответа) “А известна ли вам компьютерная фирма Hewlett Packard?” (После ответа) “А теперь — внимание на экран. Вы видите надпись (соответствующая надпись демонстрируется участникам конкурса):

L O V E H P

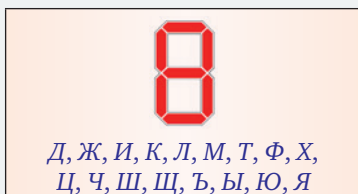
Но это не объяснение в любви к фирме Hewlett Packard, а название одной из частей ноги человека или задней конечности животного. Какой?

12. Посмотрите, пожалуйста, на картинку\*:



Что изображено на ней?

13. Посмотрите, пожалуйста, на картинку:



Что изображено на ней?

14. Посмотрите, пожалуйста, на две таблицы:

0000
0001
0011
0010
0110
0111
0101
0100
1100
1101
1111
1110
1010
1011
1001
1000

00000
00001
00011
00111
01111
11111
11110
11100
11000
10000

00000

Что общего между ними (естественно, кроме того, что в них записаны только единицы и нули)?

15. Вы, наверное, знаете, что в латинском алфавите 26 букв. В одной из распространенных компьютерных программ используется, правда не явно, 26-ричная система счисления. В какой?

**Вариант вопроса**

В одной из распространенных компьютерных программ используется, правда не явно, 26-ричная система счисления. В какой?

**Второй вариант вопроса**

В одной из распространенных компьютерных программ для нумерации столбцов используется 26-ричная система счисления. В какой?

16. Ведущий: “Вы знаете, что такое *перфолента*, или *перфорированная лента*? Это был носитель

информации в электронно-вычислительных машинах (ЭВМ) первых поколений. На них информация представлялась в двоичном виде с помощью отверстий (означало 1) и, так сказать, “их отсутствия” (отсутствие отверстия означало 0). Отверстия размещались в несколько рядов.



А теперь — внимание! Вопрос: “В одной из первых отечественных ЭВМ — модели М1 использовалась пятирядная перфолента. Основная информация (все команды и числа решаемой задачи с их адресами) набивалась во втором, третьем и четвертом рядах ленты. В первом же ряду отверстие было выполнено только в одной, повторяю — только в одной позиции. В какой? (Что оно означало?)”

17. Ведущий: “Вы знаете, что такое *перфолента*, или *перфорированная лента*? Это был носитель информации в электронно-вычислительных машинах (ЭВМ) первых поколений. На них информация представлялась в двоичном виде с помощью отверстий (означало 1) и, так сказать, “их отсутствия” (отсутствие отверстия означало 0). Отверстия размещались в несколько рядов. А теперь — внимание! Вопрос: “В одной из первых отечественных ЭВМ — модели М1 использовалась пятирядная перфолента. Основная информация (все команды и числа решаемой задачи с их адресами) набивалась во втором, третьем и четвертом рядах ленты. Таким образом, на одной позиции ленты могло быть представлено трехразрядное двоичное число (или десятичное от 0 до 7). Тем не менее на ленте можно было указать номера команд и адреса, большие 7. Каким образом?”

**Ответы**

1. Например, когда один листок бумаги разделяется на два — получается два листка.

2. Вместо числа 44 объявлялось слово “стульчики”, так как число 44, изображенное в виде



— в “перевернутом” виде напоминало схематическое изображение двух стульев:



3. Использовались именно сливовые косточки, потому что они имеют некруглую сплюснутую форму; это удобно, так как они не перекатываются с места, на которое их положили.

4. Число в правом столбце равно нулю, если в числе, записанном в левом столбце соответствующо-

\* Все изображения представлены в электронных материалах к данному номеру журнала.

щей строки, имеется четное число единиц, и равно 1 — в противном случае. Общее значение (“основные” биты и один дополнительный бит), сформированное с учетом этого правила, позволяет контролировать наличие возможной ошибки, возникшей при передаче в одном из битов.

5. 4 — число в крайнем правом столбце равно номеру разряда (при нумерации справа налево), в котором в числах во втором столбце появляется “новая” (крайняя справа) цифра 1.

6. 2 — число в правом столбце равно номеру разряда (при нумерации справа налево), в котором в числах в левом столбце появляется “новая” цифра в каждом очередном элементе.

7. В приведенных выражениях используются логические операции **AND** и **OR** над числами. Выполняются эти операции в процессоре компьютера (поэтому их называют также логическими командами) над числами, представленными в двоичном виде. Таблицы истинности для них совпадают с таблицами истинности для логических операций над значениями Истина и Ложь. Число 1 соответствует значению Истина, число 0 — значению Ложь. Эти операции являются поразрядными — в каждом разряде действия проводятся изолированно, то есть результат не зависит от значения переноса из старшего разряда:

AND	5	1	0	1
	3		1	1
		0	0	1
	5	1	0	1
OR	3		1	1
		1	1	1

Итак, ответ: 7 (111<sub>2</sub>).

8. Второй ученик увидел записанные равенства в зеркале, расположенном справа или слева от записанных равенств:

I + IX = X	X = XI + I
IX + II = XI	IX = II + XI
I + IX = X	X = XI + I
IX + II = XI	IX = II + XI

9. Приведенные надписи — имена **ТАНЯ** и **МИТЯ**, написанные на стекле и прочитанные с другой стороны стекла.

**Примечание.** В “доказательство” такого ответа можно показать участникам конкурса соответствующие надписи на стекле.

10. Приведенные надписи — это слово **НОЛЬ**, увиденное в зеркале, размещенном “вдоль” слова.



11. Приведенная надпись — это слово **ГОЛЕНЬ**, увиденное в зеркале, размещенном “вдоль” слова. Голень — это часть ноги человека или задней конечности животного от колена до стопы.



12. На рисунке слева изображен так называемый “семисегментный индикатор” — прибор, используемый для указания цифр. Иногда он применяется и для отображения букв. Но лишь немногие из букв имеют интуитивно понятное семисегментное представление. Так как количество букв ограничено, то семисегментные индикаторы используют только для отображения простейших сообщений:

ON	On
OFF	Off
STOP	Stop
CLOSE	Close
PLAY	Play
PAUSE	Pause

Справа на рисунке приведен перечень латинских букв, которые не имеют интуитивно понятного семисегментного представления.

13. На рисунке приведен перечень прописных русских букв, которые не имеют интуитивно понятного семисегментного представления (см. также ответ на предыдущий вопрос).

14. В обеих таблицах числа в двух соседних строках отличаются только в одном разряде. В левой таблице представлены числа так называемого “кода Грея”, в правой — так называемого “кода Джонсона”.

15. Речь идет об электронной таблице Microsoft Excel или подобной программе. В ней столбцы обозначаются с помощью 26 латинских букв — сначала от A до Z, потом идут столбцы AA, AB, ... до AZ, затем BA, BB, ... . Если сопоставить букве A число 1, букве B — 2 и т.д., то можно сказать, что столбцы нумеруются в 26-ричной системе счисления, начиная с 1. Настройки программы позволяют увидеть соответствующие числа в десятичной нумерации.

16. Единственное отверстие в первом ряду пробивалось в конце программы (после его считывания выполнение программы прекращалось).

17. Номера команд и адреса указывались на ленте в нескольких подряд расположенных позициях, в каждой из которых, как было сказано в вопросе, в двоичном виде представлялось число от 0 до 7, то есть номера команд и адреса указывались в восьмеричной системе счисления, например, 17 или 625.

**Примечание к ответу.** Если быть точным, то использовавшийся способ представления чисел был двоично-восьмеричным (цифры восьмеричной системы счисления представлялись в двоичном виде).



## ГРАФИКА

Занимательный практикум:  
изучаем 3D-технологии

*О.Б. Богомолова,*  
преподаватель Дворца творчества  
детей и молодежи “Преображенский”, Москва  
*Д.Ю. Усенков,* старший научный сотрудник  
Института информатизации образования РАО, Москва

► 3D-технологии<sup>1</sup> — сегодня весьма популярная, даже “модная” тема. Появляется все больше 3D- (и даже 4D- или 5D-) кинотеатров, все больше выпускается 3D-фильмов, в магазинах появляется все больше 3D-телевизоров и 3D-дисплеев для компьютеров.

Что же делать со всем этим многообразием? Что такое 3D-технологии и как ими пользоваться? Можно ли самому создавать 3D-фотографии и 3D-фильмы? В школах на уроках информатики этому пока не учат, книг про 3D тоже пока еще недостаточно. Выручает, конечно, Интернет, где можно найти кое-какие материалы и даже специальный бесплатный электронный журнал “Мир 3D/3D World”, посвященный 3D-технологиям (он выпускается с сентября 2011 года, его выпуски можно скачать с сайта <http://mir-3d-world.w.pw>), где рассказано об основных принципах объемного (трехмерного) видения, технологиях съемки и получения 3D-изображений (как еще говорят, “стереоскопических”, то есть объемных).

Многие методы получения 3D-изображений довольно сложны и требуют дорогостоящей (пока еще) аппаратуры. Но есть и возможности съемки и показа 3D-фотографий и видео, вполне доступные каждому, не требующие особых затрат на оборудование и расходные материалы. Достаточно обычного фотоаппарата и обычного

компьютера, а также нужны специальные анаглифические (двухцветные) очки: одно их стекло (или пленка) красное, а другое — синее. Такие очки можно сделать самому (как — будет сказано ниже) или купить по цене 20–30 руб. за штуку. А, немного потренировавшись, можно научиться рассматривать 3D-изображения и вовсе, как говорится, “невооруженным глазом” — без каких-либо очков или вспомогательных приспособлений (об этом рассказывалось в [1]).

Продолжая цикл занимательных занятий по информационным технологиям (см. [2]), предлагаем подборку практических заданий, позволяющих освоить 3D-технологии: создание стереофотографий и стереофоторамок, а также ознакомиться с основами 3D-видео.

Для выполнения заданий используются следующие бесплатные программы:

— графический редактор **Paint.Net** — <http://paintnet.ru>;

— программа для создания 3D-изображений **StereoPhotoMaker** — <http://stereo.jp.org/eng/stphmkr>;

— плеер 3D-видео **Stereoscopic Player** — [http://www.3dvtv.at/Downloads/Index\\_en.aspx](http://www.3dvtv.at/Downloads/Index_en.aspx);

— видеоплеер **KMPlayer** (версия с анаглифическим фильтром преобразования 2D в 3D) — <http://www.khchetverg.ru/forum/index.php?topic=1158.0>;

— конвертер 2D-изображений в 3D **2d-3d red-blue converter** — <http://sglx.web.ur.ru/converter>;

— “драйвер” для получения 3D-изображения в играх **iZ3D Driver** — <http://www.iz3d.com/driver>.

### 1. 3D-технологии: создание анаглифической фотографии из готовой стереопары

**Стереопара** — два согласованных фотокадра, отснятых с точек, расстояние между которыми по горизонтали соответствует расстоянию между зрачками глаз человека.

1. Откройте программу **StereoPhotoMaker**, щелкнув мышью на ее пиктограмме:



2. Откроется диалоговое окно программы (см. рис. 1).

<sup>1</sup> В основу 3D-технологий положена идея съемки (или создания иным способом) двух изображений для каждого глаза пользователя и воспроизведения каждого изображения из такой пары только для соответствующего ему глаза. — Прим. ред.



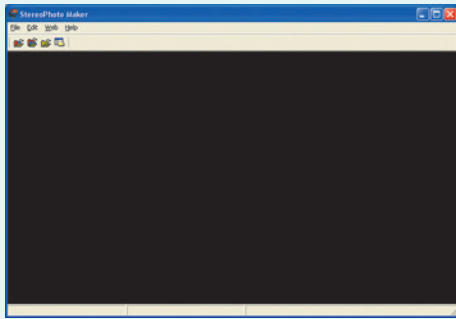


Рис. 1

3. Загрузите в программу исходную стереопару:  
 ✓ выберите в меню **File** пункт **Open Left/Right Images**  
 или

✓ щелкните на кнопке  (**Open Left and right Image**).

Откроется стандартное окно открытия файла (рис. 2):

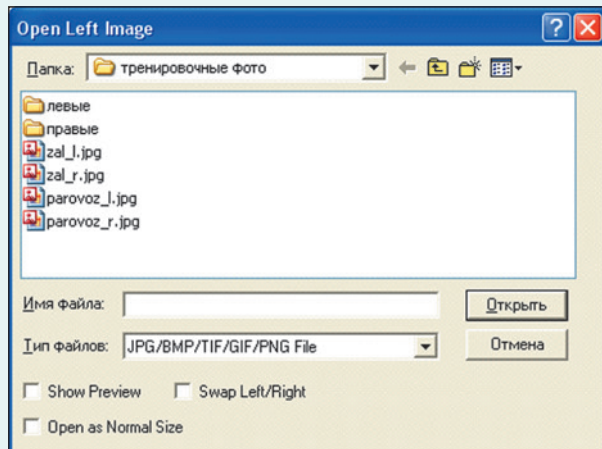


Рис. 2

При этом:

а) если файлы стереокadres имеют одинаковое начало имени и различаются только окончаниями имен (*\_l* — левый кадр, *\_r* — правый кадр), то оба кадра загружаются автоматически;

б) в других случаях окно открытия файла после выбора первого кадра (левого) остается на экране и ждет выбора второго кадра.

После загрузки фото окно программы имеет вид (см. рис. 3).

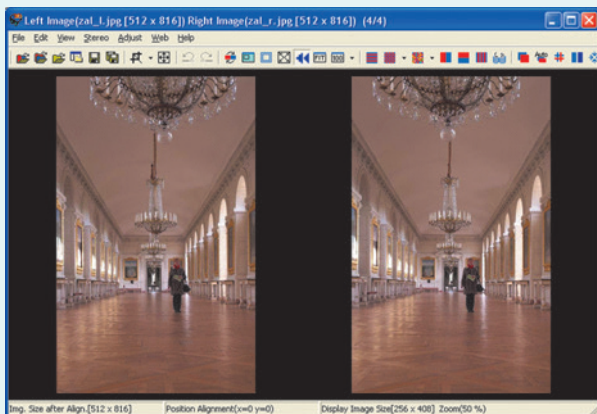



Рис. 3

4. На панели инструментов щелкните на кнопке  (четвертая справа). Дождитесь завершения обработки (что индицируется заполнением серой полоски внизу слева). В появившемся окне с информацией щелкните на кнопке **Close**:

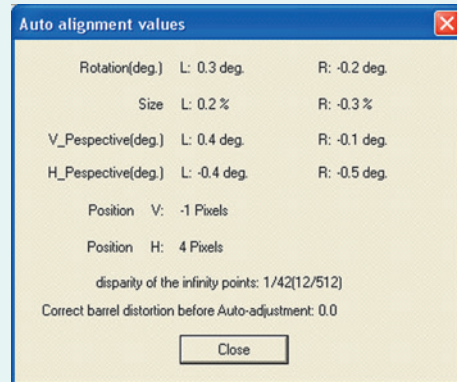






Рис. 4

При успешном выполнении операции кадр не будет слишком сильно урезан.

5. Щелкните на панели инструментов



на кнопке, соответствующей нужному типу стерео. В данном случае — на третьей слева из них ( — **Color Anaglyph**). (Первый раз надо щелкнуть на мини-кнопке  правее нее и в раскрывшемся меню выбрать пункт **color (red/cyan)**.) Вместо двух кадров в окне программы появится анаглифическое стереоизображение. (Возврат к исходному виду с двумя кадрами — кнопка .)

6. Рассмотрите полученное стереофото через очки. Если пространство “вывернуто наизнанку” (близкие предметы как бы “врезаны” в расположенные дальше них), то щелкните на кнопке  (**Swap Left/Right** — “Поменять левое и правое”). Повторный щелчок на этой же кнопке вновь меняет кадры местами.

7. Добившись достаточного качества стерео, сохраните стереофото:

✓ в меню **File** выберите пункт **Save Stereo Image** или

✓ щелкните на кнопке  (**Save Stereo Image**).

В открывшемся окне сохранения файла укажите папку, имя файла и щелкните на кнопке **Сохранить**.

## 2. 3D-технологии: создание анаглифической объемной фоторамки с объемной фотографией

1. В графическом редакторе **Paint.Net** создайте фоторамку, используя заготовки рамки (как, например, на рис. 5) и элементов оформления<sup>2</sup>, размещая их в слоях. Используя инструмент выделения прямоугольной области, выделите и кадрируйте созданную фоторамку, при необходимости обрезав пустые области справа и снизу.

**Обязательное требование:** не размещайте элементы оформления близко к краям рисунка! Оставьте отступ от краев хотя бы в 10–15 пикселей!

<sup>2</sup> Указанные заготовки можно найти в Интернете.

Сохраните полученный рисунок в файле формата *pdn* (собственный формат графического редактора Paint.Net), например, *ramka.pdn*.



Рис. 5

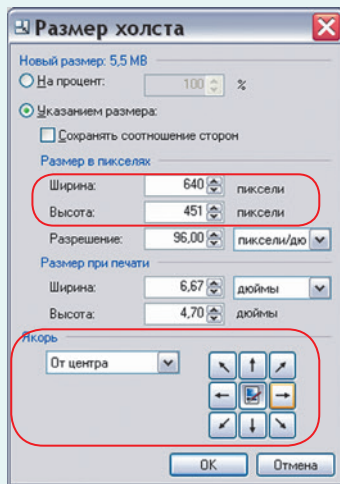


Рис. 6

2. Создайте вокруг рисунка пустые “поля”:  
— в меню **Рисунок** выберите пункт **Размер холста**;

— в открывшемся окне (см. рис. 6) в списке Якорь выберите пункт **От центра**, а затем в группе **Размер в пикселях** в полях **Ширина** и **Высота** увеличьте имеющиеся значения на 40 единиц (пикселей).




3. Сохраните результат в новый файл формата *.pdn*, добавив к имени файла (перед точкой и расширением) пару символов: “\_r”: *ramka\_r.pdn*.

4. Повторно сохраните тот же рисунок в еще один новый файл, в конце имени которого вместо “\_r” запишите “\_l”: *ramka\_l.pdn*.

5. Откройте ранее сохраненный файл с именем *ramka\_r.pdn* как второй рисунок (в появившемся окне запроса выберите верхнюю кнопку — **Открыть**), рис. 7.

6. Теперь у вас в графическом редакторе открыто два одинаковых рисунка под разными именами (из разных файлов), переключаться между ними можно щелчками мыши на миниатюрах в верхнем

правом углу редактора. Слева в миниатюрах “левый” рисунок (с буквой *l* в имени файла), справа — “правый” (с буквой *r* в имени файла).

7. Попеременно переключаясь между рисунками (и внимательно контролируя, какой именно рисунок у вас открыт — левый или правый), выбирайте по очереди слои с элементами графического оформления, выделяйте целиком соответствующие элементы и, включив инструмент перемещения выделенной области (“серая стрелка” — ) и нажимая клавиши управления курсором “влево” () и “вправо” (), смещайте выбранный элемент на небольшое расстояние:

— на левом рисунке — правее, а на правом рисунке — левее, если вы хотите сделать этот элемент “выпуклым”;

— на левом рисунке — левее, а на правом рисунке — правее, если вы хотите сделать этот элемент “вдавленным”.

Величина перемещения должна быть тем больше, чем более “выпуклым”/“вдавленным” надо сделать этот элемент оформления, но не менее 3 и не более 15 пикселей.

Например, для выпуклой “звезды”:



Рис. 8. Левый рисунок

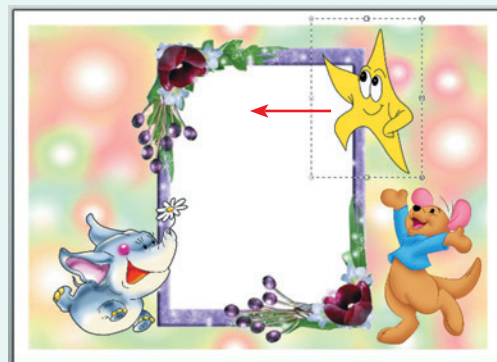


Рис. 9. Правый рисунок

8. Сохраните оба файла (под теми же именами и в формате *pdn*), а затем пересохраните их уже в формате *jpg* (при сохранении надо в первом окне

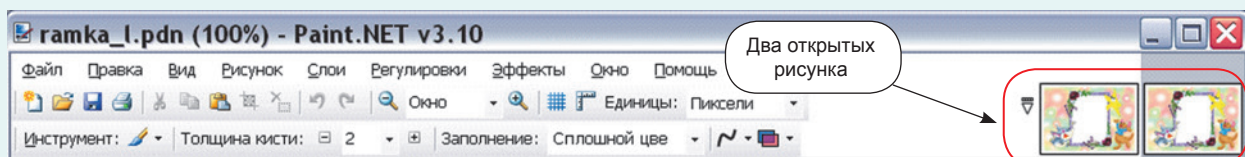



Рис. 7

с запросом качества нажать кнопку **OK**, а во втором окне с запросом на объединение слоев нажать кнопку **Flatten**).

9. Загрузите полученные картинки в качестве кадров стереопары в программу **StereoPhoto Maker**. (При выборе “левого” рисунка (с буквой *l* в имени файла) и ответе **Да** на запрос — оба рисунка будут загружены одновременно.)

Сразу же (!) переведите программу в режим анаглифа (кнопка  ▾) и сохраните полученное анаглифическое стереоизображение (рис. 10).

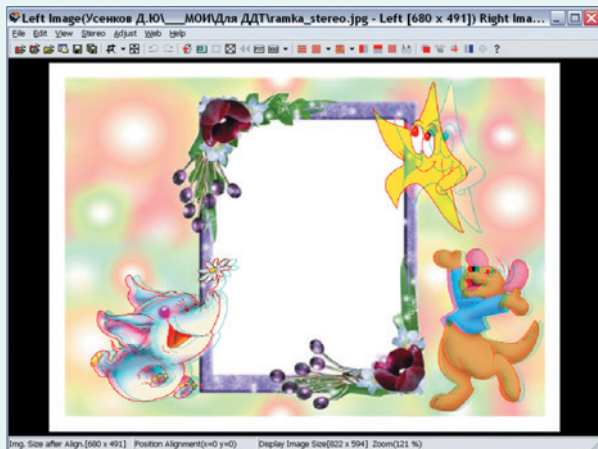


Рис. 10

10. Загрузите полученную стереорамку снова в графический редактор **Paint.Net** как слой (для этого надо при загрузке в окне запроса нажать кнопку **Добавить в слой (слои)**). Аккуратно сделайте прозрачной внутренность рамки (а также выделите слой **Фон**, с помощью инструмента “волшебная палочка” выделите все его содержимое и сделайте прозрачным).

Загрузите желаемую анаглифическую стереофотографию в отдельный слой и разместите ее “на заднем плане” (за рамкой). Смасштабируйте ее и разместите наиболее удобным способом. Кадрируйте полученное изображение так, чтобы вокруг картинки оставались небольшие белые поля (*не срежьте* фрагменты элементов оформления, которые ранее при их сдвигке “вылезли” на белые поля, — делайте от них небольшой отступ!).

11. Полученный результат — стереофото в стереорамке — сохраните сначала в отдельный файл в формате *pdn*, а затем — в формате *jpg* или другом общепринятом (при сохранении нажмите кнопку **Flatten** для объединения слоев).

Результат показан на рис. 11.

### 3. 3D-технологии: просмотр стереовидео

1. Откройте программу **Stereoscopic Player**, щелкнув мышью на ее пиктограмме:



2. Откроется диалоговое окно программы (в ответ на сообщение об истекшем сроке ключа нажмите кнопку **OK**):



Рис. 11

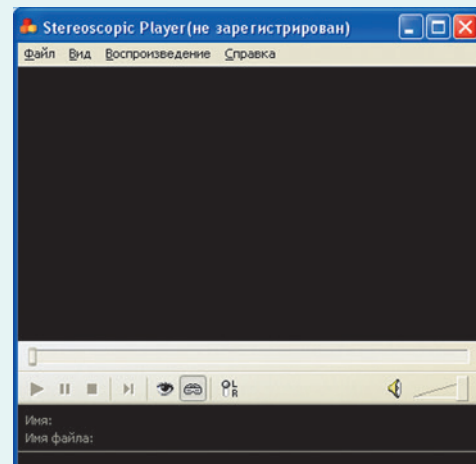


Рис. 12

3. Откройте стереофильм:

- ✓ выберите в меню **Файл** пункт **Открыть файл**;
- ✓ в окне открытия файла выберите стереофильм<sup>3</sup>.

4. Стереоплеер пытается опознать формат открываемого стереофильма автоматически и выдает окно запроса входного формата (см. рис. 13).

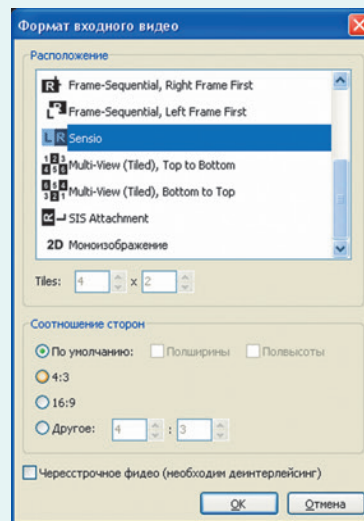


Рис. 13

Если вы точно знаете формат стереофильма, то выберите его в списке. При необходимости (если формат выбран ошибочно) повторите открытие файла, выбрав другой формат.

<sup>3</sup> Стереофильмы можно найти и скачать в Интернете. В данном случае подразумевается стереовидео в формате горизонтальной стереопары.

В данном случае нужно выбрать формат “LR Sensio”.

5. Выберите способ просмотра стереовидео:

- ✓ в меню Вид выберите пункт **Метод просмотра**;
- ✓ в открывшемся меню выберите пункт **Anaglyph**;
- ✓ в раскрывшемся вложенном меню выберите (или проверьте выбор) пункт **Optimized Anaglyph Красно-Голубой**.

6. Управление просмотром производится при помощи привычной медиановодки:



При этом кнопки переключают режим просмотра “моно/стерео”, а кнопка — меняет местами правый и левый кадры (контролируйте это визуально по стереоэффекту).

#### 4. 3D-технологии: просмотр обычного фильма как стерео

1. Откройте программу **KMPlayer**, щелкнув мышью на ее пиктограмме:



2. Откроется диалоговое окно программы (рис. 14).



Рис. 14

3. Откройте стереофильм:

- ✓ щелкните мышью на кнопке в нижней панели программы:

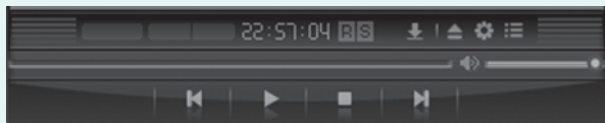


Рис. 15

- ✓ выберите файл обычного фильма.

4. Воспроизведение начинается сразу в режиме анаглифа. Управление просмотром осуществляется при помощи медиановодки:



Рис. 16

В данном случае программа **KMPlayer** при помощи встроенного специального фильтра пытается воспроизвести стереоэффект для обычного фильма. В этом простейшем случае стереоэффект слабый и скорее “домысливается” мозгом. Однако аналогичные программы-фильтры, устанавливаемые на современных стереотелевизорах, обеспечивают эффект, почти не отличимый от реального 3D, получаемого из стереопары.

#### 5. 3D-технологии: псевдостерео

1. Откройте программу **2d-3d red-blue converter**, щелкнув мышью на ее пиктограмме:



Далее нужно в появляющихся окнах нажимать следующие кнопки (рис. 17).

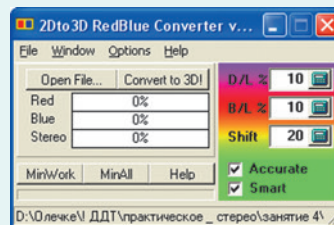
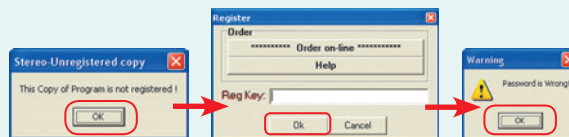


Рис. 17

2. Выберите обычную фотографию: с помощью меню **File | Open** либо кнопкой **Open File**. (Фото размерами не более 1024 × 768.)

3. Нажмите кнопку **Convert to 3D!** (см. рис. 18). Дождитесь окончания процесса.

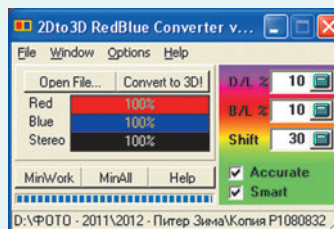


Рис. 18

4. Переключитесь в открывшееся окно с анаглифом. Если результат удовлетворительный — сохраните стереофото (сделав с него так называемый “скриншот”, нажав клавишу **PrintScreen**, после чего вставьте его в графическом редакторе).

#### 6. 3D-технологии: стереодрайвер для игр

1. **Установка**

— запустите файл **iZ3DDriverSetup.1.11b2-rus.exe**;

— выберите язык установки — **Русский**;

— выберите Я принимаю условие соглашения и нажмите кнопку **Далее**;

— в окне с надписью **Change Log** нажмите кнопку **Далее**;

— в окне выбора папки нажмите **Далее**;

— в окне выбора компонентов выберите **Полная установка** и нажмите кнопку **Далее**;

— в окне выбора папки в меню **Пуск** нажмите кнопку **Далее**;

— в окне **Все готово к установке** нажмите кнопку **Установить**;

— дождитесь конца установки (DirectX не обновлять!). В окне **Control Center** нажмите кнопку **OK**;

— в последнем окне проверьте: должны быть выбраны флажок **Запустить Control Center** и переключатель **Включить стерео**. Нажмите кнопку **Завершить**.

2. **Настройка в окне Control Center** (см. рис. 19)

— в меню слева выберите пункт **DirectX**, а в основной части окна в разделе **Вывод** — **Anaglyph (бесплатный)** и **Red/Cyan**;

— выберите переключатель **Включить стерео**;

- в нижнем правом углу нажмите кнопку **Применить**;
- закройте окно настройки.

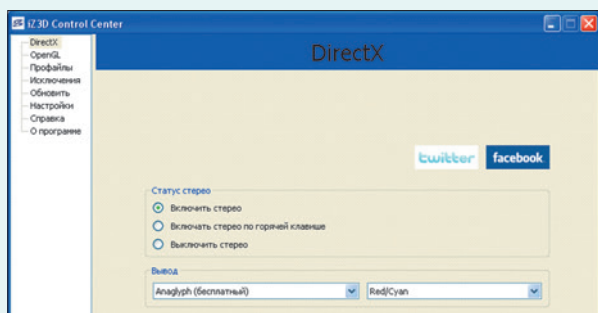



Рис. 19

Повторный запуск окна настройки — двойной щелчок на иконке  в панели задач.


### 3. Запуск

— запустить игру (как обычно). На стартовые и промежуточные сюжетные видеоролики, на меню, окна загрузки и пр. драйвер может не действовать — только в самой игре. (Если в игре стерео не получается, значит, данный драйвер с этой игрой не совместим.)

#### Возможные проблемы:

1. Если в начале игры появится диалог на английском языке — нажмите **Y**. Затем нажмите комбинацию клавиш **Shift** + **<Num>** + **\***, чтобы закрыть диалог.

2. Так как компьютер должен вырабатывать не одну, а две картинки сразу (левый и правый кадры стереопары), то игра может “подтормаживать”. В этом случае нужно снизить в настройках игры качество видео.

3. Проверка драйвера: в окне настройки (см. выше) слева выбрать **Справка**, потом в основной части окна щелкать мышью на картинках ( — выход из теста).

### 7. 3D-технологии: псевдоанаглиф

1. В графическом редакторе **Paint.Net** откройте монохромный рисунок объекта на белом фоне, например, как на рис. 20:

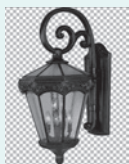


Рис. 20

2. Сделайте фон прозрачным (инструмент “волшебная палочка”, удаление выделенных участков фона):



Рис. 21

3. Установите для рисунка монохромный режим (меню **Регулировки** | **Градации серого**):

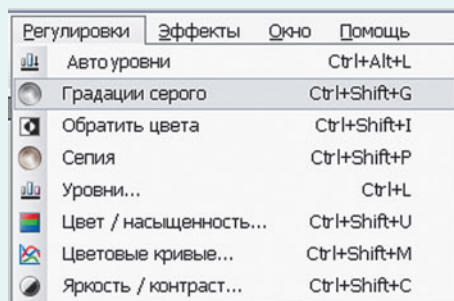


Рис. 22

4. Скопируйте весь рисунок (**Ctrl** + **A**), (**Ctrl** + **A**) и вставьте его как новый рисунок (**Правка** | **Вставить в новое изображение**).

5. Выберите первую копию рисунка. В меню выберите пункт **Регулировки** | **Цветовые кривые** (см. рис. 23).

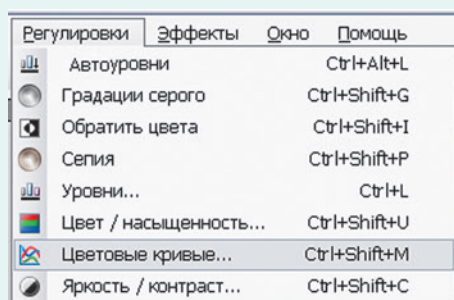


Рис. 23

6. В открывшемся окне выберите в верхнем списке вместо пункта **Яркость** пункт **RGB**. Снимите флажки **Green** и **Blue**, оставив помеченным только флажок **Red** (рис. 24). Мышь “зацепите” красную линию за середину (наведите курсор мыши на эту точку, нажмите и удерживайте левую кнопку мыши) и перетащите эту точку в верхний левый угол:

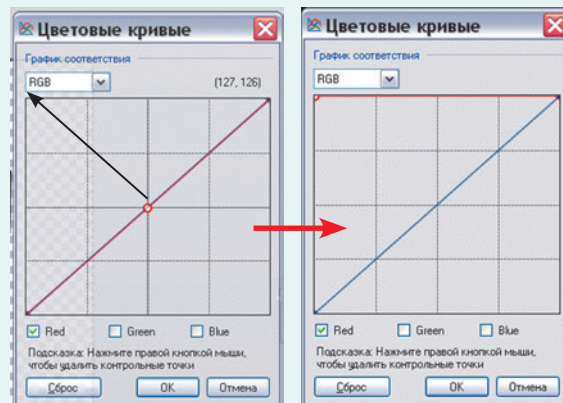


Рис. 24

Нажмите кнопку **ОК** — рисунок будет выполнен красным цветом:

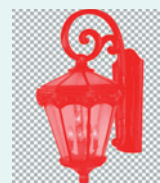


Рис. 25

7. Выбрав вторую копию рисунка, аналогичным образом вызовите для него окно **Регулировки | Цветовые кривые** (см. рис. 26). В нем нажмите кнопку **Сброс**, снимите флажок **Red** и оставьте помеченными только флажки **Green** и **Blue**. Мышью перетащите середину синей линии в верхний левый угол (аналогично тому, как это делали для красной линии предыдущего рисунка). Нажмите кнопку **ОК** — весь рисунок будет выполнен голубым цветом:

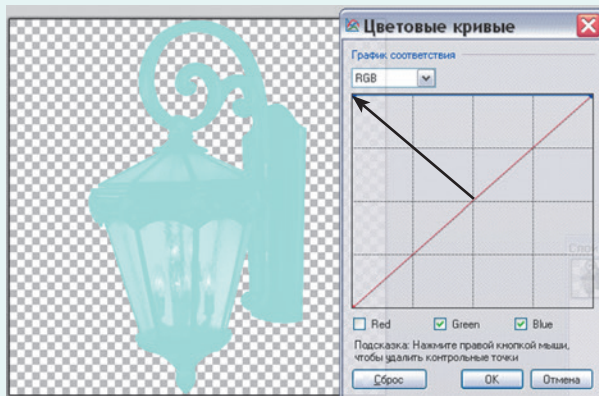


Рис. 26

8. Для каждого из двух полученных рисунков (красного и голубого):

- выберите в меню пункт **Слой | Свойства слоя**;
- в открывшемся окне в поле **Прозрачно** введите значение 150;
- нажмите кнопку **ОК**.

9. Сохраните рисунки: каждый — в своем файле формата *png*.

10. Загрузите рисунки из полученных файлов на слайд презентации **Microsoft PowerPoint** (красный должен быть сверху). Примените к ним выравнивание по центру и по середине. Выделите один рисунок (верхний по порядку слоев) и нажатиями клавиш **→** или **←** сместите его на несколько пикселей правее или левее, контролируя объемность через анаглифические очки. (Смещение надо делать небольшим.)

Результат показан на рис. 27.



Рис. 27

### Литература

1. Богомолова О.Б., Усенков Д.Ю. Стереотехнологии в школе: как и для чего. / Информатика, № 10/2012.
2. Богомолова О.Б., Усенков Д.Ю. Занимательный практикум: изучаем графический редактор. / Информатика, № 5/2013.
3. Богомолова О.Б., Усенков Д.Ю. Искусство презентации. М.: БИНОМ. Лаборатория знаний, 2010.

## Приложение

### Самостоятельное изготовление анаглифических очков

**Оправа:** можно использовать плотный картон толщиной 1–1,5 мм (например, от обувной коробки). Распечатав приведенные выкройки на принтере, вырежьте по ним части очков (для последующей склейки). Ширина “лицевой” части очков (без учета “дужек” — заушин) должна равняться 140 мм. Можно вырезать две “лицевые” части очков для последующей их склейки с проложенными между ними пленками светофильтров.

#### Светофильтры:

а) можно использовать любые готовые светофильтры соответствующих цветов;

б) можно взять любую пленку или лист целлофана и закрасить ее перманентными маркерами (фло-мастерами) соответствующих цветов;

в) можно изготовить светофильтры из специальной прозрачной пленки для печати на струйном принтере, закрасив ее тонером соответствующих цветов от струйного принтера (тонер можно извлечь из отработанных картриджей — см. ниже) либо напечатав на этой пленке квадраты нужного размера и цвета на самом струйном принтере (возможно, потребуются повторить печать несколько раз на одном и том же листе пленки до получения нужной плотности окрашивания).

В любом случае левый фильтр должен иметь красный цвет, а правый — голубой (то есть приведенные выкройки обращены передней стороной к вам).

#### Рекомендации по изготовлению светофильтров при помощи чернил от струйного принтера

1. Извлечь из использованного (но еще не засохшего) струйного картриджа порошковые губки, пропитанные остатками краски (голубой, желтой и пурпурной). Каждую губку поместить в одноразовый шприц и выдавить немного краски в любые емкости (например, в аптечные пузырьки).

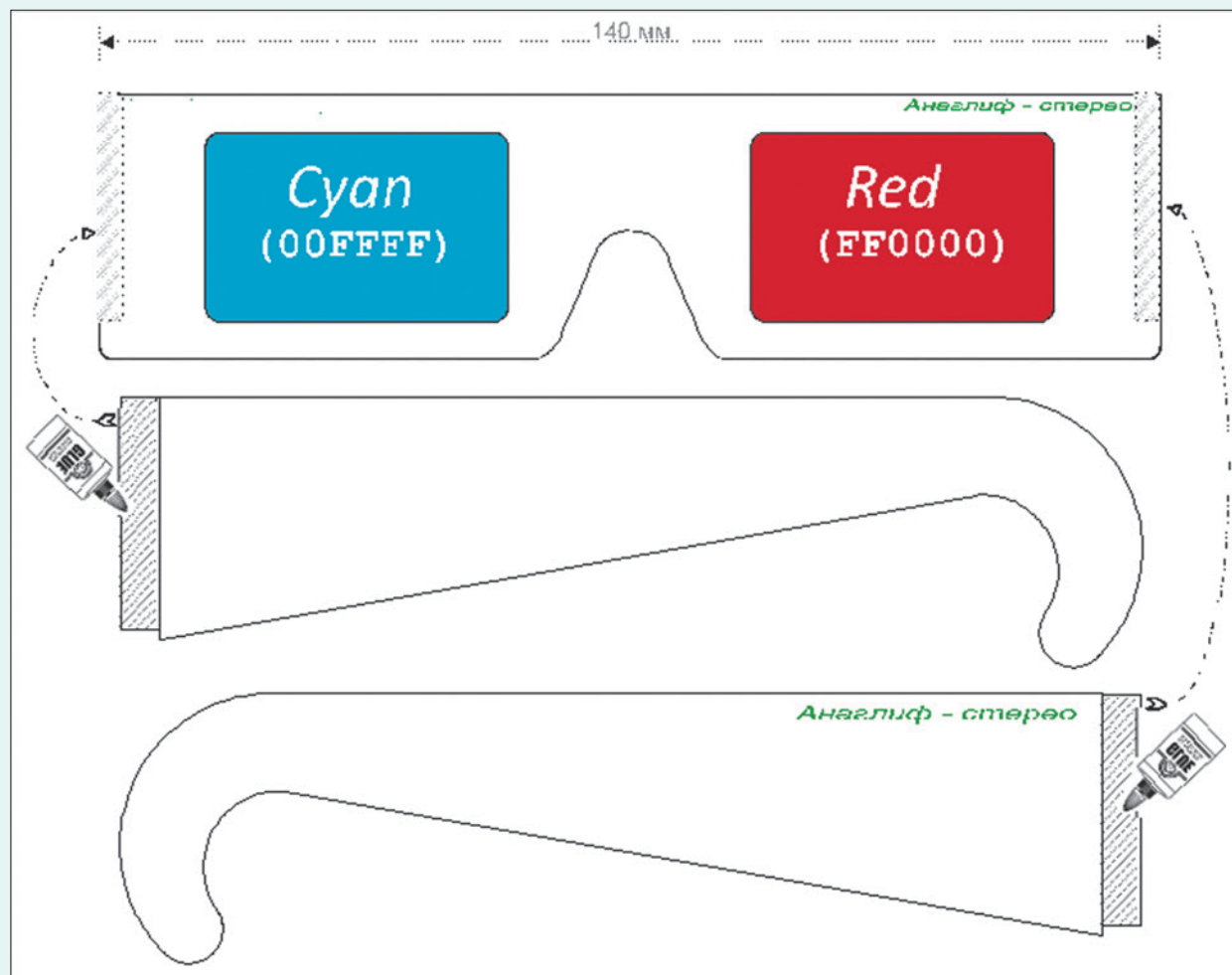
2. Голубой краситель используется сразу. Для получения красного красителя нужно смешать в равных пропорциях желтую и пурпурную краски.

3. Взять специальную прозрачную пленку для печати на струйном принтере (со специальным желатиновым слоем) и вырезать из нее четыре “стекла” нужного размера (на 3–5 мм больше, чем размеры “окон” в картонной выкройке очков). Голубую краску разделить на две примерно равные по объему порции. Одну порцию краски вылить на стеклянную пластинку и приложить к ней желатиновым слоем кусок пленки,

чтобы тонер как можно более равномерно окрасил пленку и впитался в желатиновый слой. Повторить ту же операцию со второй порцией краски и вторым куском пленки. Приложить два окрашенных в один цвет куса пленки желатиновым слоем друг к другу, получив своеобразный “сэндвич” (краска внутри, не окрашенные стороны пленки — снаружи). Повторить ту же

операцию для тонера другого цвета и двух других кусков пленки.

Полученные светофильтры наклеить с задней стороны на “лицевую” часть очков напротив вырезанных “окон”. Можно дополнительно наклеить сзади на “оправу” еще одну копию “лицевой” части очков, чтобы светофильтры оказались зажаты между двумя картонными полосками.



Красный светофильтр — для левого глаза, голубой — для правого

## ЗАДАЧНИК

**Ответы, решения, разъяснения к заданиям, опубликованным в разделе “В мир информатики” ранее**

### Задача “Численность класса”

Напомним условие: “В классе 1010 мальчиков, что составляет 101000% от численности всего класса. Сколько в классе девочек и какова общая численность класса? Ответы получите в двоичной системе, в которой записаны все приведенные значения”.

#### Решение

Все расчеты следует проводить в десятичной системе. Количество мальчиков  $1010_2 = 10_{10}$ , их доля —  $101000_2 = 40_{10}\%$ . Тогда количество девочек

равно  $15_{10}$ , или  $1111_2$ , а общая численность класса —  $11001_2$ .

Правильные ответы представили:

— Абрамов Алексей, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Андрущенко Александр, Остроухова Валерия, Пономаренко Анастасия и Уткина Ксения, Ставропольский край, Кочубеевский р-н, станция Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

— Ахматгалиева Диана, Калугин Данил, Коптелов Алексей, Одинцова Екатерина и Чурасов Михаил, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Березин Василий, Демьянова Елена и Хомякова Анна, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Коростелев Иннокентий и Ланской Дмитрий, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Крысанов Виктор, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Мухаметзянов Альфер, Республика Татарстан, г. Казань, МБОУ “Татарская гимназия № 1”, учитель **Багаутдинова Р.Р.**;

— Мякина Галина, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Надольный Артем, средняя школа поселка Ерофей Павлович, Амурская обл., Сквородинский р-н, учитель **Краснёнкова Л.А.**;

— Новикова Анна и Потапова Алевтина, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Яснова Дарья, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**

#### Задача “Три пирата”

Напомним, что в условии шла речь о трех пиратах А, Б и В, у которых состоялся такой разговор:

А: “У Б — 2 глаза”.

Б: “У В — 2 глаза”.

В: “У А — 2 глаза”.

А: “У нас 2 глаза на троих”.

Б: “У нас 3 глаза на троих”.

В: “У нас 4 глаза на троих”.

Требовалось определить, сколько глаз у каждого из пиратов, если каждый соврал столько раз, сколько у него глаз.

#### Решение

Рассмотрим второе высказывание пирата А — “У нас 2 глаза на троих”. Так как у троих людей может быть минимум три глаза (слепым пират быть не может), то оно — ложно. Значит, этот пират соврал один раз, то есть у него один глаз, а его второе высказывание является истинным, то есть у пирата В с глазами все в порядке. А из этого следует, что оба высказывания пирата В — ложны, в том числе и первое (“У В — 2 глаза”). Значит, пират В — тоже без одного глаза.

#### Правильные ответы прислали:

— Аветисова Карина, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Андрищенко Александр, Остроухова Валерия, Пономаренко Анастасия и Уткина Ксения, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

— Ахматгалиева Диана и Коптелов Алексей, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Герасимова Наталья и Костина Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Гололобов Дмитрий, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Донникова Анна, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Жукова Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Забродин Артем, г. Москва, школа № 827, учитель **Липина В.М.**;

— Иванова Ксения и Мухина Светлана, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Лазаренко Нина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

#### Задача “Игрушки”

Напомним условие: “У одного мальчика-инопланетянина было 12 игрушек. Когда ему подарили еще 5, то общее число игрушек стало равно 21. Как такое могло быть?”.

#### Решение

Прежде всего ясно, что в условии фигурируют не десятичные числа. Так как используется цифра 5, то основание соответствующей системы счисления  $p \geq 6$ . Далее, сумма цифр 2 и 5 оканчивается на 1 только в шестеричной системе. Проверка показывает, что при сложении в столбик в крайнем левом разряде суммы действительно получается 2.

*Ответ:* это могло быть в шестеричной системе счисления.

#### Правильные ответы представили:

— Абрамов Алексей, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Андрищенко Александр, Остроухова Валерия, Пономаренко Анастасия и Уткина Ксения, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

— Антипов Анатолий, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Березин Василий, Демьянова Елена и Хомякова Анна, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Коростелев Иннокентий и Ланской Дмитрий, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Крысанов Виктор и Шаламов Анатолий, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Мякина Галина, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Новикова Анна и Потапова Алевтина, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Рябых Анна, средняя школа поселка Ерофей Павлович, Амурская обл., Сквородинский р-н, учитель **Краснёнкова Л.А.**

#### Задание “Кто этот человек?” (декабрьский выпуск)

#### Ответ

В задании речь идет об американском ученом Дональде Кнуте.



*Ответы прислали:*

— Антоненко Елена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Газетдинов Рустам и Зиангиров Булат, Республика Башкортостан, г. Уфа, лицей № 60, учитель **Гильзер Н.В.**;

— Дегтярь Анатолий и Новиченко Владимир, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Жукова Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Коптелов Алексей и Чурасов Михаил, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Макеева Елена, средняя школа села Ириновка, Новобурасский р-н Саратовской обл., учитель **Брунов А.С.**;

— Мухаметзянов Альфер, Республика Татарстан, г. Казань, МБОУ “Татарская гимназия № 1”, учитель **Багаутдинова Р.Р.**;

— Новиков Сергей, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Хазиев Линар, средняя школа села Сулеево им. Р.Г. Галеева, Республика Татарстан, Альметьевский р-н, учитель **Валиева Д.И.**;

— Цаплина Ксения, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

Читатели, правильно ответившие на все вопросы, будут награждены дипломами. Так же дипломами будут отмечены читатели, правильно ответившие на все вопросы первой части задания (их список опубликован в предыдущем выпуске). Молодцы!

### **Задание “Пять вопросов” (рубрика “Поиск информации”)**

*Ответы*

1. Нападая на свою жертву, закрывает глаза, чтобы бьющаяся добыча их не поранила, — акула.

2. Братья Гонкуры, утверждая, что она при своем появлении никогда не бывает шедевром, а становится им потом, имели в виду книгу.

3. Город в Смоленской области, издревле славящийся своими пряниками и коврижками, — Вязьма.

4. Из чемпионов мира по шахматам умел легко перемножать в уме трехзначные числа Михаил Таль.

5. В Москву из Таганрога приехала музыкальная группа “Звери”.

*Ответы представили:*

— Антоненко Елена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Ахматгалиева Диана, Калугин Данил, Коптелов Алексей и Одинцова Екатерина, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Жукова Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Иванов Николай, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Леоненко Степан, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Леонова Элина, г. Воронеж, школа № 5 им. К.П. Феоктистова, учитель **Чернышева И.А.**;

— Михайлова Светлана, Чувашская Республика, г. Канаш, Канашский педагогический колледж, преподаватель **Воеводина Р.В.**;

— Новиченко Владислав, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Сурин Алексей, средняя школа поселка Ерофей Павлович, Амурская обл., Сквородинский р-н, учитель **Краснёнкова Л.А.**;

— Трофимов Сергей, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Цаплина Ксения, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

### **Задача “Сколько стоят буквы”**

Напомним, что требовалось определить, сколько “стоит” слово **TWELVE**, если слово **ONE** “стоит” 6 долларов, слово **TWO** — 9 долларов, а слово **ELEVEN** — 16 долларов. Одинаковые буквы стоят одинаково, а разные имеют различные цены.

*Правильные ответы прислали:*

— Андрищенко Александр, Остроухова Валерия, Пономаренко Анастасия и Уткина Ксения, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

— Антипов Анатолий, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Ахматгалиева Диана, Калугин Данил и Коптелов Алексей, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Иванов Николай, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Кренгель Евгений и Харламов Виталий, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Михайлова Светлана, Чувашская Республика, г. Канаш, Канашский педагогический колледж, преподаватель **Воеводина Р.В.**;

— Торопов Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Удалова Елизавета, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Филимонова Галина, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

*Решение*

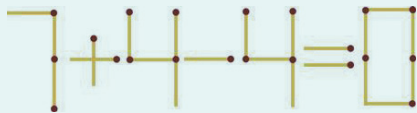
Самое короткое (и эффектное) решение такое. Если из букв слов **TWO** и **ELEVEN** удалить буквы **O**, **N** и **E** (то есть **ONE**), то из оставшихся букв можно составить слово **TWELVE** (“искомое”). Значит, “стоимость” слова **TWELVE** равна **TWO + ELEVEN – ONE = 9 + 16 – 6 = 19**.

*Ответ:* 19 долларов.

Хотя на основе аналогичных рассуждений задачу решили Диана Ахматгалиева и Павел Гируцкий, редакция решила наградить дипломами всех перечисленных читателей. Поздравляем!

### Головоломка “Получить верное равенство”

Напомним, что требовалось переложить всего одну спичку, чтобы стало верным равенство:



#### Решение

Необходимо во второй цифре 4 горизонтальную спичку расположить вертикально, получив число 11 ( $7 + 4 - 11 = 0$ ).

#### Правильные ответы представили:

— Волков Владимир и Глушаков Андрей, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Забродин Артем, г. Москва, школа № 827, учитель **Липина В.М.**;

— Иванова Мария, Круглякова Мария и Яснова Дарья, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Калугина Вероника, Попкова Татьяна и Филиппов Вадим, средняя школа села Ириновка, Новобурасский р-н Саратовской обл., учитель **Брунов А.С.**;

— Краснёнков Александр, средняя школа поселка Ерофей Павлович, Амурская обл., Сквородинский р-н, учитель **Краснёнкова Л.А.**;

— Лосева Анна, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Лошак Антон, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Минченко Валерия, средняя школа села Дохновичи Брянской обл., учитель **Клопова Е.В.**;

— Мишина Александра, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Михайлова Светлана, Чувашская Республика, г. Канаш, Канашский педагогический колледж, преподаватель **Воеводина Р.В.**;

— Мусатов Тимофей, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

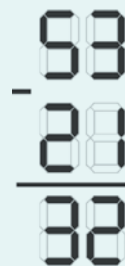
— Нуретдинов Айдар, средняя школа села Сулеево им. Р.Г. Галеева, Республика Татарстан, Альметьевский р-н, учитель **Валиева Д.И.**;

— Павленко Станислав, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

### Задача “Неисправный калькулятор”

Напомним, что требовалось установить, какое действие и с какими числами выполнялось на калькуляторе, у которого цифры изображаются с помощью так называемых “7-сегментных” индикаторов, но горели не все сегменты.

#### Ответ



#### Ответы прислали:

— Аксенов Василий, Демьянова Елена, Костюнин Александр и Хомякова Анна, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Иванов Николай, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Землянская Надежда, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Минченко Валерия, средняя школа села Дохновичи Брянской обл., учитель **Клопова Е.В.**;

— Михайлова Светлана, Чувашская Республика, г. Канаш, Канашский педагогический колледж, преподаватель **Воеводина Р.В.**;

— Новиков Сергей и Хромченкова Елизавета, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Одинцова Екатерина, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Омельченко Роман, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Яснова Дарья, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**

Все перечисленные читатели будут награждены дипломами. Поздравляем!

### Задача “Три друга”

Напомним условие: “Встретились три друга — Токарев, Слесарев, Плотников. Их отцы работают плотником, токарем и слесарем. “Интересно, что ни один из наших отцов не работает по той специальности, от которой произошла его фамилия”, — сказал мальчик, чей отец был слесарем. “А ведь ты прав”, — согласился Токарев. Кем работают отцы каждого из мальчиков?”

#### Решение

Начать рассуждения следует “с Токарева”. Так как его отец не может работать токарем, а слесарем является отец мальчика, сказавшего первую фразу, то отец Токарева — плотник. Тогда отец Слесарева — токарь, а Плотникова — слесарь.

#### Правильные ответы представили:

— Ахматгалиева Диана, Калугин Данил и Коптелов Алексей, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Бабарова Ольга, Гарифуллин Игорь, Готфрид Дарья, Гумбатов Руслан, Гумбатов Турлан, Еличева Анастасия, Жих Данила, Калиничева Анастасия, Костина Виктория, Кузьменко Алина, Маянцева Дарья, Никифоров Алексей, Оленечева Зинаида, Павлокевич Александра, Панченко Анастасия, Парфеевец Ксения, Праслова Анжела, Родионова Ольга, Сергеева Арина, Сизикова Полина и Труханова Яна, Республика Карелия, г. Сегежа, школа № 5, учитель **Меньшиков В.В.**;

— Барановская Татьяна и Жукова Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Волков Александр, Московская обл., г. Мытищи, школа “Логос М”, учитель **Куликова Т.Б.**;

— Воронова Анжелика и Хомутов Андрей, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Коростелев Иннокентий и Марун Виталий, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Краснёнков Александр, средняя школа поселка Ерофей Павлович, Амурская обл., Сквородинский р-н, учитель **Краснёнкова Л.А.**;

— Михайлова Светлана, Чувашская Республика, г. Канаш, Канашский педагогический колледж, преподаватель **Воеводина Р.В.**;

— Мухаметзянов Альфер, г. Казань, МБОУ “Татарская гимназия № 1”, учитель **Багаутдинова Р.Р.**;

— Новиченко Владислав, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Омельченко Роман, средняя школа поселка Осинка, Алтайский край, учитель **Евдокимова А.И.**;

— Талибуллина Райля, средняя школа села Сулево им. Р.Г. Галеева, Республика Татарстан, Альметьевский р-н, учитель **Валиева Д.И.**;

— Трус Анастасия, средняя школа села Дохновичи Брянской обл., учитель **Клопова Е.В.**;

— Удалова Елизавета, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

### Задача “Люди в комнатах”

Напомним условие: “В  $n$  комнатах находятся  $n + 1$  человек. На дверях первой написано: “Здесь находится один человек”, на дверях второй: “Здесь находятся два человека”, ..., на дверях  $n$ -й: “Здесь находятся  $n$  человек”. Известно, что ровно одна из этих надписей неверна. Чему может равняться  $n$  и сколько человек находится при этом в каждой комнате?”

*Решение*

Прежде всего из условия следует, что  $n > 2$ .

Может ли быть  $n = 4$ ? В этом случае общее число людей равно 5. Их распределение в четырех комнатах может быть одним из шести (если допустить, что в комнате может никого не быть):

- 1) 1, 1, 1, 2;
- 2) 0, 1, 2, 2;
- 3) 0, 1, 1, 3;

4) 0, 0, 2, 3;

5) 0, 0, 1, 4;

6) 0, 0, 0, 5.

Анализ (см. рис. 1) показывает, что ни один из перечисленных вариантов в разных перестановках не подходит (количество неверных надписей будет больше одного).

Номер комнаты	1	2	3	4
Число людей в надписи на двери	1	2	3	4

Рис. 1

Значит,  $n = 3$ . А распределение четырех человек в трех комнатах может быть таким:

Номер комнаты	1	2	3
Число людей в надписи на двери	1	2	3
Фактическое число людей	1	2	1

Рис. 2

или (если опять же допустить, что комната может быть и пустой):

Номер комнаты	1	2	3
Число людей в надписи на двери	1	2	3
Фактическое число людей	1	0	3

Рис. 3

*Ответы прислали:*

— Барановская Татьяна, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Евграфов Алексей, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Михайлова Светлана, Чувашская Республика, г. Канаш, Канашский педагогический колледж, преподаватель **Воеводина Р.В.**;

— Удалова Елизавета, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Цаплина Алена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**

### Викторина “Три вопроса”

*Ответы*

1. Нерушимый период императорской династии Цинь в Китае продлился до 1912 г. (вариант ответа **А**).

2. Надгробие, первоначально установленное на могиле Н.В. Гоголя, теперь находится на могиле Михаила Булгакова (вариант ответа **В**).

3. На картине Рафаэля “Сикстинская мадонна” у ног святого Сикста размещена тиара (вариант ответа **А**).

*Правильные ответы представили:*

— Алмазов Сергей, Иванов Иван и Яковлева Ирина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Аксенов Василий, Демьянова Елена, Костюнин Александр и Хомякова Анна, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Ахматгалиева Диана, Калугин Данил, Коптелов Алексей, Одинцова Екатерина и Чурасов Михаил, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Бородюк Анна и Василенко Татьяна, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Омельченко Роман, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Хромченкова Елизавета, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

### Пять монет на столе

Напомним условие: “Средняя монета лежит вверх орлом, остальные — решкой. За один ход разрешается перевернуть любые три рядом лежащие монеты.

1. Можно ли добиться, чтобы все монеты лежали орлом вверх?

2. А если вверх орлом первоначально лежала только первая монета?

3. А если только вторая монета?”

*Ответы*

Задачу 1 решить можно. Для этого необходимо:

1) перевернуть первые три монеты;

2) перевернуть последние три монеты.

Ответы к задачам 2 и 3 — невозможно. Существует три способа переворачивания трех рядом лежащих монет — первые три, средние три и последние три. Если проанализировать каждый вариант и учесть число переворачиваний каждой монеты (четное или нечетное) до достижения нужного положения, то окажется, что это число будет разным для разных монет, то есть имеет место противоречие.

*Правильный ответ на задачу 1 прислали:*

— Аксененко Ирина и Чумаков Илья, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Алмазов Сергей и Иванов Иван, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Ахматгалиева Диана, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Михайлова Светлана, Чувашская Республика, г. Канаш, Канашский педагогический колледж, преподаватель **Воеводина Р.В.**;

— Омельченко Роман, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**

### Задача “Три задачи”

Напомним, что в задаче 1 предлагалось ответить на вопрос, можно ли построить дерево каталогов (папок) по известной информации о связях и отношениях каталогов между собой (“корневой”, “родительский” и т.п.).

*Ответ* — можно, так как в разных папках А и В могут находиться папки с одинаковым именем (F).

В задаче 2 требовалось определить количество цифр и начальную цифру числа, равного результату выражения  $2^{354} + 2^{356}$ , записанному в системе счисления с основанием 8. Учитывая, что подобные задачи имеются в заданиях ЕГЭ по информатике, остановимся на решении подробно.

Можно сначала представить приведенные числа в двоичной системе. Так как десятичное число  $2^n$  представляет собой начальную единицу, за которой следуют  $n$  нулей (убедитесь в этом!), то можем записать:

$$\begin{array}{r} 10000\dots 0_2 \\ + \quad 100\dots 0_2 \\ \hline 10100\dots 0_2 \end{array}$$

где полужирным начертанием и цветом выделены 354 нуля. Перевести полученную сумму в восьмеричную систему можно без промежуточного перевода в десятичную систему. Для этого нужно разбить все цифры результата, начиная с крайней правой, на группы из трех цифр и каждую группу двоичных цифр представить в виде соответствующей цифры восьмеричной системы — будет получено число  $500\dots 0$  с 118 нулями.

Мухаметзянов Альфер, Республика Татарстан, г. Казань, МБОУ “Татарская гимназия № 1”, учитель **Багаутдинова Р.Р.**, предложил такое решение:

$$2^{354} + 2^{356} = 2^{354} (1 + 2^2) = (2^3)^{118} \cdot 5 = 5 \cdot 8^{118}.$$

Полученный десятичный результат можно перевести в восьмеричную систему без многократного деления на 8. Число  $8^n$  в этой системе представляет собой начальную 1 единицу и  $n$  нулей. Поэтому результат будет равен числу  $500\dots 0$  с 118 нулями. Спасибо, Альфер!

Кроме Альфера, ответы на две задачи из трех прислали:

— Довгань Алексей, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Новиков Сергей и Хромченкова Елизавета, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

Комментарии к решению задачи 3, также связанной с ЕГЭ, приведены в рубрике “Крепкий орешек”.

### Задание “Жучок в пиджачке”

Стихотворение, которое написал поэт Андрей Усачев:

*Шел по улице жучок  
в модном пиджачке.  
На груди висел значок,  
а на том значке  
нарисован был жучок  
тоже в пиджачке.  
И на нем висел значок,  
а на том значке  
был еще один жучок...*

иллюстрирует так называемую “рекурсию” — прием программирования, когда некоторая процедура

или функция вызывает (использует) саму себя как вспомогательную. (Вспомните также: “У попа была собака...”).

Встречающийся ответ “В стихотворении использован цикл (повторение одних и тех же действий)” нельзя считать правильным, так как в стихотворении не просто повторяется один и тот же фрагмент, а следующий фрагмент как бы “вызывается” из предыдущего.

Задания для самостоятельной работы, предложенные в статье “Арифметика рекордов” (ноябрьский выпуск “В мир информатики”), правильно выполнили:

— Ерашова Екатерина, Республика Татарстан, г. Нижнекамск, школа № 27, учитель **Абизяева В.Н.**;

— Филимонова Галина, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Чумаков Илья, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Галина, Екатерина и Илья будут награждены дипломами. Поздравляем!

Ответы на задачу “Малыши изучают алфавит” представили:

— Антипов Анатолий, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Кренгель Евгений и Харламов Виталий, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Михайлова Светлана, Чувашская Республика, г. Канаш, Канашский педагогический колледж, преподаватель **Воеводина Р.В.**;

— Талибуллина Райля, средняя школа села Сулево им. Р.Г. Галеева, Республика Татарстан, Альметьевский р-н, учитель **Валиева Д.И.**;

— Торопов Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**

Три числовых ребуса, связанных со словом “РОССИЯ”, правильно решили (методом рассуждений):

— Баженов Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Калугин Данил, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Леоненко Степан, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**

Правильное решение задачи “Культурные увлечения” прислали:

— Берибесова Анастасия, Дикарева Елизавета, Калинина Ирина, Журавлева Анастасия и Мохов Сергей, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Марков Алексей и Яснов Федор, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**

Задачу о разрубании кубика правильно решили: — Бальцеров Семен, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н, учитель **Краснёнкова Л.А.**;

— Донникова Анна, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Филиппенко Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Правильное решение задачи “Цифры на кубике” представили:

— Азаров Павел и Азарова Светлана, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Мусатов Тимофей, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Новиков Сергей, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Филиппенко Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

## КРЕПКИЙ ОРЕШЕК

### Головоломка “Расставить шашки”

Напомним, что требовалось так расставить шашки на клетчатой доске  $6 \times 6$ , чтобы на всех горизонталях стояло разное число шашек, а на всех вертикалях — одинаковое.

Благодаря Артема Забродина, ученика школы № 827 г. Москвы (учитель **Липина В.М.**) и Надежду Землянскую, ученицу школы № 9 г. Златоуста Челябинской обл. (учитель **Мусатова И.Б.**), приславших правильный ответ и которые будут награждены дипломами, приведем начало анализа.

Число шашек на горизонталях может быть любым — от 0 до 6. Это дает 7 вариантов. Если бы было 7 горизонталей, сумма была бы равна  $0 + 1 + 2 + 3 + 4 + 5 + 6 = 21$ . Но один ряд надо убрать, и сумма должна делиться на 6 (поскольку на каждой из шести вертикалей число шашек должно быть



одинаковым). Единственная возможность — убрать 3 (убрать 9 и расставлять 12 шашек нельзя — число 12 нельзя представить в виде суммы шести разных чисел).

Далее надо так распределить шашки на горизонталях, чтобы на каждой вертикали оказалось по 3 шашки. Предлагаем читателям решить задачу и прислать ответ в редакцию.

### Задача 3 из статьи “Три задачи” (декабрьский выпуск)

Напомним условие: “Представьте, что головка матричного принтера состоит из 24 иголок, расположенных в прямоугольнике  $4 \times 6$ . С ее помощью можно печатать различные символы — русские и латинские буквы, математические и технические знаки, ноты, знаки пре-

пинания, иероглифы, арабскую вязь и многие другие символы.

На листе бумаги формата А4 (210 × 297 мм) таким принтером можно напечатать 64 строки по 128 символов в строке.

Сколько потребуется листов бумаги для печати всех вариантов символов, которые можно напечатать с помощью принтера (каждый символ — в одном экземпляре)?

Информация о размере листа избыточная и при решении задачи не используется.

Определим, сколько различных символов можно получить с помощью печатающего устройства. Всего иголок —  $4 \times 6 = 24$ . Так как на матричном печатающем устройстве каждая иглолка может быть в двух состояниях (выдвинута и нет), то количество различных символов, которые можно напечатать на нем, —  $2^{24}$ . Так как на одном листе помещается  $64 \times 128 = 2^6 \times 2^7 = 2^{13}$  символов, то для печати всех вариантов символов потребуется  $2^{24} : 2^{13} = 2^{11} = 2048$  листов.

## ЯПОНСКИЙ УГОЛОК

Решите, пожалуйста, две японские головоломки “судоку”:

1) простую:

				4		7		
		3	2			5	8	9
1		7						
	1			7				2
		2	8	9	5	1		
	9			3				5
						8		5
7	2	5			1	3		
	4		5					



2) сложную:

2				6				9
5			4					1
		4		7				
9							3	
6				2			8	
1	8						5	
		2		9			6	
	9	6	2		5	1	4	
4					1	9		

Решения (можно не всех головоломок) присылайте в редакцию.

Уважаемые коллеги!

Для поощрения самых активных участников конкурсов, проводимых в разделе “В мир информатики”, редакция может направить вам электронный вариант диплома. Заявку на диплом просьба прислать в мае-июне 2014 года в адрес редакции ([vmi@1september.ru](mailto:vmi@1september.ru)). Оформление дипломов будет проводиться в школах.

## ПОИСК ИНФОРМАЦИИ

### Три вопроса

Ответы на следующие вопросы найдите в Интернете или по другим источникам информации.

1. Какую роль исполнял герой фильма “Берегись автомобиля”, которого играл Олег Николаевич Ефремов?

2. Какой предмет одежды чаще всего можно встретить на портретах Анны Андреевны Ахматовой?

3. Собака какой породы была впервые показана на выставке 1895 года в Мюнхене?

Можно отвечать не на все вопросы.

## ЭТО ПОЛЕЗНО ЗНАТЬ

### Про степени двойки

Это стихотворение поможет вам запомнить степени числа 2 (от 21 до 210), которые, как известно, играют большую роль в информатике.

*Слон живет у нас в квартире,  
В доме 2, подъезд 4,  
По часам привык питаться:  
Утром — в 8, днем — в 16.  
Ест на завтрак непременно 32 охапки сена.  
После утренней прогулки — 64 булки.  
На обед ему приносим огурцов 128.  
Помидоров может съесть 256.  
Съест блинов 512, — это если не стараться.  
А замесить на кефире — 1024.*

## МЫСЛИ

✓ “Мало знать, надо применять. Мало хотеть, надо и делать”.  
*Иоганн Вольфганг Гёте*

✓ “Отыщи всему начало, и ты многое поймешь”.  
*Козьма Прутков*

Хорошие цены на отличное оборудование

# MimioClassroom

Комплекты для интерактивного класса



## Интерактивная доска MimioBoard ME78

Размер доски 155x116 см (диагональ 78"). Управление стилусом, инфракрасно-ультразвуковая технология позиционирования. Износостойкая магнитная поверхность ceramicsteel™. Работа в проводном и беспроводном режимах. ПО для Windows, Mac OS и Linux.



Система голосования

## MimioVote 32

Обеспечивает подготовку и проведение тестирования, автоматически производит проверку работ и упрощает выставление оценок. Журнал ответов и оценок по отдельным ученикам и в целом по классам. Беспроводные пульты, заряжаемые на специальной док-станции перенумеровываются автоматически. 32 пользователя.

Цена комплекта

• **69 900 руб.**

~~122 400 руб.~~

**Mimio Classroom** — весь спектр интерактивного оборудования для современной школы



### MimioBoard

Стационарная интерактивная доска



### MimioTeach

Интерактивная приставка к обычной классной доске



### MimioVote

Создание и проведение тестов и контрольных



### MimioView

Документ-камера с переходником для микроскопа



### MimioProjector

Интерактивный проектор

Продажа оборудования, консультации и обучение:

<http://www.mimioclass.ru>

**8 (800) 5555-33-0**

*Звонок по России бесплатный*

ООО «Рене» — генеральный дистрибьютор Mimio в России



**mimio**  
a better way to learn



Общероссийский проект  
**Школа цифрового века**

Интернет-обеспечение проекта – Издательский дом «ПЕРВОЕ СЕНТЯБРЯ»

**2014/15** учебный год

Новый сезон  
Общероссийского проекта  
**«Школа цифрового века»**  
стартует 1 августа

**Не забудьте  
продлить участие!**

Рассказ о новшествах предстоящего сезона  
и продление участия  
на сайте

[digital.1september.ru](http://digital.1september.ru)