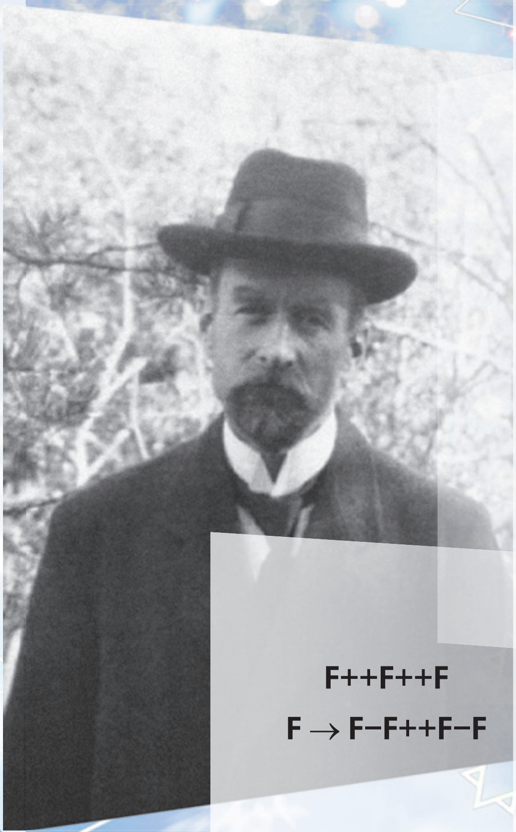
№2
Издание основано в 1995 г.
inf.1september.ru
УЧЕБНО-МЕТОДИЧЕСКИЙ ЖУРНАЛ ДЛЯ УЧИТЕЛЕЙ ИНФОРМАТИКИ

# ИНФОРМАТИКА






внутри номера
CD
и код доступа к электронной версии

издательский дом
1september.ru
Первое сентября
февраль 2013
ИНФОРМАТИКА Подписка: «Роспечать» – 32291 (бумажная версия), 19179 (электронная); «Почта России» – 79066 (бумажная версия), 12684 (электронная)

## НА ОБЛОЖКЕ

На обложке последнего зимнего номера — снежинки Коха. Классические и, возможно, самые известные фрактальные кривые. Кто только и как только их не строил ☺. Даже в различных курсах для начальной школы дети, которые только познакомились с основами какого-нибудь языка (Лого, например), уже выводят такие привычные, но такие красивые снежинки, придуманные в далеком 1904 году шведским математиком Хельге фон Кохом.

## В НОМЕРЕ



## НА ДИСКЕ


ЭЛЕКТРОННЫЕ МАТЕРИАЛЫ:

- Презентации к статьям номера
- Программа для решения задач ЕГЭ А10
- Дистрибутив программы A9CAD к статье “Что нам стоит дом построить”

ИНФОРМАТИКА

ПОДПИСНЫЕ ИНДЕКСЫ: по каталогу “Роспечати”: 32291 (бумажная версия), 19179 (электронная версия); “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 21 609 экз.
Тел. редакции: (499) 249-48-96
E-mail: 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
**Главный художник:**
Иван Лукьянов
**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** – М.Сартан
**История** – А.Савельев
**Классное руководство и воспитание школьников** – М.Битянова
**Литература** – С.Волков
**Математика** – Л.Рослова
**Начальная школа** – М.Соловейчик
**Немецкий язык** – М.Бузоева
**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
**Школьный психолог** – И.Вачков

УЧРЕДИТЕЛЬ:
ООО “ЧИСТЫЕ ПРУДЫ”
**Зарегистрировано ПИ № ФС77-44341 от 22.03.2011**
в Министерстве РФ по делам печати
Подписано в печать: по графику 15.01.2013, фактически 15.01.2013
Заказ №
Отпечатано в ОАО “Первая Образцовая типография”
Филиал “Чеховский Печатный Двор”
ул. Полиграфистов, д. 1, Московская область, г. Чехов, 142300
Сайт www.chpk.ru, E-mail: sales@chpk.ru, факс 8 (495) 988-63-87
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**


# Выбирайте выражения… общаясь с компьютером ☺

В январе околокомпьютерные СМИ опубликовали внешне забавную, но вполне серьезную новость.

Сотрудник IBM Эрик Браун, который работает “тренером” суперкомпьютера Watson, по неосторожности “скормил” ему словарь городского жаргона Urban Dictionary. Особую пикантность ситуации придало то, что Ватсона все последнее время “натаскивают” на задачи медицинской диагностики. Это востребованная и в определенной степени классическая область для задач искусственного интеллекта. В ней в 60-х годах начинались и самые первые эксперименты с экспертными системами.

Результат знакомства весьма образованного в области медицины Ватсона со специфической лексикой оказался впечатляющим. Что теперь делать, плохо понятно (по крайней мере на момент написания этой заметки не было понятно вовсе). Дело в том, что просто изъять из баз Ватсона “нехорошие” слова не так просто, если вообще возможно. “Кушая” тексты, Ватсон именно обучается, и новые знания вплетаются в сложную ткань его экспертной системы. Простым DELETE тут не обойдешься.

Суперкомпьютер Watson был представлен фирмой IBM в 2011 г. Он назван в честь основателя компании Томаса Ватсона. Основная задача суперкомпьютера — понимать вопросы, сформулированные на естественном языке, и находить на них ответы в базах данных, полученных в результате переработки структурированной и не структурированной информации. Этакая супер-Siri из 90 серверов по четыре восьмиядерных процессора каждый и с 15 терабайтами памяти.

В феврале 2011 г. Ватсон добился впечатляющего результата, победив в игре Jeopardy (российский аналог — “Своя игра”). Компьютеру тогда противостояли сильнейшие “человеческие” игроки. Во время игры Ватсон не был подключен к Интернету и оперировал лишь уже загруженными в него данными. Впрочем, учитывая объем съеденной Ватсоном информации, вряд ли это было столь важно.

Ватсон является очередным этапом исследовательских экспериментов IBM по противостоянию компьютерного разума человеку. Все прекрасно помнят захватывающую борьбу Deep Blue с Гарри Каспаровым и наверняка помнят и результат этой борьбы. Но результат Ватсона много-много круче! Речь идет уже не о шахматах — конкретной, пусть и очень сложной, но формализованной игре. Речь об экспериментах с общением на “нормальном” человеческом языке. Интересно, конечно, справился бы Ватсон с русским… Впрочем, наверняка это лишь вопрос времени.

**Сергей Островский**
(*so@1september.ru*), главный редактор

# ЕГЭ-А10: задачи с интервалами

**К.Ю. Поляков**,
д. т. н., Санкт-Петербург

*На фото:* штурмовик А10 Thunderbolt. Столь же грозное оружие, как и задача А10 ☺

## Введение

В демонстрационном варианте контрольно-измерительных материалов ЕГЭ по информатике [1] появилась новая задача на математическую логику, которая обещает быть одной из “проблемных” задач на предстоящем ЕГЭ:

***Задача 1.*** *На числовой прямой даны два отрезка: $P = [2, 10]$ и $Q = [6, 16]$. Выберите такой отрезок $A$, что формула*

$$((x \in A) \to (x \in P)) \vee (x \in Q)$$

*тождественно истинна, то есть принимает значение 1 при любом значении переменной* $\boldsymbol{x}$.

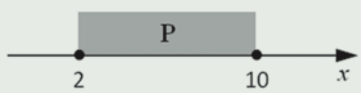
1) [0, 3]
2) [3, 11]
3) [11, 15]
4) [15, 17]

Эта задача не столько сложная, сколько “закрученная”. Во-первых, с трудом можно представить себе реальную задачу, которая сводится к такой формуле. Во-вторых, достаточно простая суть скрывается за обилием математических обозначений. Этим недостатком страдает и теоретизированное решение, приведенное в [2], которое трудно воспринимать даже человеку с высшим техническим образованием.

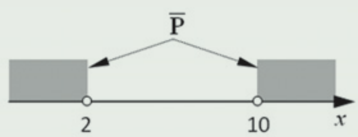
В этой заметке предлагается довольно простой метод решения подобных задач, использующий визуальное представление отрезков на числовой оси.

Прежде чем приступать к решению, приведем некоторые предварительные сведения, которыми должен владеть учащийся.

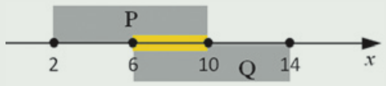
Рассмотрим интервал $P = [2, 10]$. Очевидно, что область истинности выражения $\mathbf{P}: x \in P$ представляет собой отрезок на числовой оси:




Область истинности выражения $\overline{\mathbf{P}}: x \notin P$ — это объединение интервалов $(-\infty, 2)$ и $(10, \infty)$:

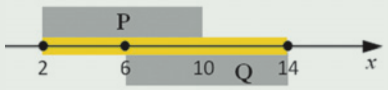


Для решения задач нам будут нужны две операции с интервалами: пересечение (определение общей части двух интервалов) и объединение. Если ввести высказывание $\mathbf{Q}: x \in Q$, то пересечение интервалов $P$ и $Q$ определяет область истинности выражения $\mathbf{P} \cdot \mathbf{Q}$[1] (она выделена желтым цветом):




Действительно, выражение $\mathbf{P} \cdot \mathbf{Q}$ истинно, если $\boldsymbol{x}$ принадлежит обоим отрезкам одновременно.

Объединение отрезков $P$ и $Q$ определяет область истинности логической суммы $\mathbf{P} + \mathbf{Q}$ ($\boldsymbol{x}$ принадлежит хотя бы одному из отрезков):




Для преобразования логических выражений нам будет нужна формула, представляющая импликацию через операции "ИЛИ" и "НЕ"[2]

$$\mathbf{A} \to \mathbf{B} = \overline{\mathbf{A}} + \mathbf{B}$$

и законы де Моргана:

$$\overline{\mathbf{A} \cdot \mathbf{B}} = \overline{\mathbf{A}} + \overline{\mathbf{B}}, \quad \overline{\mathbf{A} + \mathbf{B}} = \overline{\mathbf{A}} \cdot \overline{\mathbf{B}}.$$

## Решение задачи из демоварианта

Сначала приведем заданное выражение к более понятной форме. Введем логические высказывания

$$\mathbf{P}: x \in P, \quad \mathbf{Q}: x \in Q \quad \text{и } \mathbf{A}: x \in A.$$

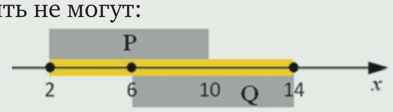
Тогда выражение, заданное в условии, запишется в форме

$$\mathbf{Z} = (\mathbf{A} \to \mathbf{P}) + \mathbf{Q}.$$

Раскрыв операцию "импликация" через "ИЛИ" и "НЕ", получаем

$$\mathbf{Z} = \overline{\mathbf{A}} + \mathbf{P} + \mathbf{Q}.$$

Это выражение должно быть истинно для любого $\boldsymbol{x}$, поэтому область истинности выражения $\mathbf{Z}$ должна охватывать всю числовую ось. Нам известны отрезки $P$ и $Q$, они конечны и всю числовую ось перекрыть не могут:




Оставшуюся часть должна перекрыть область истинности выражения $\overline{\mathbf{A}}$. Это означает, что $\overline{\mathbf{A}}$ может быть ложно только внутри отрезка [2, 14]; соответственно, выражение $\mathbf{A}$ может быть истинно только на этом отрезке. Поэтому правильный ответ — это отрезок, целиком попадающий внутрь отрезка [2, 14]. Проверка заданных вариантов ответа показывает, что верный ответ — 2 (отрезок [3, 11]).

## Вариации

В этом разделе мы рассмотрим еще несколько задач на ту же тему, которые теоретически могут встретиться в КИМ.

***Задача 2.*** *На числовой прямой даны два отрезка: $P$ = [2, 20] и $Q$ = [15, 25]. Выберите такой отрезок $A$, что формула*

$$((x \notin A) \to (x \notin P)) \vee (x \in Q)$$

*тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.*

1) [0, 15]
2) [10, 25]
3) [2, 10]
4) [15, 20]

Отличие от Задачи 1 состоит в том, что в двух скобках вместо знака "принадлежит" используется "не принадлежит". Как и раньше, введем логические высказывания

$$\mathbf{P}: x \in P, \quad \mathbf{Q}: x \in Q \quad \text{и } \mathbf{A}: x \in A.$$

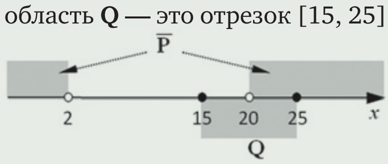
Тогда выражение, заданное в условии, запишется в форме

$$\mathbf{Z} = (\overline{\mathbf{A}} \to \overline{\mathbf{P}}) + \mathbf{Q}.$$

Раскрыв операцию "импликация" через "ИЛИ" и "НЕ", получаем

$$\mathbf{Z} = \mathbf{A} + \overline{\mathbf{P}} + \mathbf{Q}.$$

Поскольку выражение должно быть истинно для любого $\boldsymbol{x}$, области истинности всех слагаемых должны перекрыть всю числовую ось. Область $\overline{\mathbf{P}}$ состоит из двух полуосей, $(-\infty, 2)$ и $(20, \infty)$: участков числовой оси, которые не входят в отрезок [2, 20], а область $\mathbf{Q}$ — это отрезок [15, 25]:




Область истинности выражения $\mathbf{A}$ должна перекрывать оставшуюся часть — полуинтервал [2, 15) (открытый справа, потому что точка $x$ = 15 уже перекрыта отрезком $Q$). Из всех отрезков, приведенных в условии, только отрезок [0, 15] (вариант 1) полностью перекрывает полуинтервал [2, 15), это и есть правильный ответ.

***Задача 3.*** *На числовой прямой даны три отрезка: $P$ = [10, 27], $Q$ = [15, 30] и $R$ = [25, 40]. Выберите такой отрезок $A$, что формула*

$$((x \in Q) \to (x \notin R)) \wedge (x \in A) \wedge (x \notin P)$$

*тождественно* ***ложна****, то есть принимает значение 0 при любом значении переменной $\boldsymbol{x}$.*

1) [0, 15]
2) [10, 40]
3) [25, 35]
4) [15, 25]

В отличие от предыдущих задач здесь, во-первых, задано три интервала и, во-вторых,

[1] Далее конъюнкцию (логическое умножение) мы будем обозначать знаком "·", а дизъюнкцию (логическое сложение) — знаком "+". Эти обозначения, в отличие от тех, что применяются в заданиях ЕГЭ, проще воспринимаются и позволяют сразу выявлять аналогии с алгеброй.

[2] Черта сверху обозначает отрицание (инверсию) логического выражения.

требуется, чтобы выражение было тождественно **ложно** (а не истинно).

Введем логические высказывания

$$\mathbf{P}: x \in P,\ \ \mathbf{Q}: x \in Q,\ \ \mathbf{R}: x \in R\ \text{ и } \mathbf{A}: x \in A.$$

Учтем, что в формуле дважды используется знак “$\notin$” (“не принадлежит”), поэтому выражение можно записать в виде:

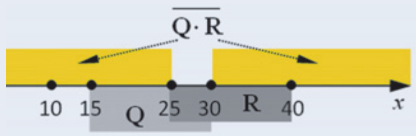
$$\mathbf{Z} = (\mathbf{Q} \to \overline{\mathbf{R}}) \cdot \mathbf{A} \cdot \overline{\mathbf{P}}$$

Представим импликацию через операции “ИЛИ” и “НЕ”:

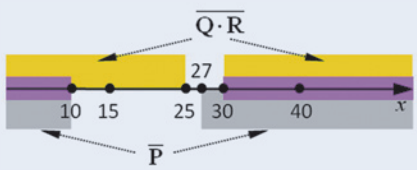
$$\mathbf{Z} = (\overline{\mathbf{Q}} + \overline{\mathbf{R}}) \cdot \mathbf{A} \cdot \overline{\mathbf{P}}$$

Это выражение должно быть тождественно ложно при всех ***x***. Поэтому роль неизвестного сомножителя **A** состоит в том, чтобы обнулить выражение везде, где произведение $(\overline{\mathbf{Q}} + \overline{\mathbf{R}}) \cdot \overline{\mathbf{P}}$ равно 1. Поэтому для этих значений ***x*** выражение **A** должно быть равно нулю, а для остальных ***x*** его значение не играет роли.

Поскольку по закону де Моргана $\overline{\mathbf{Q}} + \overline{\mathbf{R}} = \overline{\mathbf{Q} \cdot \mathbf{R}}$, область истинности выражения $\overline{\mathbf{Q}} + \overline{\mathbf{R}}$ — это область вне общей части отрезков *Q* и *R* (она показана желтым цветом на рисунке):




Теперь умножим это выражение на $\overline{\mathbf{P}}$ (ему соответствует область вне отрезка [10, 27]), построив область $(\overline{\mathbf{Q}} + \overline{\mathbf{R}}) \cdot \overline{\mathbf{P}}$; эта область, где одновременно истинны $\overline{\mathbf{Q}} + \overline{\mathbf{R}}$ и $\overline{\mathbf{P}}$, выделена на рисунке фиолетовым цветом:




В этой “фиолетовой” области выражение **A** должно быть обязательно равно 0, и только внутри отрезка [10, 30] может быть истинно. Таким образом, среди ответов нужно найти отрезок, который целиком помещается внутри отрезка [10, 30]. Этому условию удовлетворяет только отрезок [15, 25] (ответ 4).

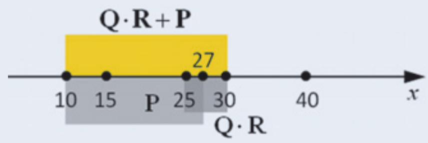
Возможен еще один подход к решению этой задачи, который фактически сводит ее к предыдущей. Для этого требуется построить обратное выражение, выполнив инверсию сложного выражения по законам де Моргана.

Выражение **Z** тождественно ложно тогда и только тогда, когда обратное ему, $\overline{\mathbf{Z}}$, тождественно истинно; таким образом, если выполнить инверсию для **Z**, мы сведем Задачу 3 к задаче из демоварианта ЕГЭ-2013, разобранной выше.

Используя законы де Моргана, получаем:

$$\overline{\mathbf{Z}} = \overline{(\overline{\mathbf{Q}} + \overline{\mathbf{R}}) \cdot \mathbf{A} \cdot \overline{\mathbf{P}}} = \overline{(\overline{\mathbf{Q}} + \overline{\mathbf{R}})} + \overline{\mathbf{A} \cdot \overline{\mathbf{P}}} = \mathbf{Q} \cdot \mathbf{R} + \overline{\mathbf{A}} + \mathbf{P}$$

Выражение **Q · R** истинно на общей части (пересечении) отрезков **Q** и **R**, то есть на отрезке [25, 30]. Добавляя к этому диапазону отрезок **P**, получим отрезок [10, 30], где истинно выражение $\mathbf{Q} \cdot \mathbf{R} + \mathbf{P}$:




Остальную часть числовой оси (при ***x*** < 10 и ***x*** > 30) должно перекрыть выражение $\overline{\mathbf{A}}$, то есть **A** должно быть ложно вне отрезка [10, 30]. Далее, аналогично предыдущему способу, находим правильный ответ 4 (отрезок [15, 25]).

***Задача 4.*** *На числовой прямой даны три отрезка: P =* [5, 10], *Q =* [10, 20] *и R =* [25, 40]. *Выберите такой отрезок A, что выражения*

$$(x \in A) \to (x \in P)\ \ \textit{и}\ \ (x \in Q) \to (x \in R)$$

*тождественно* ***равны****, то есть принимают одинаковые значения при любом значении переменной* ***x*** *(кроме, возможно, конечного количества точек).*

1) [7, 20]
2) [2, 12]
3) [10, 25]
4) [20, 30]

В этой задаче оговорка “*кроме, возможно, конечного количества точек*” означает, что в некоторых точках — на концах отрезков — заданные выражения могут иметь различные значения.

Введем логические высказывания

$$\mathbf{P}: x \in P,\ \mathbf{Q}: x \in Q,\ \mathbf{R}: x \in R\ \text{и}\ \mathbf{A}: x \in A.$$

Обозначим буквами два заданных логических выражения:

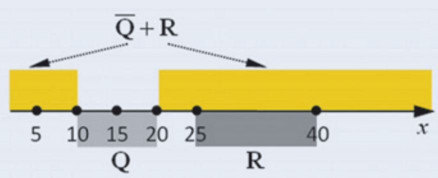
$$\mathbf{Y} = \mathbf{A} \to \mathbf{P},\ \mathbf{Z} = \mathbf{Q} \to \mathbf{R}.$$

Выразим импликации через операции “ИЛИ” и “НЕ”:

$$\mathbf{Y} = \mathbf{A} \to \mathbf{P} = \overline{\mathbf{A}} + \mathbf{P},\ \mathbf{Z} = \mathbf{Q} \to \mathbf{R} = \overline{\mathbf{Q}} + \mathbf{R}$$

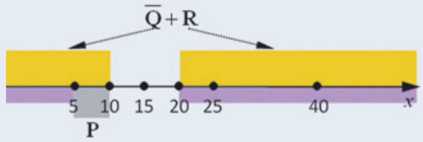
Заметим, что неизвестная величина **A** входит только в выражение **Y**. Общая идея состоит в том, чтобы построить на числовой оси область истинности для полностью известного выражения $\mathbf{Z} = \overline{\mathbf{Q}} + \mathbf{R}$, а затем дополнить отрезок *P* до этой области; это “дополнение” будет соответствовать области $\overline{\mathbf{A}}$.

Область истинности выражения $\mathbf{Z} = \overline{\mathbf{Q}} + \mathbf{R}$ состоит из отрезка *R* и области вне отрезка *Q*:




Обратите внимание, что в данном случае область $\mathbf{Z} = \overline{\mathbf{Q}} + \mathbf{R}$ (она выделена желтым цветом) совпадает с $\overline{\mathbf{Q}}$ (конечно, так будет не всегда).

Теперь рассмотрим область истинности выражения **P** (она выделена серым цветом):




Чтобы область истинности выражения $\mathbf{Y}=\overline{\mathbf{A}}+\mathbf{P}$ совпала с желтой областью, выражение $\overline{\mathbf{A}}$ должно "перекрыть" всю фиолетовую область (возможно, заходя в область **P**, но не внутрь отрезка [10, 20]). Поэтому выражение **A** обязательно должно быть истинно на отрезке [10, 20]; обязательно должно быть ложно на полуосях (–∞, 5) и (20, +∞), а на отрезке [5, 10] его значение может быть любым. Из предложенных вариантов ответов этим требованиям удовлетворяет только отрезок [7, 20] (ответ 1).

***Задача 5.*** *На числовой прямой даны три отрезка: $P$ = [10, 15], $Q$ = [5, 20] и $R$ = [15, 25]. Выберите такой отрезок $A$, что выражения*

$$(x \notin A) \to (x \in P) \text{ и } (x \in Q) \to (x \in R)$$

*принимают* ***разные*** *значения при любом значении переменной* ***x*** (кроме, возможно, конечного количества точек).

1) [7, 20]
2) [2, 15]
3) [5, 12]
4) [20, 25]

По аналогии с предыдущей задачей, здесь допускается, что заданные функции могут иметь одинаковые значения в отдельных точках — на концах отрезков.

Введем логические высказывания

$$\mathbf{P}: x \in P,\ \mathbf{Q}: x \in Q,\ \mathbf{R}: x \in R \text{ и } \mathbf{A}: x \in A.$$

Обозначим буквами два заданных логических выражения:

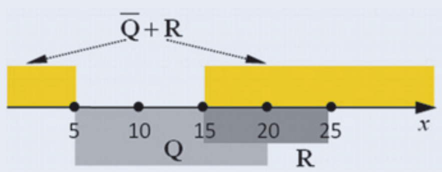
$$\mathbf{Y}=\overline{\mathbf{A}} \to \mathbf{P},\ \mathbf{Z}=\mathbf{Q} \to \mathbf{R}.$$

Выразим импликации через операции "ИЛИ" и "НЕ":

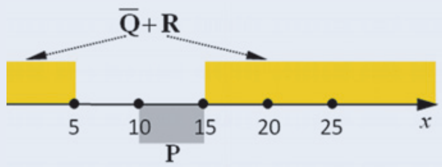
$$\mathbf{Y}=\overline{\mathbf{A}} \to \mathbf{P}=\mathbf{A}+\mathbf{P},\ \mathbf{Z}=\mathbf{Q} \to \mathbf{R}=\overline{\mathbf{Q}}+\mathbf{R}$$

Заметим, что неизвестная величина **A** входит только в выражение **Y**. Для решения задачи построим на числовой оси область истинности для полностью известного выражения $\mathbf{Z}=\overline{\mathbf{Q}}+\mathbf{R}$, а затем дополним отрезок $P$ до "обратной" области, в которой выражение **Z** ложно; это "дополнение" будет соответствовать области **A**.

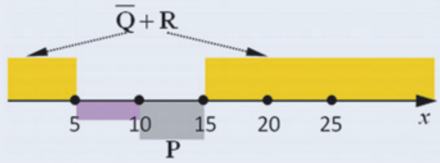
Область $\mathbf{Z}=\overline{\mathbf{Q}}+\mathbf{R}$ — это объединение отрезка $R$ и области вне отрезка $Q$:




Теперь рассмотрим область **P** (она выделена серым цветом)




Чтобы выполнить заданное условие (противоположность значений $\mathbf{Y}=\mathbf{A}+\mathbf{P}$ и $\mathbf{Z}=\overline{\mathbf{Q}}+\mathbf{R}$ при любых ***x***, за исключением конечного числа точек), область истинности выражения $\mathbf{Y}=\mathbf{A}+\mathbf{P}$ должна совпадать с областью, где выражение **Z** ложно; для этого выражение **A** должно "перекрыть" всю фиолетовую область (возможно, заходя в область **P**), но не должно заходить в "желтую" область:




Из предложенных вариантов ответов этим требованиям удовлетворяет только отрезок [5, 12] (ответ 3).

## Задачи для тренировки

Учитывая существующий дефицит литературы по этой теме, приведем несколько задач, которые можно использовать для тренировки. Ответы к этим задачам можно найти на странице *http://kpolyakov.narod.ru/school/ege.htm*. Кроме того, на диске в приложении к этому номеру размещена программа на языке Python 3, которая решает задачи этого типа.

**1.** На числовой прямой даны два отрезка: $P$ = [5, 15] и $Q$ = [12, 18]. Выберите такой отрезок $A$, что формула

$$((x \in A) \to (x \in P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной ***x***.

1) [3, 11]
2) [2, 21]
3) [10, 17]
4) [15, 20]

**2.** На числовой прямой даны два отрезка: $P$ = [5, 10] и $Q$ = [15, 18]. Выберите такой отрезок $A$, что формула

$$((x \in A) \to (x \in P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной ***x***.

1) [3, 11]
2) [6, 10]
3) [8, 16]
4) [17, 23]

**3.** На числовой прямой даны два отрезка: $P$ = [25, 30] и $Q$ = [15, 20]. Выберите такой отрезок $A$, что формула

$$((x \in A) \to (x \in P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной ***x***.

1) [10, 15]
2) [12, 30]
3) [20, 25]
4) [26, 28]

**4.** На числовой прямой даны два отрезка: $P$ = [2, 20] и $Q$ = [15, 30]. Выберите такой отрезок $A$, что формула

$$((x \notin A) \to (x \notin P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной ***x***.

1) [0, 15]
2) [3, 20]
3) [10, 25]
4) [25, 40]

**5.** На числовой прямой даны два отрезка: $P = [10, 25]$ и $Q = [0, 12]$. Выберите такой отрезок $A$, что формула

$$((x \notin A) \rightarrow (x \notin P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [10, 15]
2) [20, 35]
3) [5, 20]
4) [12, 40]

**6.** На числовой прямой даны два отрезка: $P = [10, 20]$ и $Q = [12, 15]$. Выберите такой отрезок $A$, что формула

$$((x \notin A) \rightarrow (x \notin P)) \vee (x \in Q)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [10, 15]
2) [20, 35]
3) [5, 20]
4) [12, 40]

**7.** На числовой прямой даны два отрезка: $P = [10, 20]$ и $Q = [5, 15]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \rightarrow (x \in Q)) \vee (x \in A)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [10, 15]
2) [20, 35]
3) [15, 22]
4) [12, 18]

**8.** На числовой прямой даны два отрезка: $P = [10, 20]$ и $Q = [15, 25]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \rightarrow (x \in Q)) \vee (x \in A)$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [8, 17]
2) [10, 12]
3) [15, 22]
4) [12, 18]

**9.** На числовой прямой даны два отрезка: $P = [10, 40]$, $Q = [5, 15]$ и $R = [35, 50]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \rightarrow (x \in Q)) \vee ((x \in A) \rightarrow (x \in R))$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [10, 20]
2) [15, 25]
3) [20, 30]
4) [120, 130]

**10.** На числовой прямой даны три отрезка: $P = [0, 20]$, $Q = [5, 15]$ и $R = [35, 50]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \rightarrow (x \in Q)) \vee ((x \in A) \rightarrow (x \in R))$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [–15, –5]
2) [2, 7]
3) [10, 17]
4) [15, 20]

**11.** На числовой прямой даны три отрезка: $P = [15, 30]$, $Q = [0, 10]$ и $R = [25, 35]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \rightarrow (x \in Q)) \vee ((x \in A) \rightarrow (x \in R))$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [10, 17]
2) [15, 25]
3) [20, 30]
4) [35, 40]

**12.** На числовой прямой даны три отрезка: $P = [20, 50]$, $Q = [15, 20]$ и $R = [40, 80]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \rightarrow (x \in Q)) \vee ((x \in A) \rightarrow (x \in R))$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [10, 25]
2) [20, 30]
3) [40, 50]
4) [35, 45]

**13.** На числовой прямой даны три отрезка: $P = [10, 50]$, $Q = [15, 20]$ и $R = [30, 80]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \rightarrow (x \in Q)) \vee ((x \notin A) \rightarrow (x \notin R))$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [10, 25]
2) [25, 50]
3) [40, 60]
4) [50, 80]

**14.** На числовой прямой даны три отрезка: $P = [0, 40]$, $Q = [20, 45]$ и $R = [10, 50]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \rightarrow (x \in Q)) \vee ((x \notin A) \rightarrow (x \notin R))$$

тождественно истинна, то есть принимает значение 1 при любом значении переменной $\boldsymbol{x}$.

1) [5, 20]
2) [10, 15]
3) [15, 20]
4) [35, 50]

**15.** На числовой прямой даны два отрезка: $P = [5, 15]$ и $Q = [10, 20]$. Выберите такой отрезок $A$, что формула

$$(x \in P) \wedge (x \notin Q) \wedge (x \in A)$$

тождественно ложна, то есть принимает значение 0 при любом значении переменной $\boldsymbol{x}$.

1) [0, 7]
2) [8, 15]
3) [15, 20]
4) [7, 20]

**16.** На числовой прямой даны два отрезка: $P = [12, 22]$ и $Q = [7, 17]$. Выберите такой отрезок $A$, что формула

$$(x \notin P) \wedge (x \in Q) \wedge (x \in A)$$

тождественно ложна, то есть принимает значение 0 при любом значении переменной $\boldsymbol{x}$.

1) [0, 5]
2) [7, 12]
3) [10, 20]
4) [5, 22]

**17.** На числовой прямой даны два отрезка: $P = [10, 20]$ и $Q = [5, 15]$. Выберите такой отрезок $A$, что формула

$$((x \in Q) \to (x \in P)) \wedge (x \in A)$$

тождественно ложна, то есть принимает значение 0 при любом значении переменной $\boldsymbol{x}$.

1) [0, 6]
2) [5, 8]
3) [7, 15]
4) [12, 20]

**18.** На числовой прямой даны три отрезка: $P = [15, 30]$, $Q = [5, 10]$ и $R = [20, 25]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \to (x \in Q)) \wedge ((x \notin A) \to (x \in R))$$

тождественно ложна, то есть принимает значение 0 при любом значении переменной $\boldsymbol{x}$.

1) [0, 20]
2) [0, 10]
3) [10, 15]
4) [25, 30]

**19.** На числовой прямой даны три отрезка: $P = [15, 30]$, $Q = [5, 10]$ и $R = [10, 20]$. Выберите такой отрезок $A$, что формула

$$((x \in P) \to (x \in Q)) \wedge (x \notin A) \wedge (x \in R)$$

тождественно ложна, то есть принимает значение 0 при любом значении переменной $\boldsymbol{x}$.

1) [0, 12]
2) [10, 17]
3) [15, 20]
4) [15, 30]

**20.** На числовой прямой даны три отрезка: $P = [10, 15]$, $Q = [10, 20]$ и $R = [5, 15]$. Выберите такой отрезок $A$, что выражения

$$(x \in A) \to (x \in P) \text{ и } (x \in Q) \to (x \in R)$$

тождественно равны, то есть принимают равные значения при любом значении переменной $\boldsymbol{x}$ (за исключением, возможно, конечного количества точек).

1) [5, 12]
2) [10, 17]
3) [12, 20]
4) [15, 25]

**21.** На числовой прямой даны три отрезка: $P = [5, 10]$, $Q = [15, 20]$ и $R = [25, 30]$. Выберите такой отрезок $A$, что выражения

$$(x \in A) \to (x \in P) \text{ и } (x \in Q) \to (x \notin R)$$

тождественно равны, то есть принимают равные значения при любом значении переменной $\boldsymbol{x}$ (за исключением, возможно, конечного количества точек).

1) [5, 10]
2) [15, 20]
3) [10, 20]
4) [15, 25]

**22.** На числовой прямой даны три отрезка: $P = [10, 25]$, $Q = [15, 30]$ и $R = [25, 35]$. Выберите такой интервал $A$, что формулы

$$(x \notin A) \to (x \notin P) \text{ и } (x \in Q) \to (x \in R)$$

тождественно равны, то есть принимают равные значения при любом значении переменной $\boldsymbol{x}$ (за исключением, возможно, конечного количества точек).

1) (10, 12)
2) (0, 10)
3) (5, 15)
4) (15, 25)

**23.** На числовой прямой даны три отрезка: $P = [10, 30]$, $Q = [15, 30]$ и $R = [20, 35]$. Выберите такой интервал $A$, что формулы

$$(x \notin A) \to (x \notin P) \text{ и } (x \in Q) \to (x \notin R)$$

тождественно равны, то есть принимают равные значения при любом значении переменной $\boldsymbol{x}$ (за исключением, возможно, конечного количества точек).

1) (10, 25)
2) (15, 20)
3) (15, 30)
4) (5, 20)

**24.** На числовой прямой даны три отрезка: $P = [5, 15]$, $Q = [10, 20]$ и $R = [15, 20]$. Выберите такой отрезок $A$, что выражения

$$(x \in A) \to (x \in P) \text{ и } (x \notin Q) \to (x \notin R)$$

тождественно равны, то есть принимают равные значения при любом значении переменной $\boldsymbol{x}$ (за исключением, возможно, конечного количества точек).

1) [3, 10]
2) [7, 12]
3) [12, 17]
4) [22, 25]

**25.** На числовой прямой даны три отрезка: $P = [5, 25]$, $Q = [5, 15]$ и $R = [10, 20]$. Выберите такой отрезок $A$, что выражения

$$(x \notin A) \to (x \notin P) \text{ и } (x \notin Q) \to (x \in R)$$

принимают разные значения при любом значении переменной $\boldsymbol{x}$ (за исключением, возможно, конечного количества точек).

1) [5, 12]
2) [10, 18]
3) [18, 25]
4) [20, 35]

## Литература


1. Демонстрационный вариант контрольно-измерительных материалов ЕГЭ 2013 года по информатике и ИКТ *http://www.fipi.ru/binaries/1384/inf11.zip*.

2. А10. Разбор демонстрационного варианта // Электронный ресурс (*http://ege-go.ru/a10/a10-demo/*).



*Автор благодарит*
*д. ф.-м. н. М.А. Ройтберга*
*за полезные замечания*
*по содержанию статьи.*

АВТОРСКИЕ МЕТОДИКИ БЫСТРОГО ОБУЧЕНИЯ ПРИ ПОДГОТОВКЕ К
ЕГЭ ПО ИНФОРМАТИКЕ
ПРОГРЕССИВНЫЕ ТЕХНОЛОГИИ ОБУЧЕНИЯ
Яковец Иван представляет
Видеокурс
подготовка к ЕГЭ
информатика
[ базовый уровень ]
2013
Яковец Иван | Видеокурс подготовка к ЕГЭ информатика [ базовый уровень ]
НАГЛЯДНЫЙ СПОСОБ ОБЪЯСНЕНИЯ СЛОЖНЫХ ТЕМ
ВЕСЬ МАТЕРИАЛ ОЗВУЧЕН ДИКТОРСКИМ ГОЛОСОМ
ЗАКРЕПЛЕНИЕ ПОЛУЧЕННЫХ ЗНАНИЙ НА ПРАКТИКЕ
www.ZXLINE.ru
Вам больше не надо тратить деньги на дорогих репетиторов!
Видеокурс рассчитан как на обычного школьника, так и на опытного учителя.
Все видео включает доступное объяснение и наглядные примеры.
Огромный, сложный материал объясняется легко и просто поставленным дикторским голосом.
Применение полученных знаний на практике реализуется в ходе создания и программирования реальных роботов.
Использование прогрессивных форм обучения:
интеллект карт, видеоуроков, компьютерного тестирования.

# Помехоустойчивое кодирование. Основы

**Д.М. Златопольский,**
Москва

*В процессе передачи информация подвергается различным воздействиям, которые этому мешают. Воздействия могут быть непреднамеренными (вызванными естественными причинами) или специально организованными (созданными) с какой-то целью некоторым “противником”. Непреднамеренными воздействиями на процесс передачи (помехами) могут являться уличный шум, электрические разряды (в том числе молнии), магнитные возмущения (магнитные бури), туманы, взвеси (для оптических линий связи) и т.п. Эти помехи, или, как их называют специалисты, “шумы”, искажают информацию, которая может теряться и изменяться: искажение звука в телефоне, атмосферные помехи в радио, искажение или затемнение изображения в телевидении, ошибки при передаче в телеграфе.*

## 1. Как найти ошибку?

При двоичном кодировании во время передачи информации из-за шумов возможны изменения нуля на единицу и наоборот. Возьмем сообщение *РОССИЯ*, которое при системе кодирования символов, приведенной в табл. 1[1], выглядит так: 011011111100110.

Если в первом знаке произойдет ошибка, то будет принято сообщение 111011111100110, которое декодируется (расшифровывается) в слово *СОСЯРО*[2].

Таблица 1

| Символ | Код |
|---|---|
| И | 00 |
| Р | 01 |
| О | 10 |
| Я | 110 |
| С | 111 |

[1] Приведенная в ней система кодирования называется “неравномерной” [1]. Такими, например, являются азбука Морзе, код Хаффмана и др.

[2] Убедитесь, что других вариантов декодирования нет.

Произошла непоправимая путаница, а ее "виновником" был всего лишь один неверно переданный бит.

Чтобы обнаружить и исправить ошибку, применяют так называемое "помехоустойчивое кодирование", то есть кодируют сообщение таким образом, чтобы принимающая сторона знала, произошла ошибка или нет, и часто могла исправить ошибки в случае их возникновения. Это кодирование заключается в том, что в передаваемое сообщение включаются дополнительные символы. Они не несут информации, непосредственно связанной с передаваемым сообщением, но могут дать информацию о произошедших при передаче ошибках. Иными словами, их назначение — контролировать правильность передачи основного сообщения. Поэтому вводимые дополнительные символы так и называют — *контрольными* (или *проверочными*).

Самый очевидный способ, позволяющий выявлять и исправлять ошибки, состоит в том, что каждый информационный бит 0 повторяется блоком из $n$ нулей, а каждый бит 1 — блоком из $n$ единиц. Если принять, что помехи в канале связи искажают менее половины битов в каждом передаваемом блоке, то при декодировании $n$-битового блока, содержащего, быть может, ошибочные символы, решение принимается, так сказать, "большинством голосов". Если в принятом блоке нулей больше, чем единиц, то он декодируется как 00... (то есть считается, что был послан нулевой бит), в противном случае — как 11...1. Если длину блока $n$ выбрать достаточно большой, то мы практически обезопасим себя от возможных ошибок, однако передача сообщений будет идти черепашьими темпами. По этой причине указанный код (его называют "кодом с повторением"[3]) не имеет большого практического значения, однако правило его декодирования ("голосование") содержит в себе весьма полезную идею, которая с успехом применяется в других помехоустойчивых кодах.

Заметим, что если каждый информационный бит 0 или 1 повторять блоком из $n$ нулей или единиц, не несущих полезной информации, то общая длина сообщения увеличится в $n$ раз. В теории кодирования в этом случае говорят, что пропускная способность (или скорость) кода равна $1/(1 + n)$ [3]. Кроме того, используется также такое понятие, как "избыточность", или "степень избыточности" [1]. Избыточность закодированного кода — это количество проверочной информации в сообщении. Рассчитывается она по формуле:

$$k/(i + k),$$

где

$k$ — количество контрольных битов;

$i$ — количество информационных битов.

[3] Этот способ широко применяется в житейской практике, когда, для того чтобы быть правильно понятым, нужное сообщение (слово или фразу) повторяют несколько раз.

Например, если мы передаем три бита и к ним добавляем один контрольный бит, то избыточность составит $1/(3+1) = 1/4$ (25%).

Теперь постараемся выяснить, на что мы можем рассчитывать, когда к каждому передаваемому слову добавляется всего лишь один контрольный бит. Пусть $b_1b_2...b_n$ — двоичное слово. Выберем контрольный бит $b_{n+1}$ с таким расчетом, чтобы на его значение одинаково влиял каждый из битов данного слова. Это естественное требование будет выполнено, если, например, принять, что контрольный бит $b_{n+1}$ будет равен нулю, если в слове $b_1b_2...b_n$ содержится четное число единиц, и единице — в противном случае. Например, присоединяя контрольный бит к слову 1010, получаем слово 10100, а из слова 1110 получим слово 11101.

В математике выражение для расчета значения бита $b_{n+1}$ записывается следующим образом:

$$b_{n+1} = b_1 + b_2 + \ldots + b_n \pmod 2$$

(запись "(mod 2)" читается — "равны по модулю 2").

Нетрудно увидеть, что все "удлиненные" слова $b_1b_2...b_nb_{n+1}$ при этом будут содержать четное число единиц. Это можно записать так:

$$b_1 + b_2 + \ldots + b_n + b_{n+1} = 0 \pmod 2 \qquad (1.1)$$

Допустим, что в процессе передачи удлиненного закодированного таким образом слова $b_1b_2...b_nb_{n+1}$ вкралась одна ошибка или даже любое нечетное количество ошибок. Тогда в искаженном слове $b'_1b'_2...b'_{n+1}$ число единиц станет нечетным. Это и послужит указанием на искажение в передаче слова. В конечном итоге все сводится к проверке соотношения (1.1) для битов принятого слова. Итак, правило приема следующее: если равенство (1.1) выполняется, то считаем, что сообщение передано правильно, в противном случае отмечаем, что произошла ошибка и, когда это возможно, требуем повторить передачу слова. Понятно, что иначе ошибки не исправить. Например, если принято неправильное слово 11100, в котором был искажен один бит, то одинаково возможно, что было послано любое из слов:

01100, 10100, 11000, 11110, 11101.

Описанный способ контроля выявит ошибки, которые произошли в трех и даже в пяти битах. А вот в случае двойной ошибки или вообще четного числа ошибок нас подстерегает большая неприятность — ведь тогда соотношение (1.1) не нарушится, и мы воспримем искаженное слово как верное ☹.

Рассмотренный метод кодирования сообщений, который называют "кодом с общей проверкой на четность", позволяет, следовательно, обнаружить любое нечетное число ошибок, но "пропускает" искажения, если число ошибок четно.

Код с повторением, описанный в начале, и код с общей проверкой на четность — до некоторой степени антиподы. Возможности первого исправлять ошибки теоретически безграничны, но он крайне “медлителен”. Второй очень быстр (всего один дополнительный бит), но зачастую “легкомыслен” ☺. В реальных каналах связи, как правило, приходится считаться с возможностью ошибок более чем в одном бите, поэтому в чистом виде код с общей проверкой на четность применяется крайне редко. Гораздо чаще применяют коды с несколькими проверочными битами (и, соответственно, с несколькими проверками на четность). Они позволяют не только обнаруживать, но и исправлять ошибки, и не только одиночные, но и кратные, и притом делать это гораздо эффективнее, чем упоминавшийся нами код с повторением. Это можно проиллюстрировать на одном красноречивом и в то же время простом примере.

Рассмотрим множество всех двоичных слов длины 9 (с их помощью можно закодировать $2^9 = 512$ сообщений). Расположим все биты каждого слова $b_1b_2...b_9$ в квадратной таблице следующим образом:

Таблица 2

| | | |
|---|---|---|
| $b_1$ | $b_2$ | $b_3$ |
| $b_4$ | $b_5$ | $b_6$ |
| $b_7$ | $b_8$ | $b_9$ |

К каждой строке и к каждому столбцу этой таблицы добавим еще по одному (проверочному) биту с таким расчетом, чтобы в строках и столбцах получившейся таблицы было четное число единиц (табл. 3):

Таблица 3

| | | | |
|---|---|---|---|
| $b_1$ | $b_2$ | $b_3$ | $p_1$ |
| $b_4$ | $b_5$ | $b_6$ | $p_2$ |
| $b_7$ | $b_8$ | $b_9$ | $p_3$ |
| $p_4$ | $p_5$ | $p_6$ | |

При этом, например, для первой строки и первого столбца будут выполняться проверочные соотношения:

$$p_1 = b_1 + b_2 + b_3 = 0 \pmod 2;$$
$$p_4 = b_1 + b_4 + b_7 = 0 \pmod 2.$$

(Для других строк и столбцов — аналогично.)

Заметим, что $p_1 + p_2 + p_3 = p_4 + p_5 + p_6 \pmod 2$.

Обе эти суммы равны 0, если в слове $b_1b_2...b_9$ четное число единиц, в противном случае обе они равны 1. Это дает возможность поместить в таблице еще один контрольный бит $p_7$, равный $p_7 = p_1 + p_2 + p_3 = p_4 + p_5 + p_6 \pmod 2$ — см. табл. 4.

Таблица 4

| | | | |
|---|---|---|---|
| $b_1$ | $b_2$ | $b_3$ | $p_1$ |
| $b_4$ | $b_5$ | $b_6$ | $p_2$ |
| $b_7$ | $b_8$ | $b_9$ | $p_3$ |
| $p_4$ | $p_5$ | $p_6$ | $p_7$ |

Например, слову 011010001 отвечает следующая таблица:

Таблица 5

| | | | |
|---|---|---|---|
| 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 |

Эти “маленькие хитрости” позволяют, оказывается, исправить любую одиночную ошибку, возникшую в процессе передачи, а сверх того и обнаружить любую двойную ошибку. В самом деле, если произошла одна ошибка, то нарушаются проверочные соотношения ровно для одной строки и ровно для одного столбца, как раз той строки и того столбца, на пересечении которых стоит ошибочный бит. Если же произошла двойная ошибка, то это приводит к нарушению проверок на четность либо в двух строках, либо в двух столбцах, либо сразу в двух строках и двух столбцах. По этим признакам мы и обнаруживаем двойную ошибку (однако исправить ее мы не можем — см. задание 6 для самостоятельной работы учащихся ниже).

Добавим к сказанному, что данный код позволяет обнаруживать многие, хотя и не все, ошибки более высокой кратности (в трех, четырех и т.д. битах). Например, обнаруживаются тройные ошибки в битах $b_1, b_2, b_3$ или в битах $b_1, b_2, b_6$. А вот тройная ошибка в битах $b_1, b_2, b_5$ не может быть обнаружена, она будет воспринята как одиночная.

В рассмотренном примере передаваемые слова содержат 9 информационных битов и 7 контрольных; так что общая длина слова равна 16. Такого числа символов достаточно, чтобы исправлять любые одиночные и обнаруживать любые двойные ошибки. Чтобы достичь того же эффекта для кода с повторением, нужно каждый информационный бит повторить 4 раза, так что общая длина кодового слова будет равна 36. Сравнение явно не в пользу кода с повторением.

Продемонстрированное в этом примере сочетание проверок на четность по строкам и столбцам допускает широкие обобщения. Простейшее из них следующее. Пусть сообщение кодируется двоичным словом длины $mn$. Расположим все биты в прямоугольную таблицу (матрицу) с $m$ строками и $n$ столбцами:

Таблица 6

| | | | |
|---|---|---|---|
| $b_{11}$ | $b_{12}$ | ... | $b_{1n}$ |
| $b_{21}$ | $b_{22}$ | ... | $b_{2n}$ |
| ... | ... | ... | ... |
| $b_{m1}$ | $b_{m2}$ | ... | $b_{mn}$ |

Как и ранее, добавим к каждой строке и к каждому столбцу по одному контрольному биту, так чтобы во всех строках получились четные суммы (табл. 7).

Таблица 7

| | | | | |
|---|---|---|---|---|
| $b_{11}$ | $b_{12}$ | … | $b_{1n}$ | $b_{1,n+1}$ |
| $b_{21}$ | $b_{22}$ | … | $b_{2n}$ | $b_{2,n+1}$ |
| … | … | … | … | … |
| $b_{m1}$ | $b_{m2}$ | … | $b_{mn}$ | $b_{m,n+1}$ |
| $b_{m+1,1}$ | $b_{m+1,2}$ | … | $b_{m+1,n}$ | $b_{m+1,n+1}$ |

Аналогично прежнему выбираем символ $b_{m+1,\,n+1}$. Полученное множество битов образует закодированное сообщение, с помощью которого можно находить и исправлять любые одиночные ошибки и обнаруживать любые двойные ошибки.

### Вопросы и задания для самостоятельной работы учащихся

**1.** Что такое "помехоустойчивое кодирование"? В чем оно заключается?

**2.** Определите избыточность и пропускную способность (скорость) передачи сообщения, закодированного по:

а) таблице кодировки, содержащей 256 различных символов, с одним контрольным битом для каждого символа;

б) таблице кодировки, содержащей 65 536 различных символов, с одним контрольным битом для каждого символа.

**3.** Пусть, например, передаются только прописные русские буквы, двоичные коды которых приведены в табл. 8.

Таблица 8

| Буква | Код | Буква | Код |
|---|---|---|---|
| А | 00000 | Р | 10000 |
| Б | 00001 | С | 10001 |
| В | 00010 | Т | 10010 |
| Г | 00011 | У | 10011 |
| Д | 00100 | Ф | 10100 |
| Е | 00101 | Х | 10101 |
| Ж | 00110 | Ц | 10110 |
| З | 00111 | Ч | 10111 |
| И | 01000 | Ш | 11000 |
| Й | 01001 | Щ | 11001 |
| К | 01010 | Ъ | 11010 |
| Л | 01011 | Ы | 11011 |
| М | 01100 | Ь | 11100 |
| Н | 01101 | Э | 11101 |
| О | 01110 | Ю | 11110 |
| П | 01111 | Я | 11111 |

С использованием этого кода и одного контрольного бита для каждого символа должны быть переданы сообщения:

1) БАЙТ; 3) АЛГОРИТМ;
2) ПРИВЕТ; 4) ИНФОРМАТИКА.

Были получены (соответственно) сообщения:

1) 000011000000001001010 0100;
2) 0111101000010100001100101100100010001;
3) 00000001011100011001110110001101000110 0100011000;
4) 0100010110111010000111011000010110000000 0001011010100011111111.

Есть ли ошибка в каждом из них? Ответ получите, не используя табл. 8.

Какова избыточность и скорость кода в рассмотренных случаях?

**4.** Разработайте программу, с помощью которой для заданного двоичного кода символа определяется контрольный бит и формируется соответствующий "удлиненный" код символа. Заданный и "удлиненный" коды символа представляются в виде строковой величины. Программу разработайте в двух вариантах:

1) с использованием условного оператора;
2) без его использования.

*Указания по выполнению варианта 2*

Используйте логическую операцию **XOR**, применяемую к числам (см., например, статью "Логические и сдвиговые операции" в "Информатике" № 16/2011).

**5.** Подумайте, можно ли с помощью проверки табл. 4 обнаруживать конкретный одиночный ошибочный бит без использования контрольного бита $p_7$.

**6.** Допустим, что в переданном сообщении, проверяемом с помощью табл. 4, произошла двойная ошибка. В каких случаях и что можно сказать о "месте" ошибки в этой таблице?

**7.** Разработайте программу, которая по заданному сообщению из 9 битов:

1) формирует таблицу (массив), аналогичную табл. 4;
2) при проверке полученного варианта таблицы устанавливает факт наличия одиночной ошибки;
3) при наличии одиночной ошибки:
— определяет и исправляет ошибочный бит;
— восстанавливает (декодирует) исправленное двоичное сообщение;
4) устанавливает факт наличия двойной ошибки.

**Примечание.** В качестве проверяемых согласно заданиям 2–4 таблиц (или 16-битовых сообщений) должны вводиться значения, отличающиеся на 1–2 двоичных символа от "правильных" (или совпадающие с последними).

## 2. Код Хемминга

Пусть нам требуется передавать сообщения из 16 букв. Для их кодирования можно использовать двоичные слова из четырех битов, но при этом код не будет корректировать ошибки. При использовании 5-го, контрольного, бита, как мы уже знаем, можно обнаружить, но не исправить любую одиночную ошибку. Впрочем, из первой части статьи следует, что если добавить пять контрольных битов, то можно не только находить одиночные ошибки, но и обнаруживать двойные. Возникает вопрос: нельзя ли для этой цели обойтись меньшим количеством контрольных битов?

Вычислим сначала, каково минимальное число контрольных битов, необходимое для исправления любых одиночных ошибок. Нетрудно убедиться, что двух добавочных символов для этого недостаточно (предлагаем читателю проверить это самостоятельно).

Попробуем обойтись тремя контрольными битами, то есть будем использовать для кодирования сообщений двоичные слова длины 7 (4 + 3). Наша задача — определить, произошла ли ошибка, и если произошла, то в каком месте. Или, что то же самое, — указать одно из восьми чисел от 0 до 7, соответствующих номеру ошибочного бита (0 — соответствует отсутствию ошибки).

Пусть требуется передать сообщение, кодируемое словом $b_1b_2b_3b_4$. Добавим к этому слову три бита $b_5b_6b_7$, определяемые равенствами (здесь и далее все равенства берутся по модулю 2):

$$b_5 = b_2 + b_3 + b_4;\quad b_6 = b_1 + b_3 + b_4;\quad b_7 = b_1 + b_2 + b_4. \qquad (2.1)$$

Оказывается, что с использованием трех указанных проверочных битов можно выяснить, допущена ли при передаче слова $b_1b_2b_3b_4b_5b_6b_7$ одиночная ошибка и, если допущена, определить, где именно.

Вычислим три суммы:

$$s_1 = b_4 + b_5 + b_6 + b_7; \qquad (2.2)$$
$$s_2 = b_2 + b_3 + b_6 + b_7; \qquad (2.3)$$
$$s_3 = b_1 + b_3 + b_5 + b_7. \qquad (2.4)$$

Рассмотрев все возможные варианты (полученные "вручную" или с использованием программы, разработанной согласно заданию 9 для самостоятельной работы учащихся), можно получить следующую таблицу:

Таблица 9

| $b_1$ | $b_2$ | $b_3$ | $b_4$ | $b_5$ | $b_6$ | $b_7$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| … | … | … | … | … | … | … |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Из нее[4] можно увидеть, что:

1) значение $s_1$ в правильно переданном сообщении должно быть равно 0. Если при проверке полученного сообщения оказалось, что $s_1 = 1$, то это значит, что ошибка в одном из битов $b_4$, $b_5$, $b_6$, $b_7$;

2) аналогично $s_2$ должно быть также равно нулю. В случае, когда $s_2 = 1$, ошибочно переданным является один из битов $b_2$, $b_3$, $b_6$, $b_7$ (см. табл. 9);

3) ситуация со значением $s_3$ также аналогична (при $s_3 = 1$ ошибка в одном из битов $b_1$, $b_3$, $b_5$, $b_7$).

[4] Дальнейшие выводы можно сделать и на основе анализа выражений в правой части соотношений (2.2)–(2.4).

Анализ номеров битов в трех указанных группах показывает, что номер ошибочно переданного бита может быть найден по десятичному значению двоичной записи $s_1s_2s_3$.

Пусть, например, $s_1 = 1$, $s_2 = 0$, $s_3 = 1$.

Имеем $s_1s_2s_3 = 101$, то есть ошибка в бите $b_5$.

Итак, мы имеем три проверочных соотношения:

$$\begin{aligned} s_1 &= b_4 + b_5 + b_6 + b_7 = 0,\\ s_2 &= b_2 + b_3 + b_6 + b_7 = 0,\\ s_3 &= b_1 + b_3 + b_5 + b_7 = 0, \end{aligned} \qquad (2.5)$$

которые позволяют либо установить, что ошибки нет ($s_1s_2s_3 = 000$), либо однозначно указать ее место.

Изученный здесь код называется "кодом Хемминга длины 7 с четырьмя информационными символами" [2].

В середине 1940-х годов Ричард Хемминг работал в знаменитой фирме Bell Labs на вычислительной машине Bell Model V. Это была электромеханическая машина, использующая релейные блоки, скорость которых была очень низка: один оборот за несколько секунд. Данные вводились в машину с помощью перфокарт, и поэтому в процессе чтения часто происходили ошибки. В рабочие дни использовались специальные коды, чтобы обнаруживать и исправлять найденные ошибки, при этом оператор узнавал об ошибке по свечению лампочек, исправлял и запускал машину. В выходные дни, когда не было операторов, при возникновении ошибки машина автоматически выходила из программы и запускала другую.

Хемминг часто работал в выходные дни и все больше и больше раздражался, потому что часто должен был перезагружать свою программу из-за ненадежности перфокарт. На протяжении нескольких лет он проводил много времени над построением эффективных алгоритмов исправления ошибок. В 1950 году он опубликовал способ кодирования, который на сегодняшний день известен как код Хемминга.

В общем случае слова двоичного кода Хемминга, позволяющего исправить одиночную ошибку, имеют длину $2^m - 1$ ($m$ — натуральное число, $m \geq 3$). Из них $m$ битов являются контрольными, оставшиеся $(2^m - 1 - m)$ битов — информационными. Для определения положения ошибки нужно провести $m$ проверок. Проверки строятся по аналогии с рассмотренным случаем. Значения $m$ проверок ($s_1$, $s_2$, …, $s_m$), как и выше, образуют номер положения ошибки.

Вернемся, однако, к вопросу, поставленному в начале второй части статьи. Добавим к битам кода Хемминга длины 7 еще один контрольный бит $b_0$:

$$b_0 = b_1 + b_2 + b_3 + b_4 + b_5 + b_6 + b_7 \qquad (2.6)$$

а к проверочным соотношениям (2.5) — еще одно (общую проверку на четность):

$$s_0 = b_0 + b_1 + b_2 + b_3 + b_4 + b_5 + b_6 + b_7 = 0 \qquad (2.7)$$

Новая система передачи информации по-прежнему позволит передавать 16 различных слов, потому что, как и раньше, биты $b_1$, $b_2$, $b_3$, $b_4$ могут быть какими угодно; по ним из соотношений (2.1) определяются биты $b_5$, $b_6$, $b_7$, а из равенства (2.6) и бит $b_0$. В случае одиночной ошибки добавленное соотношение (2.7) нарушается, а значения $s_1$, $s_2$, $s_3$ образуют номер положения ошибки. Если же произошла двойная ошибка, то соотношение (2.6) будет выполнено, а хотя бы одно из равенств (2.5) нарушится (**Почему?**). Это и позволяет обнаружить любую двойную ошибку. Итак, для исправления одиночных и обнаружения двойных ошибок к четырем информационным битам достаточно добавить четыре контрольных. Можно показать, что обойтись меньшим числом проверочных символов невозможно.

Построенное множество слов $b_0b_1b_2b_3b_4b_5b_6b_7$, удовлетворяющих соотношениям (2.1)–(2.7), — пример так называемого “расширенного кода Хемминга длины 8 с четырьмя информационными символами” [2].

В заключение заметим, что к настоящему времени разработано много различных систем помехоустойчивого кодирования, отличающихся друг от друга функциональным назначением, алгоритмами кодирования и декодирования, избыточностью и другими характеристиками.

## Задания для самостоятельной работы учащихся

**8.** Определите положение одиночной ошибки в искаженном сообщении 110011 кода Хемминга длины 7.

**9.** Разработайте программу, которая по четырем заданным информационным битам выводит на экран 7-битовый код Хемминга длины 7.

**10.** Подготовьте лист электронной таблицы для формирования кода Хемминга длины 7 с четырьмя информационными символами:

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | Введите 4 информационных бита: | | | | | | | |
| 2 | Полученный код: | | | | | | | |
| 3 | | *Информационные биты* | | | | *Контрольные биты* | | |

**11.** Разработайте программу, которая определяет и исправляет (устанавливает место) одиночную ошибку в искаженном сообщении кода Хемминга длины 7.

**Примечания**

1. На вход программе подается 7-битовое сообщение, которое от семи битов, приведенных в табл. 9, отличается не более чем на один бит (произвольное сообщение вводить нельзя).

2. В случае отсутствия ошибки на экран должен выводиться соответствующий текст.

**12.** Подготовьте лист электронной таблицы для проверки полученного кода Хемминга длины 7 с четырьмя информационными символами на наличие ошибки:

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | Введите полученный код: | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| 2 | | | | | | | | |
| 3 | Ошибка в бите номер 4 | | | | | | | |

**Примечания**

1. В ячейки B1:H1 вводятся 7 битов, которые от семи битов, приведенных в табл. 9, отличаются не более чем на один бит.

2. В случае отсутствия ошибки в ячейке A3 выводится соответствующий текст.

**13.** Определите общее число вариантов недопустимых (ошибочных) сообщений, которые могут быть в удлиненном слове кода Хемминга длины 7, считая, что возможно любое количество ошибок.

**14.** Определите, сколько различных слов можно передать, используя код Хемминга следующего после рассмотренного в предыдущих задачах значения длины. Каково общее число недопустимых (ошибочных) сообщений, которые могут быть при этом получены в виде удлиненного слова? Принять, что в источнике информации ошибки исключены (они возможны только при ее передаче).

**15.** Докажите, что в общем случае в коде Хемминга, позволяющем исправить одиночную ошибку, требуется именно *m* проверок.

**16.** Определите, какие биты (информационные и контрольные) должны быть использованы в каждом из проверочных соотношений, аналогичных соотношениям (2.5), применительно к коду Хемминга длины согласно заданию 14.

**17.** Пусть 11010011 и 11001111 — искаженные при передаче слова расширенного кода Хемминга длины 8. Установите, какое из этих слов содержит одиночную ошибку, а какое — двойную. В случае одиночной ошибки определите ее положение.

**18.** Разработайте программу, которая по четырем заданным информационным битам выводит на экран 8-битовый код Хемминга длины 7.

**19.** Подготовьте лист электронной таблицы для формирования кода Хемминга длины 8 с четырьмя информационными символами:

| | A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Введите 4 информационных бита: | | | | | | | | |
| 2 | Полученный код: | | | | | | | | |
| 3 | | *0-й бит* | *Информационные биты* | | | | *Контрольные биты* | | |

**20.** Разработайте программу, которая определяет одиночную или находит двойную ошибку в искаженном сообщении кода Хемминга длины 8. В случае одиночной ошибки должно выводиться сообщение о номере ошибочного бита.

**Примечание.** На вход программе подается 8-битовое сообщение, в котором по сравнению с правильным кодом имеется не более чем две ошибки (произвольное 8-битовое сообщение вводить нельзя).

**21.** Подготовьте лист электронной таблицы для проверки полученного кода Хемминга длины 8 с четырьмя информационными символами на наличие одиночной или двойной ошибки (для одиночной ошибки должен быть указан номер ошибочного бита):

| | A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Введите полученный код: | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 2 | | | | | | | | | |
| 3 | Одна ошибка в бите номер 3 | | | | | | | | |

**Примечания**

1. В ячейки B1:I1 вводится код, который от правильного отличается не более чем на два бита.

2. В случае отсутствия ошибки в ячейке A3 выводится соответствующий текст.

**22.** Ответьте (в виде рассуждений) на вопрос, оформленный синим цветом в конце основной части статьи.

*Указания по выполнению*

1. Сформулируйте условие, при котором двойная ошибка не может быть найдена при проверке по описанной методике.

2. Рассмотрите два возможных варианта:

1) один из двух ошибочных битов — $b_0$;

2) ошибочные биты — среди битов $b_1$, $b_2$, $b_3$, $b_4$, $b_5$, $b_6$, $b_7$.

## Литература


1. *Андреева Е.В., Босова Л.Л., Фалина И.Н.* Математические основы информатики. М.: Бином. Лаборатория знаний, 2005.

2. *Аршинов М.Н., Садовский Л.Е.* Коды и математика. М.: Наука, 1983.

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


## Ответы на задания для самостоятельной работы учащихся

**2.**

1) скорость передачи — 8/9; избыточность — 1/9;

2) скорость передачи — 16/17; избыточность — 1/17.

**3.** В закодированных словах БАЙТ и ИНФОРМАТИКА ошибок нет, в слове ПРИВЕТ ошибочно передана буква Е, в слове АЛГОРИТМ — буква Р.

Таблица 10

| Символ | Код | Контрольный бит |
|---|---|---|
| А | 00000 | 0 |
| Б | 00001 | 1 |
| В | 00010 | 1 |
| Г | 00011 | 0 |
| Е | 00101 | 0 |
| И | 01000 | 1 |
| Й | 01001 | 0 |
| К | 01010 | 0 |
| Л | 01011 | 1 |
| М | 01100 | 0 |
| Н | 01101 | 1 |
| О | 01110 | 1 |
| П | 01111 | 0 |
| Р | 10000 | 1 |
| Т | 10010 | 0 |
| Ф | 10100 | 0 |

### ГЕНИЙ ОДНОЙ ИДЕИ

В книге Стефана Цвейга “Звездные часы человечества” есть замечательный рассказ “Гений одной ночи” об офицере французской армии Руже де Лиле, написавшем в течение одной ночи в пылу охватившего его вдохновения знаменитую “Марсельезу”. Это было в 1792 году в революционном Марселе. Песня в течение нескольких дней распространилась по Франции, быстро приобрела колоссальную популярность во всем мире и впоследствии стала национальным гимном Французской Республики. История сохранила имя Руже в памяти потомков благодаря этой единственной песне.

По аналогии Ричарда Хемминга можно назвать “гением одной идеи”. Он сформулировал ее в 1950 году в своей единственной научной статье, посвященной кодам для коррекции ошибок. Статья содержала конструкцию блочного кода, корректирующего одиночные ошибки, которые возникают при передаче сообщений.

*Быховский Марк.* Ричард Хемминг и начало теории кодирования // PC Week/RE, № 21, 11.06.2002 г.

**4.**

1) в приведенной ниже программе на школьном алгоритмическом языке использованы следующие основные величины:

*исх_код* — заданный двоичный код символа;

*контр_бит* — соответствующий ему контрольный бит;

*удл_код* — “удлиненный” код символа;

*кол*1 — количество единиц в заданном коде.

```
алг Формирование_удлиненного_кода
нач лит исх_код, удл_код,
    сим контр_бит, цел i, кол1
  ввод исх_код
  кол1 := 0
  нц для i от 1 до длин(исх_код)
    |Каждый символ исходного кода
    |сравниваем с единицей
    если исх_код[i] = "1"
      то
        кол1 := кол1 + 1
    все
  кц
  |Определяем контрольный бит,
  |проверяя "четность" значения кол1
  если mod(кол1, 2) = 0
    то
      контр_бит := "0"
    иначе
      контр_бит := "1"
  все
  |Формируем удлиненный код символа
  удл_код := исх_код + контр_бит
  |Выводим ответ
  вывод нс, удл_код
кон
```

2)

```
алг Формирование_удлиненного_кода
нач лит исх_код, удл_код, сим_контр_бит,
    цел i, цифр_бит, цифр_контр_бит,
    лог успех
  ввод исх_код
  |Значение контрольного бита в виде
  |числа (цифры)
  цифр_контр_бит := 0 |Начальное значение
  нц для i от 1 до длин(исх_код)
    |Преобразуем очередной символ кода
    |в число
    цифр_бит := лит_в_цел(исх_код[i], успех)
    |Учитываем это число в общем значении
    |величины цифр_контр_бит
    цифр_контр_бит := цифр_контр_бит
                      XOR цифр_бит
  кц
  |Контрольный бит рассчитан
  |Преобразуем его в символ
  сим_контр_бит := цел_в_лит(цифр_контр_бит)
  |и добавляем к исходному коду
  удл_код := исх_код + сим_контр_бит
  |Выводим ответ
  вывод нс, удл_код
кон
```

**Примечание.** Логическая операция XOR использована условно (в школьном алгоритмическом языке такая операция отсутствует).

**5.** Это можно сделать только в случаях, когда ошибка произошла в одном из битов $b_1, b_2, \ldots, b_9$. Для ошибок в контрольных битах можно установить только номер строки или номер столбца с ошибочным битом.

**6.** Если обе ошибки произошли в одной строке, то можно установить только номера столбцов, в которых это имело место, если в одном столбце — только номера строк, в противном случае — номера двух столбцов и двух строк, но конкретные их сочетания с ошибками установить невозможно.

**8.** Ошибочный бит — второй ($s_1 = 0$, $s_2 = 1$, $s_3 = 0$).

**13.** $2^7 - 2^4 = 112$.

**14.** $2^{11}$ (следующая длина двоичного кода Хемминга, позволяющего исправить одиночную ошибку, — $2^4 - 1 = 15$, из них контрольных битов — 4, информационных — 11).

Общее число недопустимых (ошибочных) сообщений — $2^{15} - 2^{11} = 30\,720$.

**15.** Ошибка может иметь номер от 0 до $2^m - 1$. В двоичном виде число $2^m - 1$ — $m$-значное. Значит, согласно методике определения места одиночной ошибки, требуются $m$ проверок (необходимо найти $m$ значений контрольных сумм).

**16.**

$s_1 = b_8 + b_9 + b_{10} + b_{11} + b_{12} + b_{13} + b_{14} + b_{11}$,

$s_2 = b_4 + b_5 + b_6 + b_7 + b_{12} + b_{13} + b_{14} + b_{15}$,

$s_3 = b_2 + b_3 + b_6 + b_7 + b_{10} + b_{11} + b_{14} + b_{15}$,

$s_4 = b_1 + b_3 + b_5 + b_7 + b_9 + b_{11} + b_{13} + b_{15}$.

**Примечание.** При расчете значения $s_1$ происходит суммирование битов, номера которых обладают следующим свойством — в их двоичной записи в четвертом справа разряде (с весомостью 8) представлена единица; при расчете значений $s_2$, $s_3$, $s_4$ — битов, двоичные номера которых имеют единицу соответственно в третьем, втором и первом справа разрядах (с весомостью 4, 2 и 1).

**17.** В слове 11010011 — одиночная ошибка в третьем бите ($s_0 = 1$, $s_1 = 0$, $s_2 = 1$, $s_3 = 1$), в слове 11001111 — двойная ($s_0 = 0$, $s_1 = 0$, $s_2 = 0$, $s_3 = 1$).

**22.** Любая двойная ошибка не может быть обнаружена при проверке по описанной методике — если ни одно из равенств (2.5) не нарушится (при выполнении отношения (2.7)). Это могло бы быть, если бы имелась хотя бы одна пара битов, оба бита которой либо встречаются в отдельных равенствах (2.5), либо отсутствуют в них. Исследуем такую возможность.

Когда один из двух ошибочных битов — $b_0$, то второй обязательно нарушит одно из равенств (2.5), как при одиночной ошибке.

Когда ошибочные биты — среди битов $b_1$, $b_2$, $b_3$, $b_4$, $b_5$, $b_6$, $b_7$, то для любой пары из них в каком-то из равенств (2.5) некоторый бит пары встречается один раз (убедитесь в этом!), что также нарушит это равенство.

Следовательно, при наличии двух ошибочных битов отношение (2.7) будет выполняться, а одно из равенств (2.5) — обязательно нарушится.

МОСКОВСКИЙ ПЕДАГОГИЧЕСКИЙ МАРАФОН УЧЕБНЫХ ПРЕДМЕТОВ

ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ г. МОСКВЫ
ИЗДАТЕЛЬСКИЙ ДОМ «ПЕРВОЕ СЕНТЯБРЯ»
МОСКОВСКИЙ ИНСТИТУТ ОТКРЫТОГО ОБРАЗОВАНИЯ

# 2013

25 МАРТА – 19 АПРЕЛЯ

## РАСПИСАНИЕ ДНЕЙ ПЕДАГОГИЧЕСКОГО МАРАФОНА

| | |
|---|---|
| **25 марта** | День учителя технологии * |
| **26 марта** | **Открытие Марафона**<br>День классного руководителя |
| **27 марта** | День школьного психолога<br>День учителя ОБЖ |
| **28 марта** | День здоровья детей, коррекционной педагогики, логопеда, инклюзивного образования и лечебной физической культуры |
| **29 марта** | День учителя начальной школы (день первый) |
| **30 марта** | День учителя начальной школы (день второй) |
| **31 марта** | День дошкольного образования |
| **2 апреля** | День учителя географии |
| **3 апреля** | День учителя химии |
| **4 апреля** | День учителя биологии |
| **5 апреля** | День учителя информатики |
| **6 апреля** | День учителя физики |
| **7 апреля** | День учителя математики |
| **9 апреля** | День учителя истории и обществознания |
| **10 апреля** | День учителя МХК, музыки и ИЗО |
| **11 апреля** | День школьного и детского библиотекаря |
| **12 апреля** | День учителя литературы |
| **13 апреля** | День учителя русского языка |
| **14 апреля** | День учителя английского языка |
| **16 апреля** | День учителя французского языка |
| **17 апреля** | День учителя немецкого языка |
| **18 апреля** | День учителя физической культуры |
| **19 апреля** | День школьной администрации<br>**Закрытие** |

## marathon.1september.ru

! Обязательная предварительная регистрация на все дни Марафона откроется 20 февраля 2013 года на сайте marathon.1september.ru.

! Каждый участник Марафона, посетивший три мероприятия одного дня, получает официальный именной сертификат (6 часов)

В дни Марафона ведущие издательства страны представляют книги для учителей

Начало работы – 9.00. Завершение работы – 15.00

## УЧАСТИЕ БЕСПЛАТНОЕ. ВХОД ПО БИЛЕТАМ

**РЕГИСТРИРУЙТЕСЬ, РАСПЕЧАТЫВАЙТЕ СВОЙ БИЛЕТ И ПРИХОДИТЕ!**

Место проведения Марафона: лицей № 1535, ул. Усачева, дом 50 (в 3 минутах ходьбы от станции метро «Спортивная»)
* Место проведения Дня учителя технологии: ЦО № 293, ул. Ярославская, д. 27 (ст. метро «ВДНХ»)

По всем вопросам обращайтесь, пожалуйста, по телефону: **8-499-249-3138** или по электронной почте: marathon@1september.ru

# Россыпь “зацепок”

Попробую коротко объяснить, что за странные странички составляют почти треть этого номера “Информатики”. Всем нам наверняка знакома такая ситуация: увидели, услышали, где-то прочитали “что-то интересное”. Это “что-то” может быть почти чем угодно и не иметь прямого (или вообще никакого) отношения к нашему предмету. Но запущенные интересной зацепкой мысли начинают крутиться, и рождается красивая задачка, интересный пример, тема для детской исследовательской работы. Некоторое время назад в редакцию “Информатики” попала замечательная книжка “The Math Book: From Pythagoras to the 57th Dimension, 250 Milestones in the History of Mathematics”. Вся она состоит из коротких зарисовок, посвященных различным математическим фактам, — тех самых “зацепок”. Эта книга и вдохновила нас на подборку, которую мы предлагаем вашему вниманию. Конечно, подборка грешит некоторой “случайностью”. Степень интересности того или иного факта и степень его отношения именно к информатике — вещи очень субъективные. Но мы надеемся, что вам будет интересно. И какое-то количество этих заметок “зацепит” именно вас.

# Игра Вари

Возраст африканской настольной игры вари насчитывает 3500 лет. Она является национальной игрой Ганы и распространена в Западной Африке и на островах Карибского бассейна. Игра, в которой нужно просчитывать ходы с целью захватить камни противника, принадлежит к семейству игр манкала.

Доска для игры вари состоит из двух рядов, в каждом по 6 лунок, и меток (обычно это камни, зерна или бобы), по 4 в каждой лунке. Каждому игроку принадлежит один ряд из лунок, и оппоненты по очереди совершают ходы. В свой ход игрок выбирает лунку и раскладывает камни из нее по одному в лунки, следующие за выбранной, в направлении против часовой стрелки. Второй игрок выбирает одну из своих лунок и проделывает то же самое. Если при раскладывании камней последний камень кладется в лунку на стороне противника, в которой находилось 1 или 2 камня, тогда игрок забирает все камни из этой лунки себе, выводя их из игры. Если в предыдущей по направлению раскладывания лунке оказывается 2 или 3 камня, то их игрок тоже забирает себе, повторяя эту операцию до тех пор, пока не наткнется на свою лунку или лунку, содержащую 1, 4 или больше камней. Игра заканчивается, если у одного из игроков больше нет камней в лунках. Выигрывает тот, кто захватил больше камней.

Игра всегда представляла огромный интерес для исследователей в области искусственного интеллекта, которые разрабатывали алгоритмы решения различных головоломок и игровые стратегии, но до 2002 года никто и не догадывался, что вари относится к тому же классу игр, что и крестики-нолики, то есть к играм, в которых безошибочный игрок всегда может свести партию к ничьей. В конце концов ученые Джон Ромейн и Генри Бал из Амстердамского свободного университета написали программу, просчитавшую все 889 063 398 406 возможных позиций, появляющихся в игре, и доказали существование идеальной стратегии, неизменно приводящей к ничьей. Огромные вычисления заняли 51 час совместной работы 144 компьютеров.

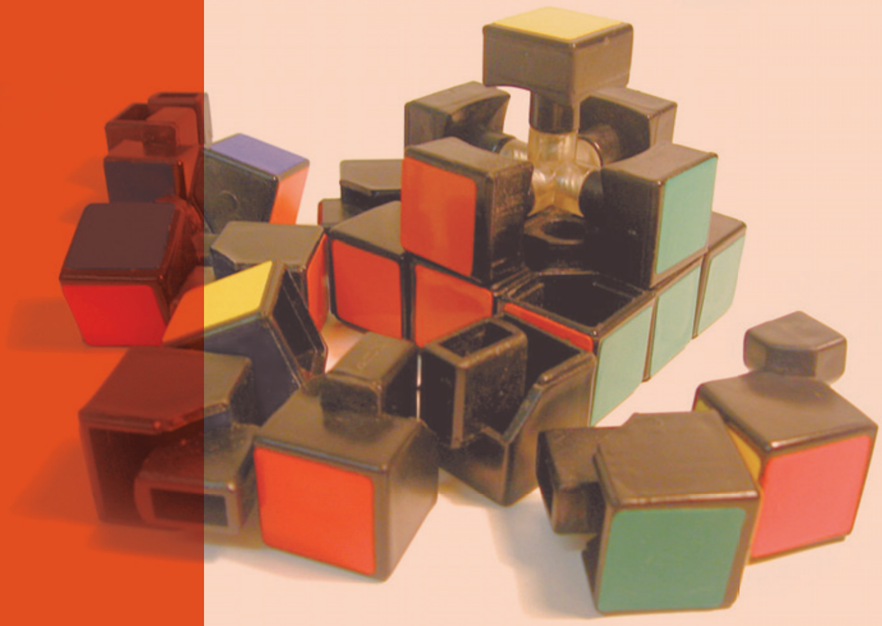
# Крестики-нолики

Крестики-нолики — одна из наиболее известных и древних игр. Несмотря на то что в современном виде крестики-нолики существуют сравнительно недавно, первые известные похожие игры появились около 1300 года до нашей эры в Древнем Египте. В крестиках-ноликах два игрока по очереди заполняют клетки таблицы размером 3 × 3 своими знаками (X у первого игрока и O у второго). Игрок, первым выстроивший три своих знака в ряд по вертикали, горизонтали или диагонали, выигрывает. На поле 3 × 3 любой из игроков всегда может довести партию до ничьей.

В Древнем Египте, во времена великих фараонов, настольные игры играли важную роль в повседневной жизни, поэтому до нас и дошли сведения о популярности игр, схожих с крестиками-ноликами. Эту игру можно рассматривать как атом, из которого впоследствии сложились более сложные настольные игры. При малейших изменениях и расширениях крестики-нолики превращаются в увлекательное приключение для игрока, и требуется немало времени, чтобы овладеть такой игрой в совершенстве.

Математики и любители головоломок много раз усовершенствовали игру, увеличивая поле, вводя новые размерности и необычные игровые поверхности, такие, как прямоугольное или квадратное поле со склеенными сторонами, образующее тор (имеющее форму бублика) или бутылку Клейна (поверхность только с одной стороной).

Рассмотрим любопытные факты, связанные с игрой в крестики-нолики. Игроки могут располагать свои знаки 9! = 362 880 способами. Существует 255 168 различных сценариев игры, которые длятся от 5 до 9 ходов. В начале 80-х годов компьютерные гении Дэнни Хиллис и Брайен Сильверман с друзьями собрали компьютер, играющий в крестики-нолики. Он был сделан из 10 000 деталей детского конструктора. В 1998 году ученые и студенты Университета Торонто создали робота, играющего с человеком в трехмерные крестики-нолики на поле 4 × 4 × 4.



# Кубик Рубика

Кубик Рубика был создан венгерским изобретателем Эрнё Рубиком в 1974 году, запатентован в 1975 году и в 1977 г. поступил в продажу в Венгрии. К 1982 году в Венгрии было продано более 10 миллионов кубиков, что превышает население этой страны. По оценкам, всего в мире продано более 100 миллионов кубиков.

Кубик представляет собой блок 3 × 3 × 3 из меньших по размеру кубиков, которые раскрашены так, что каждая из шести граней головоломки имеет свой цвет. 26 видимых кубиков закреплены так, чтобы грани самой игрушки могли вращаться. Цель игры — вернуть кубик Рубика из произвольного положения в начальное, в котором все квадраты одной грани имеют одинаковый цвет. Всего существует 43 252 003 274 489 856 000 различных положений 26 кубиков головоломки, и только в одном из этих положений каждая грань состоит из квадратов одного цвета. Если каждое возможное положение воспроизвести на отдельном кубике Рубика, то ими можно будет покрыть всю земную поверхность (включая океаны) около 250 раз. Столбец, состоящий из записей различных положений кубика, растянется на 250 световых лет. Если допустить возможность переклеивания цветных наклеек с одних квадратов на другие, то число положений увеличится до $1{,}0109 \times 10^{38}$.

Минимальное число ходов, за которое можно собрать кубик Рубика из произвольной позиции, все еще неизвестно. В 2008 году Томас Рокицки доказал, что для решения головоломки из любого начального положения игроку потребуется сделать не более 22 ходов.

Одной из вариаций кубика Рубика, которая никогда не появлялась на полках игрушечных магазинов, является четырехмерный кубик Рубика — Тессеракт Рубика. Число его различных положений составляет $1{,}76 \times 10^{120}$. Если бы куб или тессеракт каждую секунду меняли бы свои конфигурации на еще не показанные с самого первого мгновения существования нашей Вселенной, они бы до сих пор не перебрали все возможные варианты.

# NP-полнота игры тетрис

Тетрис, очень популярная видеоигра с падающими сверху фигурками, была изобретена в 1985 году русским компьютерным инженером Алексеем Пажитновым. В 2002 году американские ученые определили класс сложности игры и показали, что он имеет сходства со сложнейшими проблемами математики, которые требуют исчерпывающего анализа для нахождения решения и не решаются с помощью простых алгоритмов.

В тетрисе фигурки падают сверху на игровое поле и движутся к его нижнему краю. Во время того, как фигурка опускается, игрок может вращать ее и двигать по горизонтали. Форма фигурок, которые используются в игре, называется тетрамино, они состоят из четырех квадратов, соединенных вместе и образующих несложную геометрическую фигуру, например, букву Т. Когда падающая фигурка останавливается, натыкаясь на лежащие внизу фигурки, сверху начинает падать еще одна. Полностью заполнившиеся горизонтальные ряды пропадают, и ряды над ними падают на одну клетку вниз. Игра заканчивается, когда на поле недостаточно места для появления новой фигурки. Целью игрока является как можно более долгая игра и набор очков за заполненные ряды.

В 2002 году Эрик Дэмейн, Сьюзан Хохенбергер и Дэвид Либен-Ноэл исследовали обобщенную версию игры, в которой игровое поле может быть любых размеров. Они установили, что задача заполнения максимального числа рядов с помощью фигурок выбранной последовательности является NP-полной ("NP" обозначает "недетерминированной полиномиальной"). Хотя правильность решения таких задач можно проверить, само нахождение решения занимает огромное время. Классическим примером NP-полной проблемы является задача коммивояжера, для решения которой необходимо найти оптимальный маршрут, по которому коммивояжер должен двигаться, чтобы посетить все указанные города. Проблемы этого класса являются невероятно сложными, потому что не существует быстрого и рационального алгоритма их решения.

# Кипу

Древние инки использовали кипу, хранилище памяти из веревок и узлов, для записи чисел. До недавнего времени древнейшая из известных кипу была датирована примерно 650 годом нашей эры. Однако в 2005 году на побережье Перу в городе Караль была найдена кипу, возраст которой составляет пять тысяч лет.

Инки, жившие в Южной Америке, имели высокоразвитую цивилизацию с единой государственной религией и языком. Хоть у инков и не было письменности, они хранили обширные записи, с помощью специальной системы кодирования конвертируя их в нити, которых в одной кипу могло быть от трех до тысячи. К сожалению, испанцы, приплывшие в Южную Америку, увидев кипу, решили, что перед ними дело рук дьявола. Во имя Господа ими были уничтожены тысячи кипу, и к настоящему времени их уцелело лишь около 600.

Различные виды узлов и их расположение, направление нитей, расположение нитей на разных уровнях, использование цветов и интервалов — все это позволяло представлять числа. Группы узлов обозначали различные степени 10. С помощью узлов, возможно, фиксировали количество людских и материальных ресурсов, а также вели календарь. Кроме того, кипу могли содержать информацию о планах строительства, танцевальных движениях и даже о моментах инкской истории. Использование инками кипу развеивает представления о том, что математика могла развиваться только после появления письменности: общество может достигнуть высокого уровня развития и без письменного языка. Интересно, что в честь этого полезного изобретения древних сейчас названы менеджеры файлов в некоторых компьютерных системах.

У кипу было и жуткое предназначение. С помощью них планировали количество взрослых и детей, которых ежегодно закалывали, принося в жертву богам. Некоторые кипу описывали жертвоприношения целой империи, где нити обозначали дороги, а узлами обозначались жертвоприношения.

# Калькулятор Curta

Многими историками науки Curta считается первым коммерчески успешным портативным механическим арифмометром. Он был разработан австрийским евреем Куртом Херцштарком во время заключения в концентрационном лагере Бухенвальд. Curta, помещающаяся в руке, могла производить операции умножения, сложения, вычитания и деления. Арифмометр цилиндрической формы имел 8 ползунков для ввода данных, при использовании его обычно держали в левой руке.

В 1943 году Херцштарка обвинили в “пособничестве евреям” и “непристойных связях с арийскими женщинами”. В конце концов он оказался в Бухенвальде, где слухи о его увлечении техникой и идеях по созданию вычислительных машин быстро дошли до немцев. Ему была предоставлена возможность работать над чертежами своего арифмометра, так как начальство лагеря захотело преподнести готовое устройство нацистским лидерам в качестве подарка.

После войны, в 1946 году, князь Лихтенштейна посетил Херцштарка и предложил открыть фабрику по производству калькулятора Curta, первые экземпляры которого увидели свет в 1948 году. В течение долгого времени Curta был одним из лучших портативных вычислительных устройств и имел огромную популярность вплоть до расцвета электронных калькуляторов в 1970-х. Первая модель Curta I была 11-разрядной. Чуть большая по размерам Curta II, появившаяся в 1954 году, была уже 15-разрядной. В течение 20 с небольшим лет было собрано около 80 000 Curta I и 60 000 Curta II.

Астроном и писатель Клифф Столл пишет: “Иоганн Кеплер, Исаак Ньютон и лорд Кельвин жаловались, что они тратят много времени на проведение простых вычислений… Ох, как бы им пригодился карманный калькулятор, способный складывать, вычитать, умножать и делить числа! С выводом данных и памятью. Простой, удобный в обращении. Но их было не найти вплоть до 1947 года, когда на четверть века из Лихтенштейна к нам пришел лучший карманный арифмометр. В этой крохотной стране, налоговом раю, полном альпийских пейзажей, Курт Херцштарк создал самую оригинальную из всех машин, сделанных рукой инженера: калькулятор Curta”.

# HP-35: Первый карманный научный калькулятор

В 1972 году компания Hewlett-Packard (HP) с офисом в Пало-Альто, Калифорния, представила первый в мире карманный научный калькулятор — помещавшееся в руке устройство, работавшее в том числе с тригонометрическими и экспоненциальными функциями. Он имел широкий числовой диапазон, от $10^{-100}$ до $10^{100}$ в экспоненциальной записи, и продавался за 395 долларов США (в самой компании HP-35 называли “35”, так как он имел 35 кнопок).

Один из основателей компании Уильям Хьюлетт начал разрабатывать компактные калькуляторы даже несмотря на то, что исследования рынка в то время показывали отсутствие спроса на карманные вычислительные машины. Как же сильно они ошибались! За первые несколько месяцев количество заказов превысило ожидаемые размеры всего рынка карманных калькуляторов. В первый год было продано 100 000 HP-35, всего же до момента прекращения производства было продано еще более 300 000 калькуляторов.

На момент создания HP-35 ученые в своих вычислениях использовали логарифмические линейки. Существующие карманные вычислительные машины того времени умели складывать, вычитать, умножать и делить числа. HP-35 изменил все. Логарифмическая линейка, которая производила вычисления с точностью до трех десятичных знаков, “умерла” и затем лишь изредка использовалась в американских школах. Можно только догадываться, каких успехов достигли бы великие математики древности, будь у них доступ к HP-35 (с бесконечным запасом батарей).

Сегодня научные калькуляторы совсем недороги, и благодаря им сильно изменилось математическое образование в большинстве стран мира. Педагоги больше не учат, как на бумаге вычислять значения трансцендентных функций. В будущем преподаватели, возможно, будут уделять еще больше времени математическим приложениям и концепциям вместо рутинных вычислений.

Писатель Боб Льюис считал, что “Хьюлетт и Паккард основали Силиконовую долину в гараже Хьюлетта. Они решили, как назвать свою компанию, Hewlett-Packard или Packard-Hewlett, подбрасыванием монеты… Хьюлетт никогда не хотел стать знаменитым. В течение всей своей жизни в глубине души он оставался инженером”.

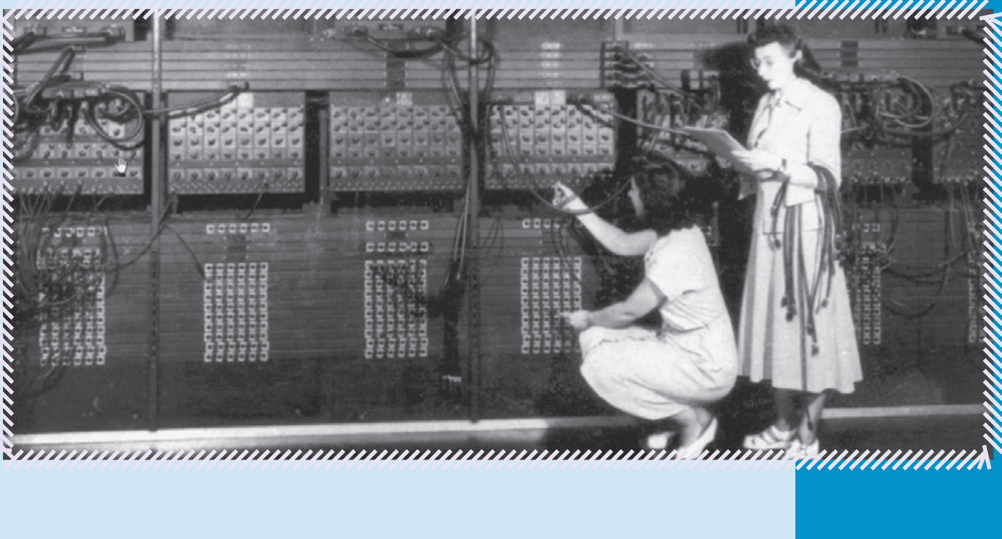
# Метод середины квадрата фон Неймана

Ученые используют генераторы случайных чисел для решения многих проблем, например, для шифрования сообщений, моделирования движения атомов и аккуратного проведения доказательств. Генератор псевдослучайных чисел (ГПСЧ) — это алгоритм, порождающий последовательность чисел, обладающих статистическими свойствами случайных чисел.

Метод середины квадрата, разработанный математиком Джоном фон Нейманом в 1946 году, является одним из первых и наиболее известных методов получения псевдослучайных чисел для компьютеров. Алгоритм стартует с произвольного числа, например, 1946, затем число возводится в квадрат (3 786 916). Его можно записать как 03 786 916. Теперь возьмем четыре средние цифры, 7869, и проделаем операции возведения в квадрат и выделения средних цифр уже с получившимся числом. На каждом шаге будем получать числа требуемой последовательности. На практике фон Нейман использовал десятизначные числа.

Фон Нейман, известный своими исследованиями в области термоядерных реакций, что в итоге привело к созданию термоядерной бомбы, понимал, что его простой алгоритм имеет изъяны, а последовательность с какого-то момента начнет повторяться, но он был доволен результатами применения своего метода. В 1951 году фон Нейман предупредил любителей подобных схем: "всякий, кто питает слабость к арифметическим методам получения случайных чисел, грешен вне всяких сомнений". Тем не менее он предпочитал свой подход более совершенным аппаратным генераторам случайных чисел, которые не записывали получившиеся последовательности, что делало трудным нахождение ошибок (было сложно воспроизвести алгоритм еще раз). К сожалению, и сам фон Нейман не имел доступа к компьютерам с достаточной памятью для хранения больших последовательностей "случайных" чисел. Но в любом случае ЭНИАК, действуя по этому замечательному своей простотой алгоритму, генерировал последовательности в сотни раз быстрее, чем другие подобные устройства.

Более поздние генераторы псевдослучайных чисел используют линейный конгруэнтный метод с соотношением вида $X_{n+1} = (aX_n + c) \bmod m$, где $n \geq 0$, $a$ — множитель, $m$ — модуль, $c$ — приращение, а $X_0$ — начальное значение. "Вихрь Мерсенна", алгоритм генерирования псевдослучайных чисел, разработанный в 1997 году Макото Мацумото и Такуджи Нисимурой, также часто применяется учеными.



*Так программировали на ЭНИАКе*

# Создание генератора случайных чисел

В современной науке генераторы случайных чисел используются для имитации природных явлений и составления выборок данных. До появления современных электронных компьютеров исследователям приходилось проявлять смекалку, чтобы получать случайные числа. Например, лорд Кельвин для этой цели использовал числа, взятые из головы и написанные на маленьких кусочках бумаги. Тем не менее он счел, что его метод не генерирует истинно случайные числа: “Даже после лучшего перемешивания, на которое только способен человек, шансы вытащить каждую из бумажек не становятся равными”.

В 1927 году британский статистик Леонард Типпетт проводил исследования, пользуясь таблицей из 41 600 случайных чисел, полученных путем взятия нескольких центральных цифр из записи координат расположения церковных приходов Англии. В 1938 году британские статистики Рональд Фишер и Фрэнк Йейтс опубликовали 15 000 случайных чисел, полученных с помощью двух колод карт и таблицы логарифмов.

В 1938-м и 1939 гг. британский статистик Морис Кендалл вместе с психологом Бернардом Бабингтоном Смитом начали искать способ генерировать случайные числа посредством машин. Их генератор случайных чисел стал первым подобным устройством, он сгенерировал таблицу из 100 000 случайных чисел. Кроме того, они сформулировали ряд тестов, чтобы определить, действительно ли получившиеся числа случайны. Числа, полученные Кендаллом и Смитом, были широко распространены до тех пор, пока корпорация RAND не выпустила в 1955 году “Миллион случайных чисел со 100 000 нормальных отклонений”. В RAND использовали похожую на рулетку машину, в чем-то сходную с машиной Кендалла и Смита, а затем проверяли числа на предмет их случайности с помощью простых математических тестов.

Кендалл и Смит использовали мотор, соединенный с кругом из картона, около 25 сантиметров в диаметре. Диск был разделен на 10 “как можно более равных” сегментов, пронумерованных от 0 до 9, и освещался неоновой лампой. Лампа периодически загоралась, и оператор записывал число, которое видел на крутящемся диске.

# Снежинка Коха

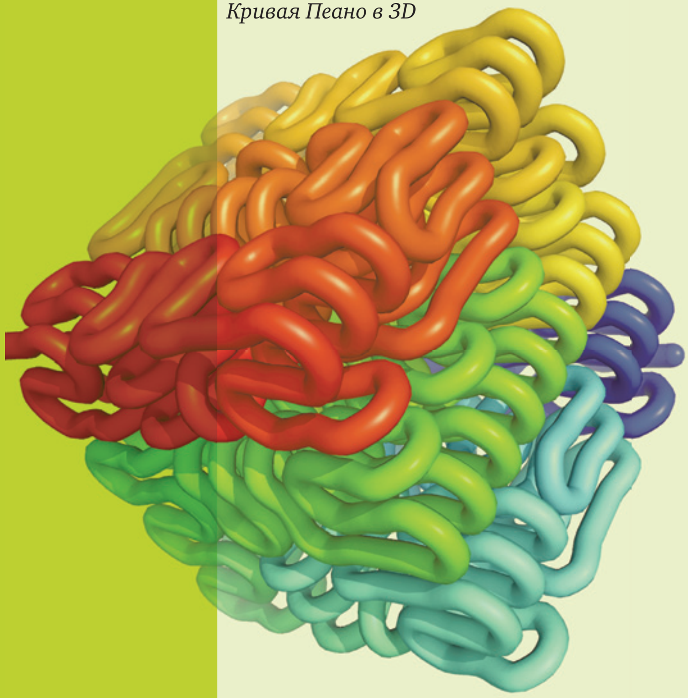
Снежинка Коха часто становится одним из первых фракталов, с которыми знакомятся студенты, кроме того, это один из первых фракталов в истории математики. Запутанная форма снежинки впервые появляется в 1904 году в работе шведского математика Хельге фон Коха “Об одной непрерывной кривой, не имеющей касательных”. Связанное понятие, кривая Коха, обозначает кривую, которая начинается не с равностороннего треугольника, как снежинка Коха, а с отрезка, и строится тем же способом.

Чтобы построить извивающуюся кривую Коха, мы берем рекурсивно уменьшающийся отрезок и наблюдаем за все возрастающим числом изломов. Возьмем отрезок и разобьем его на три равных части. Затем вместо центрального сегмента подставим два отрезка, каждый из которых эквивалентен отрезкам первого разбиения. Вместе эти два отрезка образуют клин (а вместе с замещенным отрезком они образуют равносторонний треугольник). Теперь наша кривая состоит из четырех отрезков. Для каждого из них проделаем описанную выше операцию разбиения на два отрезка и образования клина еще раз.

Начав с отрезка длиной 1 сантиметр, на $n$-м шаге мы получим кривую длины $(4/3)^n$ сантиметров. После нескольких сотен подобных операций длина кривой начинает превышать диаметр видимой части Вселенной. В конце получается кривая Коха бесконечной длины и фрактальной размерности 1,26, так как она частично покрывает двумерную плоскость, на которой она нарисована.

Но даже если длина кривой Коха и бесконечна, она покрывает область площадью всего $(2\sqrt{3}s^2)/5$, где $s$ — длина первого отрезка, что эквивалентно 8/5 площади начального треугольника. Заметим, что тангенс угла наклона касательной в точке сгиба не определен, это означает, что в точках сгиба данная функция не дифференцируема. На самом деле кривая Коха является нигде не дифференцируемой функцией, несмотря на то, что она непрерывна.

*Снежинка Коха в 3D*



*Кривая Пеано в 3D*

# Кривая Пеано

В 1890 году итальянский математик Джузеппе Пеано представил публике один из первых примеров заполняющей пространство кривой. Британский ученый Дэвид Дарлинг назвал это открытие “землетрясением в традиционной структуре математики”. А русский математик Наум Виленкин при описании этого нового класса кривых написал: “Все было в руинах, и все математические понятия внезапно утратили свое значение”.

Термин “кривая Пеано” является синонимом кривой, заполняющей пространство, и строятся подобные кривые с помощью повторяющегося процесса, который создает зигзагообразную линию, покрывающую все пространство. По словам Мартина Гарднера, “кривые Пеано оказались глубоким потрясением для математиков. Их пути кажутся одномерными, тогда как они могут покрывать и двумерные области. Стоит ли называть их кривыми? Что еще хуже, несложно нарисовать кривые Пеано, которые могли бы покрывать кубы и гиперкубы...”. Кривые Пеано непрерывны и так же, как снежинка Коха или функция Вейерштрасса, ни в какой точке не дифференцируемы. Заполняющие пространство кривые имеют хаусдорфову размерность 2.

Заполняющие пространство кривые имеют множество практических применений, с их помощью, например, можно проложить наиболее удобный маршрут для посещения большого количества городов. Так Джон Бартольди III, профессор Школы индустриальной и системной инженерии в Технологическом институте Джорджии, использовал кривые Пеано, чтобы построить систему доставки сотен завтраков для бедных и систему доставки донорской крови в различные госпитали для американского Красного Креста. Так как конечные пункты доставки сгруппированы вокруг городов, применение кривых Пеано дает хорошие результаты, ведь кривая сначала покрывает одну из областей на карте и только потом переходит к другой. Ученые также экспериментировали с применением заполняющих пространство кривых для назначения целей различных видов вооружения. Это стало возможным благодаря тому, что компьютер, запущенный на орбиту Земли, может очень эффективно использовать кривые Пеано.

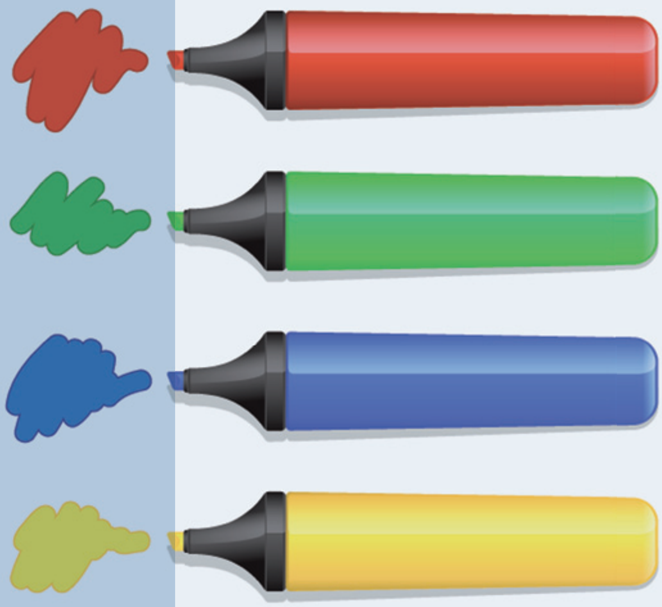
# Криптографические системы с открытым ключом

На всем протяжении истории криптографы искали способы шифрования секретных сообщений без использования громоздких шифровальных книг с записанными в них ключами, которые могли легко попасть в руки врага. Например, германская армия в период с 1914-го по 1918 год по разным причинам недосчиталась четырех шифровальных книг, которые попали в руки британских разведчиков. Подразделение британских криптографов, известное как “Комната 40”, сумело расшифровать немецкие сообщения, что дало союзникам решающий стратегический перевес в ходе Первой мировой войны.

Работая над решением проблемы управления ключами, в 1976 году Уитфилд Баффи, Мартин Хеллман и Ральф Меркл из Стэнфордского университета в Калифорнии разработали криптографические системы с открытым ключом, математический метод передачи шифрованных сообщений с использованием пары криптографических ключей: открытого и секретного. Секретный ключ должен сохраняться в тайне, а открытый, что удивительно, может быть выставлен на всеобщее обозрение без потери безопасности. Ключи связаны между собой, но секретный ключ не может быть получен из открытого никакими способами. Передаваемое сообщение шифруется с помощью открытого ключа, а расшифровано может быть только с помощью секретного.

Чтобы лучше понять системы с открытыми ключами, представим почтовый ящик, висящий на входной двери дома. Любой человек с улицы может положить в него письмо, а адрес здесь выступает в роли открытого ключа. Тем не менее только человек, владеющий ключом от двери дома, может извлечь и прочитать письмо.

В 1977 году ученые Массачусетсского технологического института Рональд Ривест, Ади Шамир и Леонард Адлеман предложили использовать большие простые числа для защиты информации. Компьютер может быстро перемножить два простых числа, но обратный процесс разложения на простые множители может оказаться очень сложным. Стоит отметить, что компьютерные ученые использовали шифрование с помощью открытых ключей и раньше, но так как эти работы проводились для британских разведывательных служб, их сохраняли в тайне для обеспечения национальной безопасности.



# Проблема четырех красок

В течение многих веков картографы верили, что для раскраски карты так, чтобы никакие две области, имеющие общую границу, не были бы одного цвета, достаточно четырех цветов. При этом допускалась ситуация, когда две области одного цвета имели общую вершину. Сегодня мы знаем наверняка, что хотя некоторые карты можно раскрасить и меньшим числом цветов, но никакая карта на плоскости не требует больше четырех. Четырех цветов также достаточно для раскраски карт, нанесенных на сферу и цилиндр. Семью цветами можно раскрасить карту на торе (поверхности, имеющей форму бублика).

В 1852 году математик и ботаник Френсис Гатри первым доказал достаточность четырех цветов для раскрашивания карты графств Англии. После него многие математики безуспешно пытались доказать это кажущееся простым утверждение в общем случае, и со временем эта проблема стала одной из наиболее известных нерешенных задач топологии.

Наконец, в 1976 году математики Кеннет Аппель и Вольфганг Хакен доказали теорему о четырех цветах, прибегнув к помощи компьютера, исследовавшего тысячи частных примеров. Это был первый случай, когда использование компьютерных вычислений сыграло важную роль в доказательстве математической теоремы. Сегодня роль компьютеров в математике сильно возросла, и подчас результаты их работы бросают вызов пониманию человека. Одним из примеров может служить проблема четырех красок. Другим стала классификация простых конечных групп, воплощенная в 10 000 страниц, написанных многими авторами. К сожалению, традиционные методы проверки корректности доказательства самими учеными неприменимы к работам, которые исчисляются тысячами страниц.

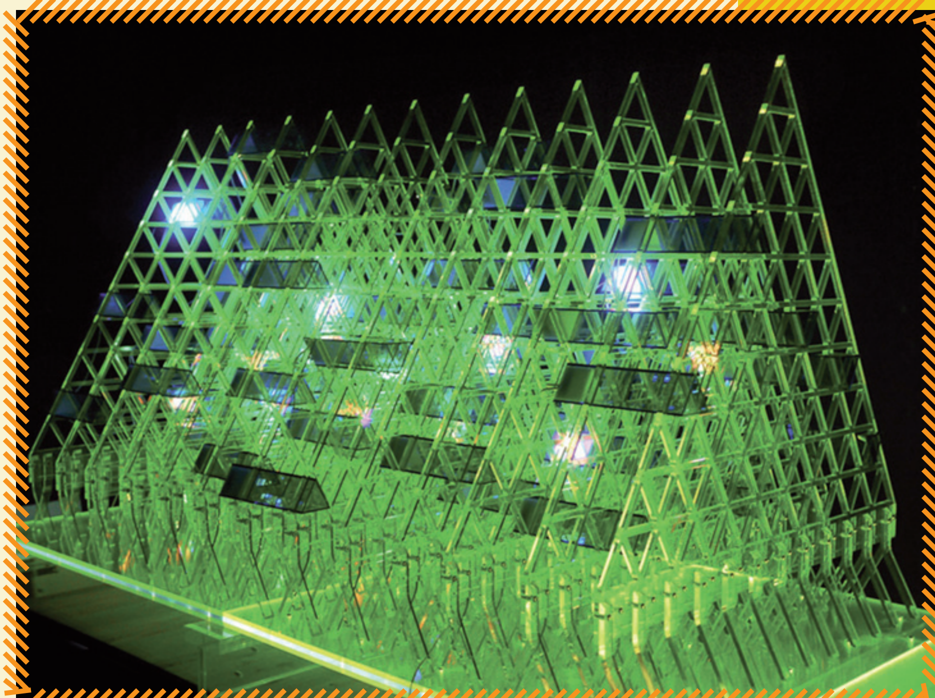
Удивительно, но решение проблемы четырех красок не оказало большого влияния на работу картографов. Со временем у них пропало желание минимизировать количество цветов на карте, и современные книги зачастую используют больше цветов, чем необходимо.

# Треугольник Паскаля

Треугольник Паскаля является одной из наиболее известных таблиц в истории математики. Блез Паскаль был первым, кто глубоко изучил эту последовательность, опубликовав свой трактат в 1654 году, хотя задолго до него, еще в XII веке, данная модель была известна персидскому поэту и математику Омару Хайяму, а первые упоминания о ней можно найти в еще более древних трактатах, написанных индийскими и китайскими математиками.

Каждое число треугольника является суммой двух чисел, располагающихся над ним. Математики в течение многих лет обсуждали роль треугольника Паскаля в теории вероятности, в разложении $(x + y)^n$ и в различных приложениях теории чисел. Математик Дональд Кнут (родился в 1938 г.) однажды заметил, что в треугольнике Паскаля заложено столько последовательностей и закономерностей, что когда в нем находят нечто новое, этому уже никто, кроме самого исследователя, не удивляется. Тем не менее увлекательные исследования обнаружили огромное количество чудес, заключенных в треугольнике, таких, как необычные геометрические закономерности на диагоналях, существование квадратных узоров, расширения треугольников и их связь с отрицательными целыми числами и большими размерностями.

Если заменить четные числа в треугольнике на точки, а вместо нечетных оставить пустые места, то получится фрактал с замысловатыми повторяющимися фигурами при разных масштабах изображения. Такие фракталы могут иметь практическое значение в качестве базовых моделей для создания новых структур с необычными свойствами. Например, в 1986 году ученые создали уплотнитель для проводов микронного размера со структурой, практически идентичной треугольнику Паскаля с пустотами на месте нечетных чисел. Площадь самого маленького треугольника составила ничтожные 1,38 микрон, и учеными было исследовано немало необычных свойств уплотнителя в магнитном поле.



*Инсталляция по мотивам треугольника Паскаля*


# Формула Стирлинга

Сегодня факториалы используются в математике повсеместно. Для неотрицательного целого числа $n$ выражение “$n$ факториал” (записывается как $n!$) означает результат произведения всех целых чисел от 1 до $n$ включительно. Для примера $4! = 1 \cdot 2 \cdot 3 \cdot 4 = 24$. Обозначение $n!$ было предложено французским математиком Кристианом Крампом в 1808 году. Факториалы играют важную роль в комбинаторике, например, при подсчете количества различных вариантов расположения элементов в последовательности. Они также возникают в теории чисел, исчислении и теории вероятности.

Из-за того, что факториал является очень быстро растущей функцией (например, 70! больше $10^{100}$, а 25 206! больше, чем $10^{100\,000}$), удобные способы приближения значений факториалов больших чисел могут быть крайне полезны. Формула Стирлинга, $n! \approx (2\pi)^{1/2}e^{-n}n^{n+1/2}$, дает аккуратную оценку числу $n!$. Здесь символ “$\approx$” означает “приблизительно равен”, $e$ и $\pi$ — математические константы, $e \approx 2{,}71828$, а $\pi \approx 3{,}14159$. Для больших значений $n$ данное выражение принимает еще более простой вид: $\ln(n!) \approx n\ln(n) - n$, что можно по-другому записать как $n! \approx n^n e^{-n}$.

В 1730 году шотландский математик Джеймс Стирлинг представил свой метод приближения значений функции $n!$ в работе “Methodus Differentialis”. Стирлинг начинал свою карьеру математика на фоне политических и религиозных конфликтов с властями. Он был дружен с Ньютоном, однако провел большую часть своей жизни после 1735 года, управляя промышленным предприятием.

Кейт Болл пишет: “По моему мнению, это одно из ключевых открытий математики XVIII века. Формула Стирлинга была одним из определяющих факторов в удивительном превращении, которое математика претерпела в XVII–XVIII веках. Логарифмы не были открыты до 1600 года. Ньютоновские “Начала”, заложившие основы исчисления, появились спустя 90 лет. В течение следующих 90 лет появились формулы, подобные формуле Стирлинга, утонченность которых была невозможна без формализации исчисления. Математика перестала быть развлечением для любителей и стала работой для профессионалов”.

# Парадокс брадобрея

В 1901 году британский философ и математик Бертран Рассел открыл парадокс или очевидное противоречие, которое привело к необходимости изменения теории множеств. Один из вариантов парадокса Рассела, также известного как парадокс брадобрея, гласит, что существует город, в котором есть только один брадобрей-мужчина, и каждое утро он бреет всех мужчин, которые не бреются сами, и больше никого. Бреет ли он себя?

Условие, казалось бы, требует, чтобы брадобрей брил себя только в том случае, если он сам себя не бреет! Хелен Джойс пишет, что “парадокс открывает нам пугающую возможность того, что вся математика стоит на очень шатком фундаменте, и ни одно утверждение не может быть истинным”.

Парадокс Рассела, в своем первоначальном виде, говорит о множестве всех множеств, которые не содержат себя в качестве своего элемента. Существует немало множеств *R*, которые не содержат себя в качестве своего элемента, — так, например, множество кубов вовсе не куб. Примерами множеств *T*, которые являются собственными членами, может быть множество всех множеств или множество всех вещей, кроме кубов. Нам кажется, что каждое множество принадлежит лишь одному из этих двух типов. Однако Рассел взял множество *S* всех множеств, которые не содержат себя в качестве своего элемента. С одной стороны, *S* содержит само себя в качестве элемента, а с другой — не может быть элементом *S*. Рассел пришел к выводу, что теория множеств нуждается в изменении, чтобы избежать в будущем подобной путаницы и возможных противоречий.

Одним из возможных опровержений парадокса брадобрея может быть утверждение, что подобного брадобрея попросту не существует. Однако парадокс Рассела привел к усовершенствованию теории множеств. Немецкий математик Курт Гедель использовал подобные наблюдения при создании своей теоремы о неполноте. Британский математик Алан Тьюринг также использовал результаты Рассела в доказательстве неразрешимости проблемы останова, которая заключается в оценке того, завершится ли работа компьютерной программы за конечное число шагов.

# Теорема о печатающих обезьянах

Теорема о печатающих обезьянах утверждает, что обезьяна, случайным образом нажимающая на клавиши пишущей машинки в течение бесконечного времени, почти наверняка напечатает заданный наперед текст, например, Библию. Рассмотрим произвольную библейскую фразу “В начале сотворил Бог небо и землю”. Сколько времени займет у обезьяны напечатать такой текст? Пусть наша машинка содержит 45 клавиш. Фраза состоит из 35 символов (включая пробелы и знаки препинания). Если вероятность нажатия нужной клавиши на машинке равняется $1/n$, где $n$ — число клавиш, тогда вероятность напечатать последовательно 35 необходимых знаков заданной фразы равняется $1/45^{35}$, а следовательно, обезьяне в среднем потребуется совершить больше $10^{56}$ попыток, чтобы добиться успеха. Если бы обезьяна печатала со скоростью 1 знак в секунду, то ей пришлось бы работать больше времени, чем существует наша Вселенная.

Что интересно, если бы мы могли сохранять только правильно набранные символы, обезьяна справилась бы с работой намного быстрее. Математический анализ показывает, что обезьяна, совершив всего 407 попыток, уже более чем с 50-процентной вероятностью напечатает необходимый текст! Этот опыт можно считать иллюстрацией того, как эволюция достигает удивительных результатов, сохраняя полезные функции и отказываясь от ненужных.

Французский математик Эмиль Борель упоминал “дактилоскопических” (печатающих) обезьян в своей статье 1913 года, в которой он доказывает теоретическую возможность создания книг миллионом обезьян, печатающих по 10 часов в день. В 1928 году физик Артур Эддингтон написал: “Если армия обезьян примется печатать на пишущих машинках, то они смогут напечатать все книги, которые есть в Британском музее. Их шансы явно предпочтительнее, чем шансы всех молекул газа в сосуде оказаться лишь в одной его половине”.

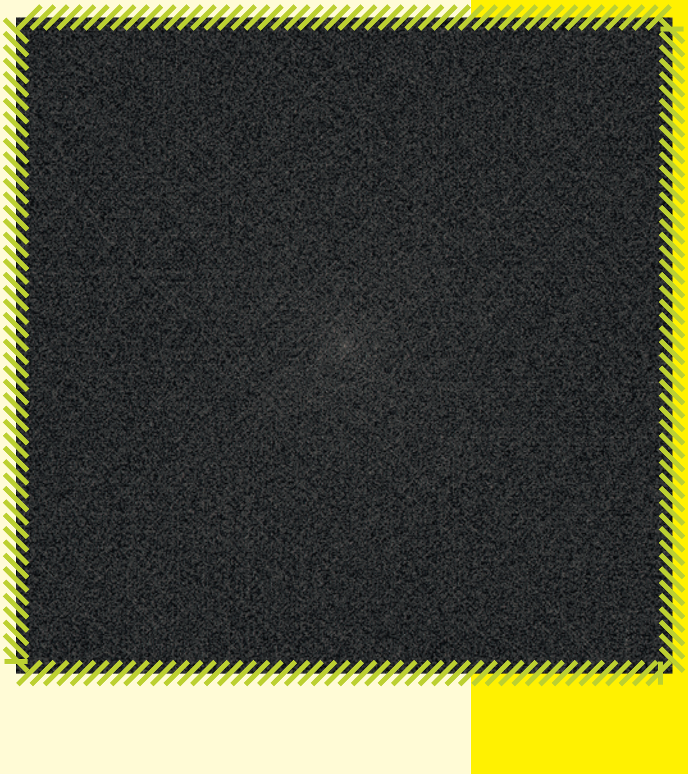
# Скатерть Улама

В 1963 году американский математик польского происхождения Станислав Улам, машинально рисуя на листке бумаги во время скучного доклада, случайно открыл удивительную спираль, которая выявляет закономерность в последовательности простых чисел. Начиная с единицы в центре, Улам по спирали, раскручивающейся против часовой стрелки, последовательно записывал натуральные числа. Затем он обвел все простые числа и заметил, что они выстраиваются вдоль диагональных прямых.

Позже компьютерное моделирование подтвердило, что хотя некоторые диагональные структуры могут получаться и из четных или нечетных чисел, но у простых чисел эта тенденция прослеживается более четко. Что еще более важно, работа Улама выдвинула на первый план компьютер, который, как оказалось, может быть использован в качестве своеобразного микроскопа, позволяющего математикам визуализировать структуры, лежащие в основе новых открытий. Это исследование начала 60-х годов постепенно привело к настоящему буму в экспериментальной математике конца XX века.

Мартин Гарднер писал: “Решетки на спирали Улама добавили каплю воображения в споры о загадочной смеси порядка и хаоса, которую можно наблюдать в распределении простых чисел… Машинальные рисунки в сумеречной зоне математики были восприняты Уламом со всей серьезностью. Он сделал предположение, которое впоследствии привело его и Эдварда Теллера к мысли о возможном создании водородной бомбы”.

Помимо своего вклада в разработку ядерного оружия во время Второй мировой войны и работы над Манхэттенским проектом, Улам также известен своей работой над двигателями космических аппаратов. Вместе с братом он уехал из Польши накануне Второй мировой войны, однако вся его семья осталась на родине и погибла во время Холокоста.



# Гугол

Термин “гугол”, обозначающий число, записываемое как единица со ста нулями, был придуман девятилетним Милтоном Сироттой. Милтон и его брат Эдвин большую часть своей жизни проработали на фабрике своего отца в Бруклине, Нью-Йорк, размельчая абрикосовые косточки, которые использовались в создании абразивных материалов для нужд промышленности. Сиротта был племянником американского математика Эдварда Казнера, популяризовавшего данный термин, которым Милтон решил обозначать очень большое число. Слово “гугол” впервые появляется в печатной работе в 1938 году.

Казнер известен как первый еврей, занявший должность профессорско-преподавательского состава в Колумбийском университете, и как соавтор книги “Математика и Воображение”, в которой он представил гугол широкой аудитории. Хоть гугол и не имеет особого значения в математике, он удобен для сравнения больших величин, а также внушает людям благоговение перед чудесами математики и огромной Вселенной, в которой мы живем. Но гугол изменил мир не только этим. Ларри Пейдж, один из основателей поисковой системы Google, назвал свою компанию в честь гугола, намеренно исказив термин.

Существует чуть больше, чем гугол различных способов разместить 70 предметов в определенной последовательности, например, построить 70 человек в очередь. Большинство ученых полагает, что если посчитать количество атомов во всех видимых нами звездах, то полученное число будет намного меньше гугола. За гугол лет успеют испариться все черные дыры нашей Вселенной. Однако число всевозможных шахматных партий больше, чем гугол. Термин “гуголплекс” обозначает число, записывающееся как единица с гугол нулями. В его записи больше знаков, чем атомов во всех видимых нами звездах.

# Философия и Забава в Алгебре

Мэри Эверест Буль была математиком, обучавшимся самостоятельно, она стала известной благодаря книге “Философия и Забава в Алгебре”, изданной в 1909 году. Она была женой Джорджа Буля, британского математика и философа, который изобрел булеву алгебру, заложившую основу современных компьютерных вычислений. Мэри Буль также известна тем, что редактировала монументальный труд своего мужа “Исследование законов мышления”, изданный в 1854 году. Ее “Философия и Забава в Алгебре” дает представление о состоянии математического образования в начале XX века.

Мэри какое-то время работала в Королевском колледже Лондона, первом женском колледже Англии. Она жила в эпоху, когда женщина, к сожалению, не могла претендовать на получение ученой степени или преподавать в колледже. Несмотря на большое желание преподавать, она смогла получить работу лишь в библиотеке, где помогла многим студентам. Ее настойчивость и рвение в изучении математики и в самообразовании сделали ее кумиром в глазах современных феминисток.

В конце своей книги Мэри Буль обсуждает комплексные числа, например, $\sqrt{-1}$, к которым она относилась с мистическим благоговением: “[Сильнейшие математики Кембриджа] думают о квадратном корне из минус единицы так, как будто он реален до тех пор, пока не теряют сон и не представляют, будто бы они и есть квадратный корень из минус единицы, который не может извлечь сам себя, и становятся они так больны, что вообще не могут идти на экзамен”. Также она писала, что “Ангелы и квадратные корни из отрицательных чисел есть посланники от Еще-Не-Известного; и приходят они, чтобы сказать, куда нам идти; и указать кратчайшую дорогу туда; туда, куда мы не можем прийти прямо сейчас”.

Математика была в крови у представителей семейства Буль. Старшая дочь Мэри вышла замуж за Чарльза Говарда Хинтона (1853–1907), который придавал мистическое значение тессерактам и способам визуализации четырехмерных изображений. Еще одна дочь, Алисия, стала известной благодаря своим работам о политопах, которые являются обобщением полигонов для больших размерностей. Этот термин был придуман ею самой.

*Королевский колледж Лондона*

# Ханойская башня

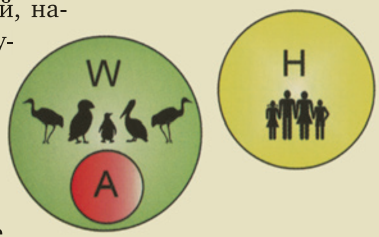
Ханойская башня была интригующей математической задачей для множества людей с самого момента своего создания французским математиком Эдуардом Люка в 1883 году, когда она была продана как игрушка. Этот математический паззл состоит из нескольких колец разного размера, которые можно нанизывать на три стержня. Первоначально все кольца нанизаны на один стержень по порядку, так что меньшее кольцо лежит на самом верху. Игрок берет одно верхнее кольцо и кладет его на другой стержень, но при этом нельзя класть большее по размеру кольцо на меньшее. Цель игры — переместить все кольца (часто их бывает 8) на другой стержень так, чтобы они снова расположились в порядке по убыванию. Минимальное количество ходов, необходимых для завершения игры, равняется $2^n - 1$, где $n$ — количество колец.

Считается, что создатели оригинальной игры вдохновились индийской легендой о храме города Бенарас, в котором на трех стержнях располагается 64 диска. Монахи, провинившиеся перед Брахмой, в наказание постоянно перемещают эти диски, причем по тем же правилам, что и в игре Ханойская башня. По преданию, когда последним движением паззл соберется до конца, наступит конец света. Правда, даже если монахи будут передвигать по одному кольцу в секунду, им потребуется совершить $2^{64} - 1$, или 18 446 744 073 709 551 615 перестановок, это займет около 585 миллиардов лет, что во много раз больше современных оценок продолжительности жизни нашей Вселенной.

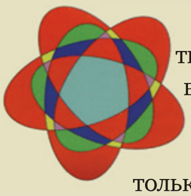
Для решения задачи с тремя стержнями существуют достаточно простые алгоритмы, и эта игра часто используется как пособие по изучению рекурсивных алгоритмов в программировании. Однако оптимальный алгоритм для решения задачи Ханойской башни для четырех и более стержней не найден. Математики сильно заинтересованы в нахождении решения этой задачи во многом потому, что она имеет множество приложений в различных областях математики, таких, как коды Грея и поиск гамильтоновых путей на $n$-мерном гиперкубе.

# Диаграммы Венна

В 1880 году Джон Венн, английский философ и священник англиканской церкви, разработал схему для визуализации элементов, множеств и логических отношений. Диаграммы Венна обычно состоят из круговых областей, обозначающих группы объектов, объединенных общими свойствами. Например, во Вселенной, населенной реальными и мифическими существами, область H обозначает людей, область W — всех крылатых существ, а область A обозначает ангелов. Анализ диаграммы показывает, что (1) Все ангелы наделены крыльями (область A полностью лежит в области W); (2) Все люди не имеют крыльев (области H и W не пересекаются) и (3) Люди не являются ангелами (области H и A также не пересекаются).



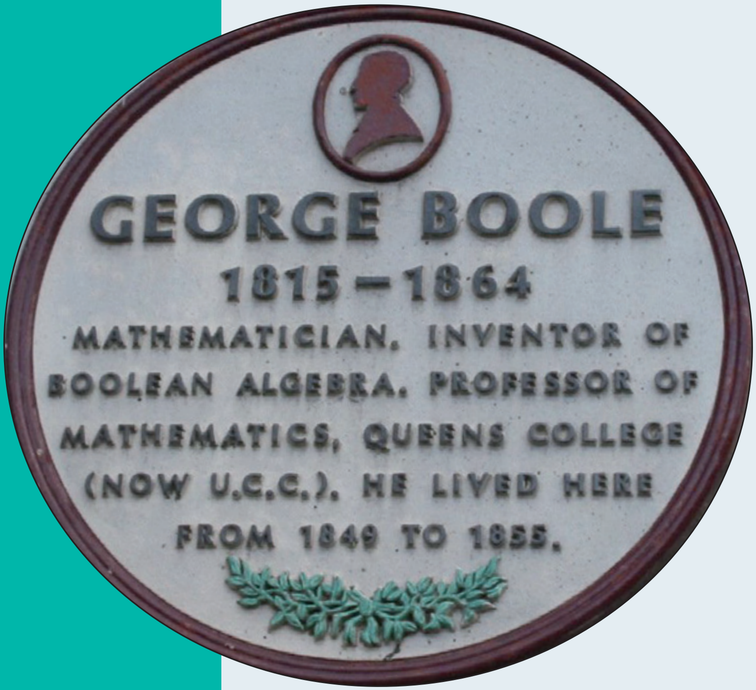

Здесь описано одно из основных правил логики, а именно, что из утверждений “A содержится в W” и “H не пересекается с W” следует, что “H не пересекается с A”. Это утверждение становится очевидным при взгляде на диаграмму.



Подобные диаграммы использовались и до Венна, например, математиками Готфридом Лейбницем и Леонардом Эйлером, но Венн первым всесторонне изучил, формализовал и обобщил их. На самом деле целью Венна было обобщение симметричных диаграмм на случай большого количества пересекающихся множеств, но он добился изображения только четырех множеств с помощью эллипсов.

Спустя век Бранко Грюнбаум, математик из Вашингтонского университета, создал симметричную относительно вращений диаграмму Венна, состоящую из пяти конгруэнтных эллипсов. На второй картинке показана одна из множества ее вариаций.

Математики со временем поняли, что симметричные относительно вращений диаграммы должны содержать простое число лепестков. Тем не менее создание диаграммы с семью лепестками оказалось настолько сложной задачей, что ученые уже начали было сомневаться в существовании подобных диаграмм. В 2001 году математик Петер Гамбурге и художник Эдит Хепп сконструировали диаграмму с 11 лепестками.




# Булева алгебра

Наиболее известной работой английского математика Джорджа Буля было “Исследование законов мышления, на которых основываются математические теории логики и вероятностей”. Буль хотел свести логику к простой алгебре, содержащей всего две величины, 0 и 1, и три основные операции: *и*, *или*, *не*. Сегодня булева алгебра широко используется в телефонных сетях и устройстве современных компьютеров. Буль считал, что его работа представляет собой “наиболее значимый вклад, который я внес или, вероятно, внесу в науку, и та вещь, благодаря которой я, возможно, останусь в памяти потомков”.

Джордж Буль умер в возрасте 49 лет от тяжелой лихорадки. Его жена была убеждена, что подобное лечат подобным, и выливала на мужа ведра воды, пока он лежал в постели, так как заболел он под холодным дождем.

Математик Огастес де Морган (1806–1871) восторгался работой Буля, написав, что “булева логическая система является одним из примеров соединения гения и упорства в одной работе… До того, как это было доказано, казалось невозможным, что, используя записанные при помощи символов алгебраические процессы, изобретенные для проведения вычислений, можно выразить любую мысль, а также снабдить всеобъемлющую логическую систему грамматикой и словарем”.

Примерно через 70 лет после смерти Буля американский математик Клод Шеннон (1916–2001), будучи еще студентом, ознакомился с его трудами и показал, как булеву алгебру можно применять в создании телефонных сетей. Он также доказал, что с помощью цепей и реле возможно решить проблемы булевой алгебры. Таким образом, Джордж Буль с помощью Шеннона заложил основы нынешнего Цифрового века.

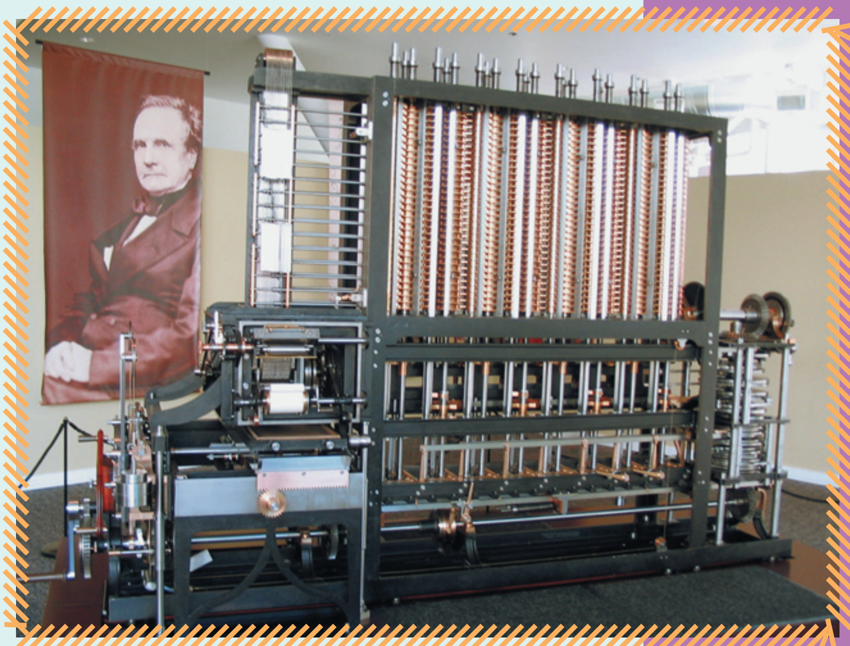
# Механический компьютер Бэббиджа

Чарльз Бэббидж — английский аналитик, статистик и изобретатель, который также интересовался религиозными чудесами. Однажды он написал: “Чудеса не нарушают установленных нами законов, но… указывают на наличие намного более высоких законов”. Бэббидж утверждал, что чудеса могут случаться и в механистическом мире. Так же как Бэббидж замечал отклонения в поведении своих вычислительных машин, так и Бог мог создать нарушения в природе. Исследуя библейские чудеса, он предположил, что шансы человека восстать из мертвых приблизительно равны $1/10^{12}$.

Бэббидж, как никто другой, внес огромный вклад в появление первых компьютеров. В частности, он известен созданием механического калькулятора огромных размеров, предшественника современных компьютеров. Бэббидж полагал, что его устройство окажется полезным для создания математических таблиц, и больше беспокоился об ошибках, допускаемых людьми при интерпретации результатов работы машины, которые та выдавала при помощи 31 металлического колеса с цифрами. Сегодня мы понимаем, что идеи Бэббиджа на сотню лет опережали его время, и уровень развития технологий не позволял ему воплотить в жизнь свои возвышенные мечты.

Разностная машина Чарльза Бэббиджа, к строительству которой он приступил в 1822 году, так его и не закончив, должна была вычислять значения полиномиальных функций и состоять из 25 000 частей. В планах Бэббиджа было также создание Аналитической машины, которая по замыслу должна была не только производить вычисления, но и с помощью перфокарт хранить данные в памяти. По его расчетам, памяти машины хватило бы на 1000 50-значных чисел, а сама машина получилась бы 30 метров в длину. Ада Лавлейс, дочь английского поэта лорда Байрона, написала программу для Аналитической машины. Несмотря на то что Бэббидж ей помогал, многие считают именно Аду первым компьютерным программистом.

В 1990 году писатели Уильям Гибсон и Брюс Стерлинг написали “Разностную машину”, в которой предлагали читателям представить, как бы изменился мир, если бы Разностная машина была создана уже в Викторианскую эпоху.



*Один из первых планов Кёнигсберга*


# Задача о кёнигсбергских мостах

Теория графов — это раздел математики, изучающий связи между объектами, и часто для упрощения проблемы можно представить ее в виде графа, представляющего собой совокупность вершин и соединяющих их линий. Проблема семи кёнигсбергских мостов в Германии (в настоящее время это территория России) — одна из старейших проблем теории графов. Жители старого Кёнигсберга любили совершать прогулки вдоль реки, иногда переходя ее по мостам и заходя на два небольших острова. В начале XVIII века люди все еще задавались вопросом, возможно ли пройти по всем семи мостам и вернуться на исходную позицию, не проходя по одному и тому же мосту дважды. Наконец, в 1736 году швейцарский математик Леонард Эйлер доказал, что такого пути не существует.

Эйлер представил мосты в виде графа, где различные берега и острова обозначались точками, а мосты — линиями. Он показал, что обойти такой граф, проходя по каждой линии лишь по одному разу, можно, но только если в графе меньше трех вершин нечетной степени (степень вершины равняется числу линий, начинающихся или заканчивающихся в этой точке). Система мостов Кёнигсберга не удовлетворяла этим критериям. Таким образом, невозможно обойти данный граф, не пройдя по одной из линий дважды. Эйлер впоследствии обобщил свои результаты на более общий случай произвольной системы мостов.

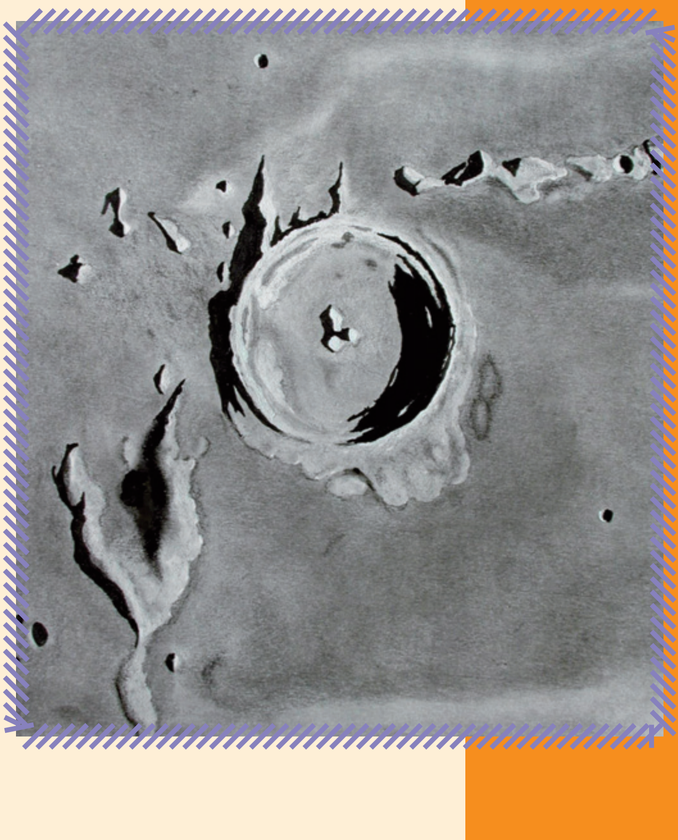
Задача о кёнигсбергских мостах занимает важное место в истории математики, а решение Эйлера представляет собой первую теорему теории графов. Сегодня теория графов имеет множество применений — от изучения метаболических путей и транспортного потока до анализа социальных сетей в Интернете. Теория графов даже может объяснить, как распространяются заболевания, передающиеся половым путем. Несложное представление мостов в виде графа без учета длины путей стало предшественником топологии — области математики, работающей с формами объектов и связями между ними.

# Решето Эратосфена

Простое число — это число больше единицы, например, 5 или 13, такое, что оно делится нацело только на себя и на 1. Число 14 не является простым, потому что оно представимо в виде произведения: 14 = 7 · 2. Простые числа очаровывают математиков вот уже более двух тысяч лет. Около 300 года до нашей эры Евклид доказал, что не существует наибольшего простого числа, и их количество бесконечно. Но как мы можем определить, что данное число простое? Около 240 года до нашей эры греческий математик Эратосфен разработал первый известный алгоритм нахождения простых чисел, который сегодня называют решетом Эратосфена. В частности, с помощью решета можно найти все простые числа вплоть до указанного целого числа. (Будучи универсальным ученым, он заведовал знаменитой Александрийской библиотекой и также был первым, кому удалось достаточно точно рассчитать диаметр Земли.)

Французский теолог и математик Марен Мерсенн (1588–1648) также был увлечен загадкой простых чисел и пытался вывести формулу, с помощью которой можно было бы находить все простые числа. Несмотря на то что вывести формулу он не сумел, его работы о числах Мерсенна вида $2^p - 1$, где $p$ — целое число, до сих пор представляют интерес для ученых. Числа Мерсенна, где $p$ — простое число, являются числами, которые несложно исследовать на простоту, поэтому они являются наибольшими известными простыми числами. Сорок пятое известное простое число Мерсенна ($2^{43\,112\,609} - 1$) было открыто в 2008 году и содержит 12 978 189 знаков!

Сегодня простые числа играют важную роль в криптографических системах с открытым ключом, которые позволяют нам посылать защищенные сообщения. Что более важно для математиков, простые числа служат ключом к решению множества нерешенных проблем математики, таких, как гипотеза Римана, касающаяся распределения простых чисел, и бинарная проблема Гольдбаха, утверждающая, что любое четное число больше двойки может быть представлено в виде суммы двух простых чисел.



*Кратер "Эратосфен" на Луне*

ЭТО ПОЛЕЗНО ЗНАТЬ

# Что нам стоит дом построить

**А.И. Азевич,**
Москва

*Сегодня ни один современный архитектор не может обойтись без AutoCAD. Этот программный пакет служит для подготовки проектов школ, домов, торговых центров. Но прежде чем замысел архитектора обретет свое реальное воплощение в виде модели — чертежа, стоит немало потрудиться. Помимо дорогого AutoCAD есть и похожие бесплатные программы. Они подойдут для начинающих. Например, A9CAD — аналог старшего собрата по чертежному цеху. С его помощью рутинный и скучный процесс построения чертежа можно превратить в настоящее искусство.*

Скачаем программу с сайта *http://www.bestfree.ru/soft/graph/draw.php*[1], установим ее на компьютер и приступим к созданию первого чертежа. Какой проект выбрать? Многофункциональный комплекс или современный кинотеатр? Нет, не будем пока погружаться в сложные рисунки. Начнем с простенького домика, а там посмотрим. Будем выполнять чертеж, по ходу осваивая многочисленные функции программы. Жаль, что интерфейс у нее англоязычный. Впрочем, это не страшно, можно воспользоваться одним из онлайн-переводчиков. Их в Интернете немало. Запасемся терпением и приступим к изучению.

Запускаем программу. На черном экране (*рис.* 1) в будущем должен появиться дом. Слева — набор инструментов для проектирования команд, вверху, как обычно, — меню и кнопки управления. Это все нужно освоить.

Черный фон неудобен для восприятия, да и для глаз утомителен. Заменим его белым. Для этого выполним последовательность команд: **File | General Settings | Background Color | white**. Фон стал белым. Конечно, белый фон лучше, чем черный, но как на нем строить чертежи? Это все равно, что в тетради без клеточек рисовать геометрические фигуры. Ровно и красиво, как ни старайся, не получится. Поэтому и здесь именно клетки, а точнее — сетка, необходимы для построения точного макета дома. Каждая вершина клетки — это узел, к которому можно привязать элементы проекта.

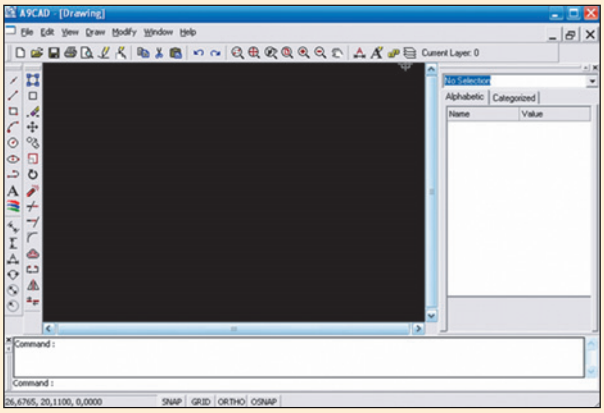


*Рис.* 1

Чтобы сетка появилась на экране, надо нажать на кнопку **GRID**, которая расположена в самом низу главного окна программы (его фрагмент приведен на *рис.* 2). Здесь же имеется еще несколько кнопок. Весьма полезна **SNAP**. Щелкнув по ней, активируем функцию привязки к сетке инструментов программы.

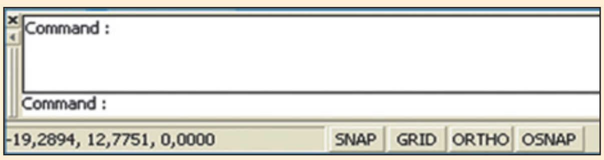



*Рис.* 2

Итак, все готово к построению. И все-таки, прежде чем возводить стены дома, познакомимся с панелью инструментов. По виду иконок нетрудно догадаться, где какая команда. Чего тут только нет —

[1] Дистрибутив этой программы приведен на диске к данному номеру журнала.

точка, линия, прямоугольник, дуга, круг и т.д. (см. *рис*. 1). Как только подведем мышку к какой-нибудь кнопке, тут же появится всплывающая подсказка, правда, на английском языке. В правой колонке — панель инструментов для редактирования. С их помощью объекты можно выделять, копировать, перемещать, удалять и т.д.

С чего же начать построение дома? С фундамента, конечно! Нарисуем прямоугольник. Для этого выберем инструмент “Прямоугольник” и на рабочем поле в нужном месте растянем его до нужных размеров. Как только “фундамент” появится на экране, дважды кликнем левой кнопкой мыши. Он “прицепится” в сетке. Нарисуем еще один прямоугольник (см. *рис*. 3). Это будет основная часть здания.

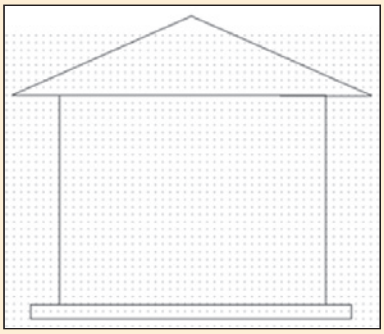


*Рис.* 3

Теперь перейдем к построению крыши. Отметим точку — начало нижней границы верхушки дома и проведем от нее прямую линию. Чтобы “остановить” линию, надо кликнуть два раза левой, а потом один раз правой кнопкой мыши. Используя линии, дорисуем крышу дома (см. *рис*. 3). На крыше сделаем небольшое круглое окошко, а в основной части дома — три прямоугольных окна. Справа поместим дверь. На крыше установим трубу; какой дом без печки, не правда ли? (*рис*. 4)

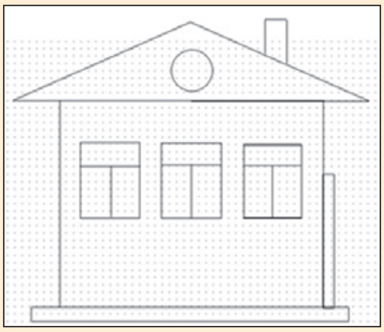


*Рис.* 4

Нехорошо, что толщина стен и окон одинакова. Исправим этот недостаток. Нажмем на значок и выделим основную часть дома. Тут же справа появится новое окно (*рис.* 5). Нажмем на параметр `Linetype` и в открывшемся окне выберем нужную толщину стен. Здесь же, кстати, можно выбрать и цвет элемента, щелкнув по надписи `Color`. Эти команды помогут сделать дом привлекательнее. Судите сами — *рис.* 5.

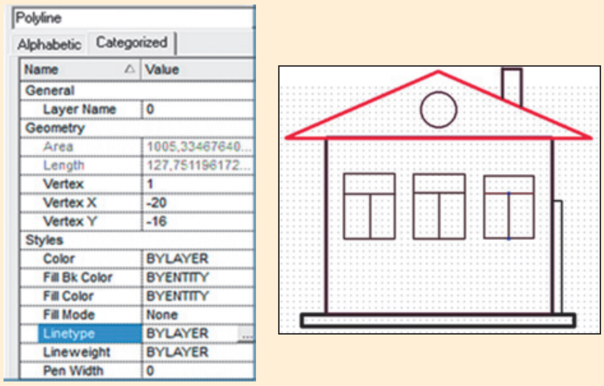



*Рис.* 5

Много сделали, но, кажется, чего-то еще не хватает. Конечно, надо раскрасить дом. Вновь воспользуемся панелью команд для редактирования (см. *рис.* 5). Изучим параметр `FillMode`. Выделим фундамент (как это сделать, было сказано выше) и щелкнем на кнопке `FillMode`. После этого открывается несколько вариантов заливки. Выберем вариант `HatchDiagCross` — фундамент окрасится в приятную “косую” клетку (см. *рис*. 7 на с. 50).

Теперь надо задать размеры деталей дома. Надо же знать, сколько потребуется материалов на его изготовление. И, кроме того, соблюдая габариты, в дальнейшем придется строить красивый дом не только на чертеже, но и в реальной обстановке. Для нанесения размеров служит специальная функция. Перед тем как ей воспользоваться, выделим нужный элемент. Пусть, например, это будет расстояние от фундамента до крыши. На главном меню программы выберем пункт **Draw**, в нем подпункт **TextStyleEditor**. В появившемся окне (*рис.* 6) укажем размеры текста и нажмем **OK**. Можно поступить и по-другому: выбрать подпункт **Dimension**, после чего — соответствующую надпись (выноска по вертикали, горизонтали, диагонали, диаметр круга или центральный угол дуги). Программа сама все измерит, а потом укажет на чертеже получившееся число.

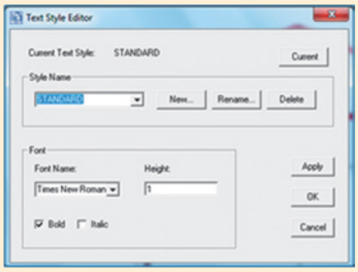



*Рис.* 6

На чертеже стоит выполнить и дополнительные построения: ось симметрии дома и линии соответствия отдельных элементов. И в этот раз на помощь придет окно, изображенное на *рис.* 5. В нем воспользуемся параметром `Linetype`. С его помощью нанесем на чертеж нужные линии. В результате получим следующий вид (*рис.* 7). Он уже похож на настоящий чертеж, с помощью которого можно строить дом.

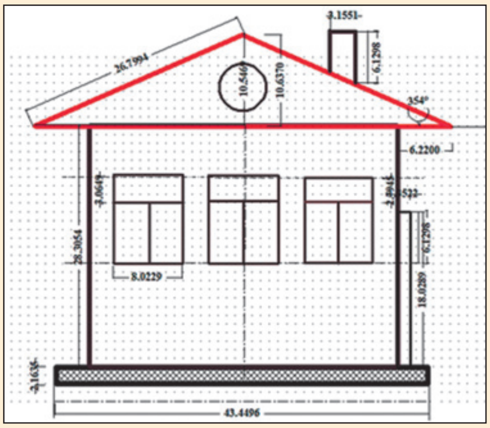



*Рис. 7*

Конечно, созданный нами проект значительно отличается от работы профессионального архитектора. У настоящих архитекторов и строителей все гораздо сложнее и больше — для здания нужна целая папка всевозможных чертежей. Но ведь с чего-то начинать надо? Особенно тем, кто хочет связать свою профессию с архитектурой или строительством. Как знать, может, с этого простого домика и родится в будущем новый Ле Корбюзье[2]?

**От редакции.** Предлагаем читателям присылать в редакцию изображения (в виде графического файла), созданные с помощью программы A9CAD. Фамилии всех приславших будут опубликованы, а лучшие работы мы поощрим.

## ДЛЯ ЭРУДИТОВ

### Викторина

Выберите из трех вариантов один с правильным ответом, используя информацию из Интернета или из других источников.

**1.** В апреле 2012 года, спустя сто лет после его кораблекрушения, останки корабля приобрели защиту Конвенции ЮНЕСКО 2001 года об охране подводного культурного наследия. Отныне государства — участники Конвенции имеют право воспрепятствовать уничтожению, разграблению, продаже и несанкционированному распространению найденных на месте катастрофы объектов. О каком лайнере речь?

а) “Лузитания”;
б) “Олимпик”;
в) “Титаник”.

**2.** Знатоки утверждают, что в свое время историческое ядро Нью-Йорка — Манхэттен выкупили у коренных индейцев за двадцать четыре доллара. Район знаменит не только Колумбийским университетом, но и, скажем, рестораном “Русский самовар”. Безусловно, визитной карточкой Манхэттена стали знаменитые на весь мир небоскребы. Один из них получил весьма бытовое прозвище. Какое?

а) “Чайник”; б) “Пылесос”; в) “Утюг”.

**3.** Недалеко от этого северного города, названного “Медвежьим углом” (в переводе с ненецкого языка), с разницей в три года прогремели два подземных атомных взрыва. Рассказывали, что они имели целью не военное, а научное значение. С их помощью зондировали глубинные земные слои. О каком городе речь?

а) Воркута; б) Салехард; в) Якутск.

**4.** В бывшем архиве политбюро обнаружили письма Генриха Ягоды к Сталину с сообщением о том, что на складе коменданта Кремля обнаружен личный сейф, который не вскрывался все 16 лет, прошедшие со дня смерти его хозяина. Там оказались золотые монеты царской чеканки на астрономическую по тем временам сумму (108 525 рублей), свыше семисот золотых изделий с драгоценными камнями, множество бланков паспортов, облигации царского времени. Кому из большевиков он принадлежал?

а) Свердлову;
б) Дзержинскому;
в) Каменеву.

**5.** Ирландский актер Ричард Харрис, сыгравший Альбуса Дамблдора в фильме “Гарри Поттер и философский камень”, трижды отказывался исполнить эту роль, так как был уже серьезно болен. Кто из его ближайших родственников пригрозил ему бессрочным бойкотом, после чего он согласился на съемки?

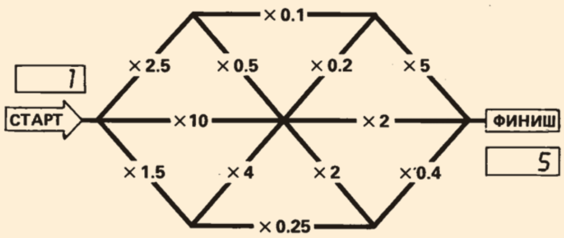
а) жена; б) сын; в) внучка.

*Источник — “Российская газета”*

## “ЛОМАЕМ” ГОЛОВУ

### Лабиринт из чисел

Попробуйте пройти через “лабиринт”, изображенный на рисунке:




Начиная с числа 1 (узел “СТАРТ”), надо, выполняя расчеты, соответствующие тому или иному отрезку, получить в результате число 5 (узел “ФИНИШ”). По каждой части “лабиринта” можно проходить только один раз (но в любом направлении).

Алгоритм решения задачи оформите в виде:

0. Начальное значение 1
1. × 1.5 = 1.5 2. × 4 = 6 3. …

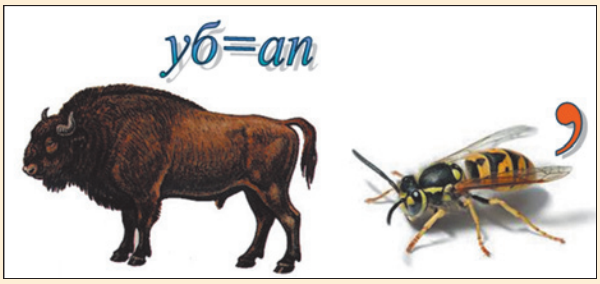
[2]Ле Корбюзье (1887–1965) — великий французский самобытный и оригинальный архитектор-реформатор.

## 14 ребусов

Решите, пожалуйста, ребусы, которые подготовил Денис Синица, ученик гимназии № 1 имени К.Калиновского, г. Свислочь, Республика Беларусь. Обратим внимание на то, что в некоторых ребусах зашифрованы одни и те же термины. Решив ребусы, определите также, с какой темой они связаны.

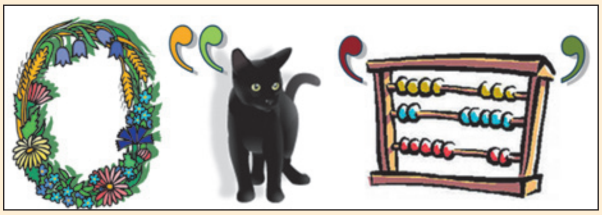
Ребус № 1


Ребус № 2



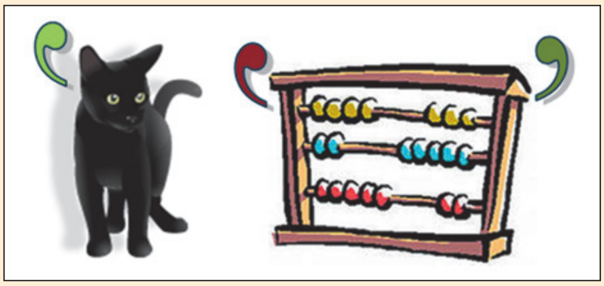

Ребус № 3


Ребус № 4

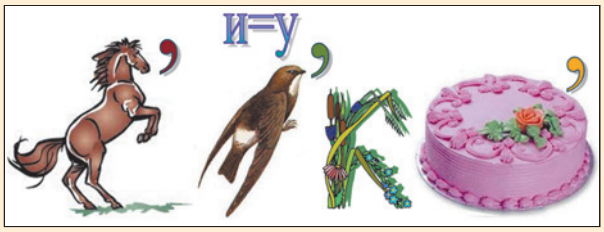



Ребус № 5


Ребус № 6



Ребус № 7




Ребус № 8


Ребус № 9


Ребус № 10

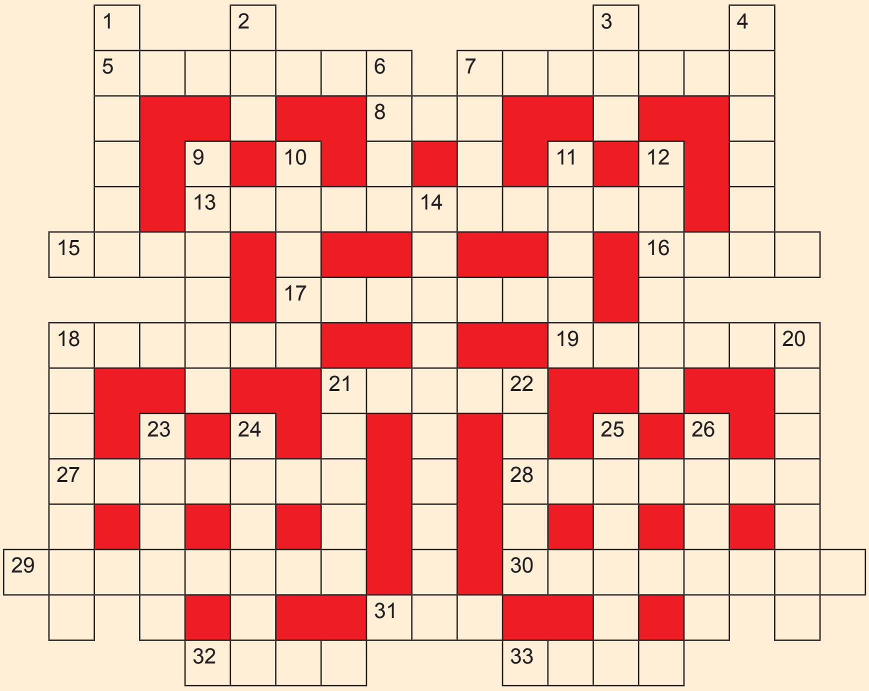
### Ребус № 11


### Ребус № 12


### Ребус № 13


### Ребус № 14

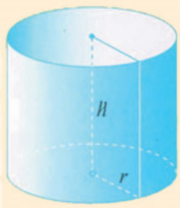

## Кроссворд



*По горизонтали*

5. Величина, над которой производится арифметическая операция.

7. Совокупность дорожек с одинаковым номером на магнитных дисках, а также:



8. Русский вариант логической связки в операции дизъюнкции.

13. Электронное вычислительное устройство.

15. … данных.

16. Замкнутая кривая, представляющая собой слегка вытянутую по горизонтали или вертикали окружность.

17. Математическое положение, истинность которого устанавливается путем доказательства.

18. Один из двух режимов ввода текста в текстовых редакторах.

19. Одно из основных чувств человека для восприятия информации.

21. “Среднее” между пунктом 11 по вертикали и операцией над информацией, производимой на мониторе.

27. Регулируемый параметр монитора.

28. Один из первых языков программирования высокого уровня.

29. Средство, облегчающее доступ к нужному месту в книге или во Всемирной паутине.

30. Веб-…

31. Структура данных, двусторонняя очередь.

32. Разговорное название результата обработки изображения с помощью устройства по пункту 9 по вертикали.

33. Воинское знамя в Древней Руси.

*По вертикали*

1. Наука о законах и формах мышления.

2. Цифра четверичной системы счисления.

3. Единица измерения количества информации.

4. Один из пунктов главного меню приложений пакета Microsoft Office.

6. Разновидность носителя информации.

7. Алгоритмическая конструкция, обеспечивающая повторение одних и тех же операций.

9. Устройство для ввода информации в персональный компьютер.

10. Деталь системного блока персонального компьютера.

11. Неисправность в работе компьютера.

12. Название клавиши.

14. Целенаправленная организация того или иного процесса, протекающего в системе, включающая в себя сбор информации, ее переработку и анализ, принятие решений, выработку воздействий на объекты и их доведение до этих объектов.

18 и 22. Знаки препинания.

20. Цифра двоичной системы счисления.

21. Женское имя.

23. Часть отсчетного устройства прибора, представляющая собой совокупность отметок (точек, штрихов, расположенных в определенной последовательности).

24. Мельчайшие частицы какого-либо вещества, выделившиеся из раствора, жидкости и осевшие на дно.

25. Название романа Фенимора Купера, а также степные равнины Северной Америки, в прошлом покрытые высокотравной растительностью.

26. Валюта, в которой получают зарплату программисты в Швеции.

## Числовой ребус

Решите, пожалуйста, числовой ребус:

**ABBA + A + B = CDDA**

Как обычно, одинаковыми буквами зашифрованы одинаковые цифры, разными буквами — разные цифры.

## Рубик рубит кубик

Изобретатель Рубик разрубает кубик размером $3 \times 3 \times 3$ на маленькие кубики размером $1 \times 1 \times 1$. Какое минимальное число раз ему придется взмахнуть топором, чтобы это сделать, если при разрубании разрешены наложения кусков кубика друг на друга?

## Два судоку

Решите, пожалуйста, две японские головоломки “судоку”:

1) простую:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | 6 | 7 | 2 | 5 | 8 | | 1 |
| 2 | 8 | | | 9 | | | 5 | 4 |
| | 9 | 7 | | | 4 | | | |
| 7 | | | 5 | | 8 | | 1 | |
| | 3 | 8 | | | | 2 | 4 | |
| | 6 | | 9 | | 2 | | | 7 |
| | | | 4 | | | 1 | 2 | |
| 4 | 7 | | | 8 | | | 6 | 9 |
| 8 | | 2 | 6 | 5 | 9 | 4 | | |

2) сложную:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | 5 | 1 | |
| | 5 | 9 | | 4 | | 2 | | |
| | 8 | 1 | 5 | | | 4 | | |
| 4 | | | 3 | | 5 | | 7 | |
| | 2 | | | | 4 | | | 5 |
| 5 | | | 1 | | 7 | | | 4 |
| 1 | 3 | | | | 6 | | 4 | |
| 7 | | | | 1 | | | | 2 |
| | | | | 5 | 8 | | | |

Ответы (можно не на все головоломки) присылайте в редакцию.

## Четыре девочки

Маша и Лена одинакового роста. Лена выше Оли, а Таня выше Маши. Расположите имена девочек в порядке возрастания их роста.


Задание предназначено для учеников начальной школы.

## Что начертил учитель?

Учитель начертил на классной доске четырехугольник. Вася утверждал, что это квадрат. Таня считала, что четырехугольник — трапеция. Армен думал, что на доске изображен ромб. Дима назвал четырехугольник параллелограммом. Выслушав каждого и обстоятельно изучив свойства четырехугольника, учитель установил, что ровно три из четырех утверждений истинны и ровно одно утверждение ложно.

Какой четырехугольник начертил учитель на классной доске?

## В санатории

В санатории на берегу моря отдыхают отец, мать, сын и две дочери. До завтрака члены семьи купаются в море, причем известно, что если отец утром отправляется купаться, то с ним обязательно идут мать и сын; если сын идет купаться, то первая дочь идет с ним; вторая дочь купается тогда и только тогда, когда купается мать; каждое утро купается по крайней мере один из родителей. Если в воскресенье утром среди купающихся членов семьи была лишь одна из дочерей, то кто в это утро ходил на море?

## Лицеисты

В лицее информационного профиля, где учатся больше 225, но меньше 245 учеников, часть учеников являются отличниками, а остальные хорошистами. После контрольной работы по информатике 2/7 отличников стали хорошистами, а хорошисты так и остались хорошистами за исключением одного человека, который стал троечником. При этом хорошистов и отличников стало поровну. Сколько учеников могло быть в лицее? Вспомогательные расчеты проведите, используя электронную таблицу Microsoft Excel или др.

Задача подготовлена по материалам сайта *www.diofant.ru* (задача 2469, автор Сергей Меньшов)

## Запутанный ответ

У одного ученика спросили:

— Как называется книга, которую ты прочитал?

Ученик ответил:

— В название книги входят 4 буквы. Если эти буквы заменить числами, которые они по порядку занимают в русском алфавите, то эти числа будут иметь следующие свойства: сумма их будет равна 40; первое число равно третьему; сумма второго и четвертого составляет половину третьего, а разность между четвертым и вторым равна одной восьмой первого числа.

Какую книгу прочел ученик?

## Чемпионат школы

В городе Олимпийске, в школе № 11, команды 11-го “А” и 11-го “Б” разыграли первенство школы среди 11-х классов в десяти видах спорта. За победу команда получала 4 очка, за ничью — 2, за проигрыш — 1 очко. Победила команда 11-го “Б” класса, причем вместе обе команды набрали 46 очков. Сколько было ничьих? Можно ли определить, сколько очков набрала каждая команда?

*Литература*

1. “На досуге”. Сборник занимательных задач. Ростов-на-Дону: Ростовское книжное изд-во, 1959.

## Четыре гнома

Каждый из четырех гномов — Беня, Веня, Женя и Сеня — либо всегда говорит правду, либо всегда врет. Отличник по информатике услышал такой разговор: Беня — Вене: “Ты врун”; Женя — Бене: “Сам ты врун”; Сеня — Жене: “Да оба они вруны, — (подумав), — впрочем, ты тоже”. Он понял, кто из них говорит правду. А вы сможете?

Источник задачи: *http://mmmf.msu.ru/archive/19992000/bugaenko/b11.html*

## Сложение двух двоичных цифр

Как известно, таблица сложения в двоичной системе счисления выглядит так:

$$\begin{array}{r} 0\\ +\,0\\ \hline 0 \end{array} \qquad \begin{array}{r} 0\\ +\,1\\ \hline 1 \end{array} \qquad \begin{array}{r} 1\\ +\,0\\ \hline 1 \end{array} \qquad \begin{array}{r} 1\\ +\,1\\ \hline 1\,0 \end{array}$$

Если обозначить слагаемые $a$ и $b$, то можно записать следующие условия для определения значения цифры результата $r$:

```
если a = 0 и b = 0 или a = 1 и b = 1
   то
      r = 0
все
если a = 0 и b = 1 или a = 1 и b = 0
   то
      r = 1
все
```

и значения переноса $p$:

```
если a = 0 и b = 0 или a = 0 и b = 1
     или a = 1 и b = 0
   то
      p = 0
все
если  a = 1 и b = 1
   то
      p = 1
все
```

Но приведенные записи можно существенно сократить.

**Задание**

Запишите формулы, определяющие значения цифры результата $r$ и значения переноса $p$ в зависимости от значения слагаемых $a$ и $b$, в которых операции сравнения (условия) не используются.

*Указания по выполнению*

Ознакомьтесь с логическими операциями, которые применяются к числам (см. статью “Логические и сдвиговые операции” в “Информатике” № 16/2011). Напомним, что эти операции выполняются в процессоре компьютера (поэтому их называют также логическими командами) над числами, представленными в двоичном виде. Рассмотрим те логические команды, которые выполняются над двумя числами (говорят, что у них два операнда):

1) **AND** (русский вариант — **И**);
2) **OR** (**ИЛИ**);
3) **XOR** (от англ. *eXclusive OR* — **исключающее ИЛИ**).

В отличие от арифметических операций над двумя операндами логические команды являются *поразрядными*. Например, при сложении двух двоичных цифр возможен перенос в старший разряд, а при логических операциях все разряды рассматриваются изолированно друг от друга. Разумеется, действия над всеми разрядами выполняются параллельно и одновременно. Описанные операции называют также “битовыми”.

Правила выполнения логических операций в каждом разряде представлены в таблице:

| X | Y | X AND Y | X OR Y | X XOR Y |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 0 |

Ответы присылайте в редакцию.

## Две банки с бактериями ☺

Предлагаем учащимся 1–7-х классов решить следующие задачи.

### Задача 1

В банке находятся бактерии. Через минуту каждая из бактерий делится пополам, затем каждая из получившихся бактерий через секунду делится пополам и т.д. Через час банка полна. Можно ли определить, через какое время банка была заполнена наполовину?

### Задача 2

В банку поместили одну бактерию. Через минуту она поделилась пополам, затем каждая из получившихся бактерий через минуту делится пополам и т.д. Через час банка стала полной. Через какое время банка будет заполнена, если в нее поместить:

а) две бактерии;
б) четыре бактерии?

## Ответы, решения, разъяснения

к заданиям, опубликованным в разделе “В мир информатики” ранее

### Задача “Кто этот человек?”

Напомним, что необходимо было, решив пять задач по программированию, по полученным результатам определить информацию о некотором человеке, после чего назвать его фамилию и имя, а также награды (премии, ордена и медали), которых удостоился этот человек.

*Ответ*

Человек, о котором идет речь в заданиях, — норвежский ученый в области теории вычислительных систем Оле-Йохан Даль (12.10.1931 — 29.06.2002). Его награды:

— командор ордена Святого Олафа;
— Премия Тьюринга за идеи, фундаментальные для развития объектно ориентированного

программирования, возникшие в ходе разработки языков программирования Simula I и Simula 67 (2001 год);

— медаль IEEE имени Джона фон Неймана (2001 год).

*Правильные ответы представили*:

— Александрова Алена, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Алишева Лейла, Республика Башкортостан, г. Уфа, гимназия № 3, учитель **Бадыков С.Р.**;

— Амосов Евгений, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Андрющенко Александр и Свистунов Николай, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

— Бородюк Анна и Василенко Татьяна, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Волков Владимир и Глушаков Андрей, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Воскресенский Денис, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Габулов Виталий, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Добрякова Светлана, Козина Мария и Чернова Наталья, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Коноплева Кристина, Свердловская обл., г. Ревда, школа № 10, учитель **Игошева А.А.**;

— Коробов Сергей, Марков Алексей и Яснов Федор, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Сапрыкин Виктор, г. Воронеж, школа № 5 им. К.П. Феоктистова, учитель **Чернышева И.А.**;

— Трошин Иннокентий, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

## Задание “Шесть вопросов” (рубрика “Поиск информации”)

*Ответы*

1. Герой рассказа Эрнеста Хемингуэя “Старик и море” по имени Сантьяго по роду занятий был рыбаком.

2. Настоящее (и полное) имя Арамиса, героя романов Александра Дюма “Три мушкетера” и др., — “Рене, шевалье (аббат) д’Эрбле, епископ Ваннский, герцог д’Аламеда (Аламедский)”. (В одном из присланных ответов указывалось — “Игорь Старыгин” ☺.)

3. Камень, которым в Древней Индии успокаивали младенцев и помогали им быстро стать на ноги, — агат.

4. За 14 миллионов рублей царская Россия продала Соединенным Штатам Америки полуостров Аляска и Алеутские острова.

5. В новогоднем убранстве японского дома добродетель символизирует сосна (в ответах приведены и другие близкие по смыслу варианты).

6. Из олимпийских чемпионов книгу о воспитании детей (“Ребенок и уход за ним”) написал Бенджамин Спок, чемпион Олимпийских игр 1924 года по гребле (в ряде ответов ошибочно указывались авторы книги “Олимпийский чемпион”).

На фотографии, опубликованной в задании, изображен Игорь Владимирович Старыгин, киноактер, сыгравший Арамиса в отечественном кинофильме “Три мушкетера”.

*Ответы представили*:

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

— Воронова Софья, Голомёдова Алина, Еремеева Дарья, Иванова Нелли и Кириллова Валерия, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Воскресенский Денис, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Герасимов Станислав и Друк Павел, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Гильзер Ирина, Республика Башкортостан, г. Уфа, лицей № 60, учитель **Гильзер Н.В.**;

— Гильманова Диана, Жолтаева Евгения, Жукова Ольга, Кирдань Анастасия и Князева Екатерина, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Коноплева Кристина, Свердловская обл., г. Ревда, школа № 10, учитель **Игошева А.А.**;

— Легостаев Алексей, Москва, кадетская школа-интернат № 5 “Преображенский кадетский корпус”, учитель **Сергеев С.А.**;

— Лопаткина София, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Маркина Ирина и Носов Владимир, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Массольд Валерия, средняя школа села Ириновка, Новобурасский р-н Саратовской обл., учитель **Брунов А.С.**;

— Наделяев Денис, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н, учитель **Краснёнкова Л.А.**;

— Назаров Виталий, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Озерова Анна, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Синюк Юлия, г. Воронеж, школа № 5 им. К.П. Феоктистова, учитель **Чернышева И.А.**;

— Тарасов Алексей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Тимофеев Владислав, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Трептау Татьяна, Вадьковская средняя школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**;

— Туголукова Ольга и Хутилашвили Виктория, Костромская обл., Буйский р-н, г.п.п. Чистые Боры, школа № 1, учитель **Васнина О.В.**;

— Шафиева Диана, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**

## Задача на восстановление IP-адреса

Напомним, что требовалось восстановить полный IP-адрес по четырем разрозненным частям:

1) 9.69 3) 1.21
2) 25 4) .41

*Решение*

Как известно, IP-адреса записываются в виде четырех неотрицательных целых чисел, меньших 256, разделенных точками. Учитывая точки в приведенных в условии частях, можно сформировать два варианта полного адреса:

1) 259.691.21.41;
2) 251.219.69.41.

Первый вариант не подходит, так как в нем имеется число, большее 255.

*Ответ*: 251.219.69.41. (Ясно, что без точки в конце ☺.)

*Ответы прислали*:

— Волков Владимир и Глушаков Андрей, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Добрякова Светлана, Козина Мария и Чернова Наталья, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Дукач Светлана, Лысенко Екатерина, Овчинникова Елизавета и Цурин Сергей, г. Лесосибирск Красноярского края, поселок Стрелка, школа № 8 им. Константина Филиппова, учитель **Лопатин М.А.**;

— Ежов Андрей, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Иванова Светлана, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Киришина Татьяна, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н, учитель **Краснёнкова Л.А.**;

— Лавренов Руслан, Вадьковская средняя школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**;

— Лигай Илларион, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Лопаткина София, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Маркина Ирина и Носов Владимир, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Назарова Елена, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

## Ребусы по информатике

*Ответы* (соответствуют номерам ребусов): 1. СТРУКТУРНАЯ. 2. МАТЕМАТИЧЕСКАЯ. 3. СЛОВЕСНАЯ. 4. АЛГОРИТМИЧЕСКАЯ. 5. СИМВОЛЬНАЯ. 6. ТАБЛИЧНАЯ. 7. ГРАФИЧЕСКАЯ. 8. ЗНАКОВАЯ. 9. СИГНАЛЫ И ЖЕСТЫ.

*Правильные ответы прислали*:

— Алтынникова Алена, средняя школа села Ириновка, Новобурасский р-н Саратовской обл., учитель **Брунов А.С.**;

— Андрющенко Александр и Свистунов Николай, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

— Воскресенский Денис, Тананаева Анастасия и Тананаева Ксения, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Добрякова Светлана, Козина Мария и Чернова Наталья, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Дукач Светлана, Лысенко Екатерина, Овчинникова Елизавета и Цурин Сергей, г. Лесосибирск Красноярского края, поселок Стрелка, школа № 8 им. Константина Филиппова, учитель **Лопатин М.А.**;

— Еремеев Владимир, Кондакова Юлия и Ушаков Андрей, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Иванова Светлана, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Игошев Константин, Свердловская обл., г. Ревда, школа № 10, учитель **Игошева А.А.**;

— Иксанова Ирина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Киришина Татьяна, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н, учитель **Краснёнкова Л.А.**;

— Ломакин Илья, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Мальцев Владислав, г. Воронеж, школа № 5 им. К.П. Феоктистова, учитель **Чернышева И.А.**;

— Назарова Елена, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Новиков Степан и Тарасов Алексей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Пугачева Диана, Иркутская обл., г. Братск, школа № 3, учитель **Середкина Т.Ю.**

Большинство приславших ответы правильно установили, что ребусы связаны с темой “Виды информации”.

## Числовой ребус “Японские города”

Напомним, что предлагалось установить, при каком основании системы счисления имеет решение (и какое именно) числовой ребус:

| | | | | | |
|---|---|---|---|---|---|
| + | | К | И | Т | О |
| | К | И | О | Т | О |
| | Т | О | К | И | О |

*Решение*

Обозначим искомое основание — $x$.

Из первого столбца справа следует, что **О** = 0. Для удобства запишем это в ребус:

| | | | | | |
|---|---|---|---|---|---|
| + | | К | И | Т | 0 |
| | К | И | 0 | Т | 0 |
| | Т | 0 | К | И | 0 |

Это означает, что перенос единицы “в уме” происходит из второго столбца в третий и из четвертого в пятый. Значит, можем записать:

**Т** + **Т** = $x$ + **И**

**И** + 1 = **К**

**К** + **И** = $x$

**К** + 1 = **Т**

Следовательно, $x$ = 7, **И** = 3, **К** = 4, **Т** = 5.

*Правильные ответы прислали*:

— Волков Владимир, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Дукач Светлана, Лысенко Екатерина, Овчинникова Елизавета и Цурин Сергей, г. Лесосибирск Красноярского края, поселок Стрелка, школа № 8 им. Константина Филиппова, учитель **Лопатин М.А.**;

— Жбан Владимир, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Иванова Светлана, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Иксанова Ирина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Лопаткина София, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

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

— Назарова Елена, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Новиков Степан и Тарасов Алексей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**

## Чай и молоко

Напомним, что требовалось, используя электронную таблицу Microsoft Excel или др., решить следующую задачу: “Из чашки молока три ложки содержимого перелили в такую же чашку чая и тщательно перемешали смесь. Затем три ложки смеси перелили обратно в чашку с молоком. Чего теперь больше: чая в молоке или молока в чае?”

*Ответ*

Чая в молоке будет столько же, сколько молока в чае.

*Правильные ответы представили*:

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

— Васина Светлана и Хомутова Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Лопаткина София, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Лукинская Екатерина и Шумкина Елена, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Мороз Евгений, Костромская обл., Буйский р-н, г.п.п. Чистые Боры, школа № 1, учитель **Васнина О.В.**;

— Новикова Ирина, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Онуфриев Сергей и Харина Елена, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Павленко Иван и Рязановская Ирина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Сороченко Иван, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

## Судоку

*Правильные ответы прислали*:

— Бадикова Ольга, Республика Башкортостан, г. Уфа, лицей № 60, учитель **Гильзер Н.В.**;

— Батурин Владислав, Сафиуллина Светлана, Терешкина Дарья и Шумаков Николай, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Воскресенский Денис, Тананаева Анастасия, Тананаева Ксения, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Киришина Татьяна, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н, учитель **Краснёнкова Л.А.**;

— Краснова Диана и Смирнягина Александра, Свердловская обл., г. Ревда, школа № 10, учитель **Игошева А.А.**;

— Крупин Василий и Хомутова Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Лопаткина София, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Лушникова Елена, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Массольд Валерия, средняя школа села Ириновка, Новобурасский р-н Саратовской обл., учитель **Брунов А.С.**;

— Новиков Степан и Тарасов Алексей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Попкова Полина, средняя школа села Ириновка, Новобурасский р-н Саратовской обл., учитель **Брунов А.С.**;

— Попов Александр (другие сведения, к сожалению, не приведены);

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

— Рязановская Ирина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Селютин Даниил, г. Ярославль, школа № 33, учитель **Цикина Е.Н.**;

— Согомонян Серине, Воронежская обл., поселок Каменка, средняя школа № 1 им. Героя Советского Союза В.П. Захарченко, учитель **Старикова М.Е.**;

— Трофименкова Елизавета, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Фоменко Анастасия, г. Воронеж, школа № 5 им. К.П. Феоктистова, учитель **Чернышева И.А.**;

— Шафиева Диана, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**

## Кроссворд

*Ответы*

*По горизонтали*: 2. Листинг. 4. Яга. 5. Три. 8. Адаптер. 12. Записка. 13. Позиция. 14. Тип.

*По вертикали*: 1. Этап. 2. Логика. 3. Горнер. 6. Два. 7. Бит. 9. Диалог. 10. Примитив. 11. Евклид.

*Правильные ответы прислали*:

— Абдрахманова Валерия, Ковалева Мария, Костылева Юлия и Миронова Екатерина, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Андрющенко Александр и Свистунов Николай, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Гуцал Анна и Любимова Ирина, Костромская обл., Буйский р-н, г.п.п. Чистые Боры, школа № 1, учитель **Васнина О.В.**;

— Дукач Светлана, Лысенко Екатерина, Овчинникова Елизавета и Цурин Сергей, г. Лесосибирск Красноярского края, поселок Стрелка, школа № 8 им. Константина Филиппова, учитель **Лопатин М.А.**;

— Закатов Влад (так указано в письме), г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Иванова Анна и Мальцева Виктория, г. Воронеж, школа № 5 им. К.П. Феоктистова, учитель **Чернышева И.А.**;

— Казакова Ирина и Хромов Юрий, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Кротов Олег, Тананаева Анастасия и Тананаева Ксения, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Кубко Виктор, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Лопаткина София и Ратников Артем, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Макарова Ольга, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Онуфриев Сергей и Харина Елена, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Павленко Иван и Рязановская Ирина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Трептау Татьяна, Вадьковская средняя школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**;

— Трофименкова Елизавета, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Ушков Семен, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Шейкин Александр, средняя школа села Ириновка, Новобурасский р-н Саратовской обл., учитель **Брунов А.С.**

## Задача “Про Федю”

Напомним условие: “После родительского собрания к учительнице подошел один из родителей.

— Вот вы не назвали моего ученика среди хороших учеников, — сказал он. — А ведь мой Федя — отличник и к тому же лучший лыжник класса.

— Да, вы правы, — ответила учительница, — но хорошим учеником мы считаем ученика, который хорошо учится, дисциплинирован, помогает в учебе отстающим и, кроме того, участвует в работе научного общества или занимается спортом. А ваш Федя…

Что еще собиралась сказать учительница Фединому папе (учитывая, что она не назвала Федю в числе хороших учеников)?”

*Решение*

Условие, по которому в школе будем определять хорошего ученика, можно записать в виде:

“хорошо учится” **И** “дисциплинирован” **И** “помогает в учебе отстающим” **И** (“участвует в работе научного общества” **ИЛИ** “занимается спортом”).

Логическое выражение в скобках применительно к Феде истинное (он — лучший лыжник класса). Так как учительница не включила Федю в число хороших учеников, то это значит, что все приведенное логическое выражение, в котором используются логические связи “**И**”, — ложно. Это может быть когда ложны оба или одно из условий — “дисциплинирован” и “помогает в учебе отстающим”.

Значит, учительнца могла сказать:

1) “А ваш Федя не дисциплинирован и отстающим ребятам в учебе не помогает”;

2) “А ваш Федя не дисциплинирован”;

3) “А ваш Федя отстающим ребятам в учебе не помогает”.

*Правильные ответы представили*:

— Васина Светлана и Хомутова Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Дубовик Анна, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Казакова Ирина и Хромов Юрий, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Кубко Виктор, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Лазаренко Илья, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Морякова Анна и Озерова Наталья, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Наделяев Денис, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н, учитель **Краснёнкова Л.А.**;

— Орешников Иван, г. Лесосибирск Красноярского края, поселок Стрелка, школа № 8 им. Константина Филиппова, учитель **Лопатин М.А.**;

— Плахов Иван, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

## Задача “На ипподроме”

Напомним, что требовалось установить, как жокеи-фавориты *A*, *B* и *C* поделили между собой три места, если перед началом бегов на ипподроме четыре знатока из числа зрителей так обсуждали их шансы:

(1): “Заезд выиграет *A* или *C*”;
(2): “Если *A* будет вторым, то выиграет *B*”;
(3): “Если *A* придет третьим, то *C* не выиграет”;
(4): “Вторым придет *A* или *B*”,

а после заезда выяснилось, что эти три фаворита действительно заняли первые места и что все четыре утверждения знатоков оказались истинными.

*Решение*

Так как фавориты *A*, *B* и *C* действительно заняли первые три места, возможны только следующие 6 исходов заезда (справа в скобках указаны утверждения, которым противоречат эти исходы):

а) *ABC* —
б) *ACB* (4)
в) *BAC* (1)
г) *BCA* (1) (4)
д) *CAB* (2)
е) *CBA* (3)

В пяти случаях предполагаемая очередность прихода фаворитов к финишу противоречит по крайней мере одному утверждению. Всем условиям задачи удовлетворяет распределение, при котором фаворит *A* пришел первым, *B* — вторым, *C* — третьим.

*Правильные ответы прислали*:

— Базанов Илья, Воскресенский Денис и Головченко Тихон, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Галушкова Карина, основная школа поселка Бочилово, Республика Карелия, Пудожский р-н, учитель **Богданова Г.В.**;

— Дольникова Анастасия и Корытов Юрий, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Дубовик Анна, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Дукач Светлана, Лысенко Екатерина, Овчинникова Елизавета и Цурин Сергей, г. Лесосибирск Красноярского края, поселок Стрелка, школа № 8 им. Константина Филиппова, учитель **Лопатин М.А.**;

— Колобов Николай и Колобов Павел, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Лазаренко Илья, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Лопаткина София и Ратников Артем, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

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

— Наделяев Денис, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н, учитель **Краснёнкова Л.А.**;

— Плахов Иван, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

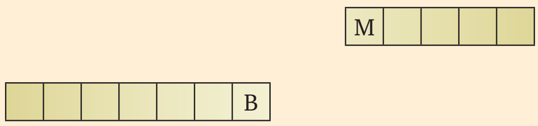
— Якубова Виктория, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

## Задача “Сколько вагонов в поезде?”

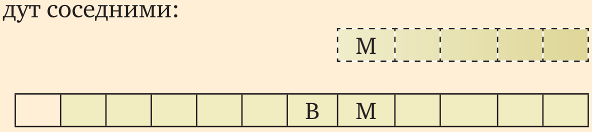
Напомним, что необходимо было определить количество вагонов в поезде, в котором Митя и Вася едут в соседних вагонах поезда, если вагон, в котором едет Митя, — пятый от “головы” поезда, а вагон, в котором едет Вася, — седьмой с “хвоста”.

*Решение*

Если принять, что “голова” поезда — справа, то части поезда, в которых едут мальчики, можно изобразить так:




По условию, мальчики едут в соседних вагонах. Возможны два варианта размещения приведенных частей поезда, при которых вагоны, обозначенные первыми буквами имен мальчиков (“М” и “В”), будут соседними:



и

| | | | | |
|---|---|---|---|---|
| М | | | | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | М | В | | | |

Соответственно, возможны два варианта ответа (общего числа вагонов): 12 и 10.

*Правильные ответы прислали*:

— Азаренко Николай, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Андриенко Татьяна и Филичкина Анна, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Базанов Илья, Воскресенский Денис, Головченко Тихон, Кротов Олег, Пономарева Татьяна и Салов Даниил, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Дольников Евгений, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Красавин Андрей, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Лякина Галина и Юферева Снежана, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Орешников Иван, г. Лесосибирск Красноярского края, поселок Стрелка, школа № 8 им. Константина Филиппова, учитель **Лопатин М.А.**;

— Пятакова Виктория, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Цыганков Евгений, Вадьковская средняя школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**

## Головоломка “Крест-накрест”

Напомним, что требовалось, переставив буквы в строках приведенного квадрата, получить “осмысленные” слова, при этом в диагоналях квадрата должны “собраться” еще два слова, связанные с информатикой и компьютерами.

*Ответ*

| | | |
|---|---|---|
| Б | И | Т |
| Ф | О | Н |
| К | О | Д |

Слова на диагоналях — БОД и ТОК.

*Правильные ответы представили*:

— Азаренко Николай, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Андрющенко Александр и Свистунов Николай, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

— Базанов Илья и Кротов Олег, г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Васина Светлана и Хомутова Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

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

— Дукач Светлана, Лысенко Екатерина, Овчинникова Елизавета и Цурин Сергей, г. Лесосибирск Красноярского края, поселок Стрелка, школа № 8 им. Константина Филиппова, учитель **Лопатин М.А.**;

— Жарова Елена и Юферева Снежана, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Задорина Наталья и Шумаков Николай, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Красавин Андрей, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Краснова Диана, Свердловская обл., г. Ревда, школа № 10, учитель **Игошева А.А.**;

— Лавренов Руслан, Вадьковская средняя школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**;

— Лопаткина София и Ратников Артем, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Филичкина Анна, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Якубова Виктория, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

## Четыре загадки-анекдота

Напомним, что было необходимо выбрать один из предложенных ответов на четыре вопроса так, чтобы получился остроумный анекдот.

*Ответы*

1. Если программист в девять утра уже на работе, значит, “он еще на работе” (вариант 3), хотя главная причина в том, что он “не успел вчера закончить программу” (вариант 4).

2. Программист не вышел на работу. День нет..., два нет... Коллеги по работе забеспокоились, пришли к нему домой проведать. Звонят в дверь. Никто не открывает. Дверь выбили, зашли и нашли его в ванной судорожно гладящего одной рукой свою голову, а в другой руке — шампунь.

На вопрос: “Что случилось?” — он ткнул пальцем на инструкцию по пользованию шампунем: “Нанести раствор на влажные волосы. Вспенить. Смыть. ...”.

Далее в инструкции было написано: “Повторить” (вариант 4).

3. Окончил программист институт и устроился на работу. Начальник спрашивает:

— Вы сильный программист?

— Ну как вам сказать...

— Ну сильный?

— В общем, да.

— Тогда — “перенесите эти компьютеры на третий этаж” (вариант 1).

4. Заходит программист в лифт и вспоминает, что ему надо попасть на 12-й этаж. Нажимает он “1”, потом “2” и начинает судорожно… “искать кнопку Enter” (вариант 2).

*Правильные ответы представили*:

— Антонов Дмитрий, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Григорьева Марина и Макаркина Анна, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Дукач Светлана, Лысенко Екатерина, Овчинникова Елизавета и Цурин Сергей, г. Лесосибирск Красноярского края, поселок Стрелка, школа № 8 им. Константина Филиппова, учитель **Лопатин М.А.**;

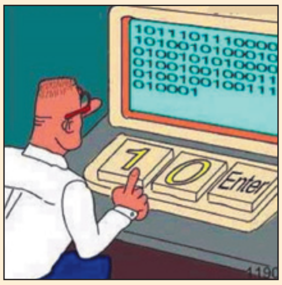
— Истомина Анна, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Кулистиков Дмитрий, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

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

— Михайлова Валерия, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Трептау Татьяна, Вадьковская средняя школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.** (Татьяна включила в ответ картинку, приведенную справа);




— Филичкина Анна, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

Ответы к заданиям, опубликованным в рубрике “Крепкий орешек”, представили:

1) задача “Расшифровка текста”:

— Клипперт Илона, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Лопаткина София, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Хомутова Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

2) головоломка “Загадочное деление”:

— Ельников Дмитрий, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Лопаткина София, г. Ярославль, школа № 33, учитель **Ярцева О.В.**

Уже после подготовки списков читателей, приславших ответы на задания, опубликованные в августовском номере журнала, в редакцию пришло письмо от Трептау Татьяны, ученицы Вадьковской средней школы (Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**), с решениями кроссворда, ребусов и ответами на задания “Поиск информации” и “Три загадки-анекдота”.

Решения ребусов Татьяна оформила в виде таблицы с фамилиями ученых и их портретами, а ответы, связанные с поиском информации, снабдила красочными иллюстрациями, среди которых — Указ императора Александра I, о котором шла речь в задании. Редакция решила наградить Татьяну дипломом. Поздравляем!

Решение головоломки “Двенадцать спичек” представили также студенты Альметьевского медицинского колледжа (Республика Татарстан, г. Альметьевск, преподаватель **Аристова Н.А.**) Аглиуллина Рузиля и Ильина Ольга. В ответе они привели фотографию, иллюстрирующую решение (ее автор — Чулпан Галиева).

## ВНИМАНИЕ! КОНКУРС

### Итоги конкурса № 94

Напомним, что предлагалось решить японскую головоломку “судоку”, в которой 2, 3 или 4 клетки были объединены в блоки, и сумма цифр в клетках блока должна быть равна числу, указанному в углу блока.

*Решение*

Полностью описывать методику решения не будем, а приведем некоторые рекомендации по решению указанных головоломок.

Для решения целесообразно подготовить табличку с различными вариантами “набора” той или иной суммы из *различных* цифр:

| Сумма | Варианты из двух цифр | Варианты из трех цифр |
|---|---|---|
| 3 | 1 и 2 (или 2 и 1) | — |
| 4 | 1 и 3 (или 3 и 1) | — |
| 5 | 1 и 4 (или 4 и 1) | — |
| | 2 и 3 (или 3 и 2) | |
| 6 | 1 и 5 (или 5 и 1) | 1, 2 и 3 (в разных сочетаниях) |
| | 2 и 4 (или 4 и 2) | |
| … | … | … |

Начинать желательно с анализа блоков, сумма в которых может быть набрана из минимального числа комбинаций цифр (3, 4, 16 и т.п.). По мере заполнения отдельных блоков количество возможных вариантов заполнения других блоков будет уменьшаться. Естественно, при решении следует учитывать правила классического судоку — цифры в квадратах 3 × 3 не должны повторяться.

*Ответ*:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 7 | 2 | 6 | 8 | 3 | 1 | 4 | 5 | 9 |
| 4 | 3 | 8 | 5 | 7 | 9 | 2 | 6 | 1 |
| 5 | 1 | 9 | 6 | 2 | 4 | 3 | 7 | 8 |
| 6 | 8 | 7 | 1 | 5 | 2 | 9 | 3 | 4 |
| 1 | 9 | 5 | 4 | 8 | 3 | 6 | 2 | 7 |
| 2 | 4 | 3 | 9 | 6 | 7 | 1 | 8 | 5 |
| 8 | 5 | 1 | 2 | 9 | 6 | 7 | 4 | 3 |
| 3 | 6 | 4 | 7 | 1 | 5 | 8 | 9 | 2 |
| 9 | 7 | 2 | 3 | 4 | 8 | 5 | 1 | 6 |

Победителями конкурса признаны читатели, не побоявшиеся взяться за решение непростого задания и правильно решившие его:

— Бадикова Ольга, Республика Башкортостан, г. Уфа, лицей № 60, учитель **Гильзер Н.В.**;

— Галушкова Карина, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Богданова Л.М.**;

— Краснова Диана, Свердловская обл., г. Ревда, школа № 10, учитель **Игошева А.А.**;

— Лопаткина София, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Манукян Гаянэ, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

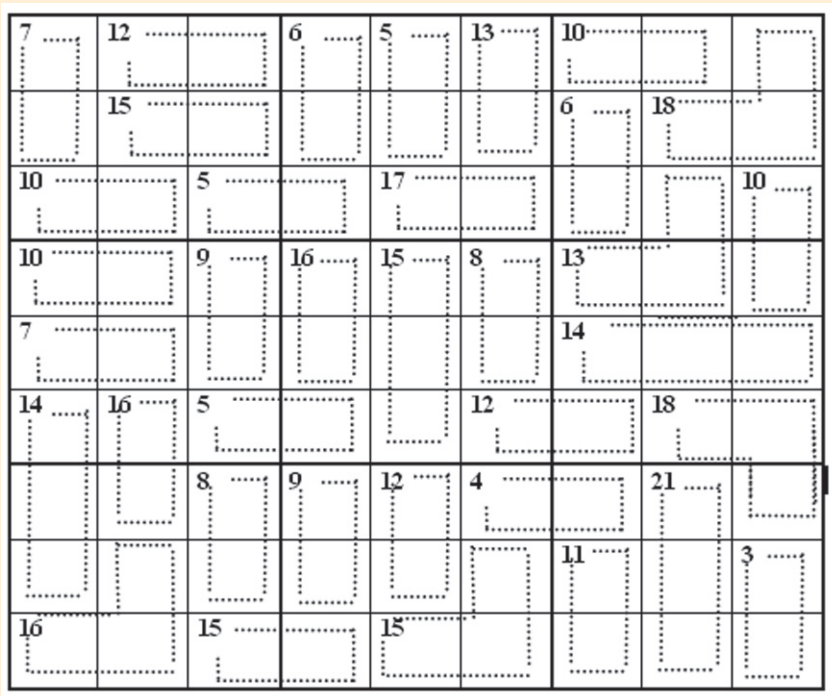
— Фоменко Анастасия, г. Воронеж, школа № 5 им. К.П. Феоктистова, учитель **Чернышева И.А.**

(Странно, что среди победителей не оказалось ни одного юноши ☹.)

Все они будут награждены дипломами. Поздравляем!

## Конкурс № 100 “Юбилейный”

В качестве задания этого конкурса также предлагаем решить судоку с суммами:




Условия описаны выше.

Ответы отправьте в редакцию до 1 марта по адресу: 121165, Москва, ул. Киевская, д. 24, “Первое сентября”, “Информатика” или по электронной почте: *vmi@1september.ru*. Пожалуйста, четко укажите в ответе свои фамилию и имя, населенный пункт, номер и адрес школы, фамилию, имя и отчество учителя информатики.

**Внимание выпускников!** Итоги этого конкурса могут быть подведены в летних номерах журнала, поэтому будущих победителей конкурса просим прийти в школу после выпускного вечера и получить дипломы.