

№9

Издание основано в 1995 г.

inf.1september.ru

УЧЕБНО - МЕТОДИЧЕСКИЙ ЖУРНАЛ ДЛЯ УЧИТЕЛЕЙ ИНФОРМАТИКИ

# ИНФОРМАТИК А



издательский  
дом  
1september.ru

## Первое сентября

сентябрь  
2015

ИНФОРМАТИКА Подписка на сайте [www.1september.ru](http://www.1september.ru) или по каталогу «Почта России»: 79066 — бумажная версия, 12684 — CD-версия



НА ОБЛОЖКЕ

► “Поделись улыбкою своей, и она к тебе не раз еще вернется!”. Когда были написаны слова этой песни, был, по сути, лишь один вариант поделиться улыбкой. Улыбнуться. Очно, так сказать. Рискнем предположить, что в наши дни электронными улыбками — смайликами — делятся много чаще, чем “живыми”. Ну а если смайлики загораются, значит, это кому-нибудь нужно! :)

С Днем рождения, смайлик!

В НОМЕРЕ

- 3** ПАРА СЛОВ
  - Нет слов
- 4** УЧЕБНИКИ
  - Кукарача снова рулит (теория и практика вступления в программирование)
- 24** УГЛУБЛЕНКА
  - Применение карт взаимосвязи понятий учебного материала
  - Табличные формулы, или Действия над диапазонами
- 48** ЗАНИМАТЕЛЬНЫЕ МАТЕРИАЛЫ ДЛЯ ПЫТЛИВЫХ УЧЕНИКОВ И ИХ ТАЛАНТЛИВЫХ УЧИТЕЛЕЙ
  - “В мир информатики” № 210

В ЛИЧНОМ КАБИНЕТЕ

Облачные технологии от Издательского дома “Первое сентября”

Все подписчики журнала имеют возможность получать электронную версию, которая является полной копией бумажной. Для получения электронной версии:

1) Откройте Личный кабинет на портале “Первое сентября” ([www.1september.ru](http://www.1september.ru)).

2) В разделе “Газеты и журналы / Получение” выберите свой журнал и кликните на кнопку “Я — подписчик бумажной версии”.

3) Появится форма, посредством которой вы сможете отправить нам копию подписной квитанции.

После этого в течение одного рабочего дня будет активирована электронная подписка на весь период действия бумажной. Справки: [podpiska@1september.ru](mailto:podpiska@1september.ru) или через службу поддержки на портале “Первое сентября”.

# ИНФОРМАТИКА

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

<http://inf.1september.ru> Учебно-методический журнал для учителей информатики  
 Основан в 1995 г.  
 Выходит один раз в месяц

**РЕДАКЦИЯ:**  
 гл. редактор С.Л. Островский  
 редакторы  
 Е.В. Андреева,  
 Д.М. Златопольский  
 (редактор вкладки “В мир информатики”)  
 Дизайн макета И.Е. Лукьянов  
 верстка Н.И. Пронская  
 корректор Е.Л. Володина  
 секретарь Н.П. Медведева  
 Фото: фотобанк Shutterstock  
 Журнал распространяется по подписке  
 Цена свободная  
 Тираж 18 000 экз.  
 Тел. редакции: (499) 249-48-96  
 E-mail: [inf@1september.ru](mailto:inf@1september.ru)  
<http://inf.1september.ru>

**ИЗДАТЕЛЬСКИЙ ДОМ “ПЕРВОЕ СЕНТЯБРЯ”**  
**Главный редактор:**  
 Артем Соловейчик  
 (генеральный директор)  
**Коммерческая деятельность:**  
 Константин Шмарковский  
 (финансовый директор)  
**Развитие, IT и координация проектов:**  
 Сергей Островский  
 (исполнительный директор)  
**Реклама, конференции и техническое обеспечение Издательского дома:**  
 Павел Кузнецов  
**Производство:**  
 Станислав Савельев  
**Административно-хозяйственное обеспечение:**  
 Андрей Ушков  
**Педагогический университет:**  
 Валерия Арсланьян (ректор)

**ЖУРНАЛЫ ИЗДАТЕЛЬСКОГО ДОМА “ПЕРВОЕ СЕНТЯБРЯ”**  
 Английский язык – Е.Богданова  
 Библиотека в школе – О.Громова  
 Биология – Н.Иванова  
 География – и.о. А.Митрофанов  
 Дошкольное образование – Д.Тюттерин  
 Здоровье детей – Н.Сёмина  
 Информатика – С.Островский  
 Искусство – О.Волкова  
 История – А.Савельев  
 Классное руководство и воспитание школьников – М.Битянова  
 Литература – С.Волков  
 Математика – Л.Рослова  
 Начальная школа – М.Соловейчик  
 Немецкий язык – М.Бузоева  
 ОБЖ – А.Митрофанов  
 Русский язык – Л.Гончар  
 Спорт в школе – О.Леонтьева  
 Технология – А.Митрофанов  
 Управление школой – Е.Рачевский  
 Физика – Н.Козлова  
 Французский язык – Г.Чесновицкая  
 Химия – О.Блохина  
 Школа для родителей – Л.Печатникова  
 Школьный психолог – М.Чибисова

**УЧРЕДИТЕЛЬ:**  
 ООО “ИЗДАТЕЛЬСКИЙ ДОМ «ПЕРВОЕ СЕНТЯБРЯ»”  
**Зарегистрировано ПИ № ФС77-58447 от 25.06.2014 в Роскомнадзоре**  
 Подписано в печать: по графику 1.07.2015, фактически 1.07.2015  
 Заказ №  
 Отпечатано в ОАО “Первая Образцовая типография” Филиал “Чеховский Печатный Двор”  
 ул. Полиграфистов, д. 1, Московская область, г. Чехов, 142300  
 Сайт: [www.chpd.ru](http://www.chpd.ru)  
 E-mail: [sales@chpk.ru](mailto:sales@chpk.ru)  
 Факс: 8 (495) 988-63-76  
**АДРЕС ИЗДАТЕЛЯ:**  
 ул. Киевская, д. 24, Москва, 121165  
**Тел./факс:** (499) 249-31-38  
**Отдел рекламы:**  
 (499) 249-98-70  
<http://1september.ru>  
**ИЗДАТЕЛЬСКАЯ ПОДПИСКА:**  
**Телефон:** (499) 249-47-58  
**E-mail:** [podpiska@1september.ru](mailto:podpiska@1september.ru)



## Нет слов

Мне часто приходит на ум, что надо придумать какой-нибудь типографический знак, обозначающий улыбку, — какую-нибудь закорючку или упавшую навзничь скобку...

*Владимир Набоков. 1969*

► Официальным днем рождения смайлика считается 19 сентября 1982 г. Неформальный патент на изобретение улыбки Википедия выдает Скотту Фалману, американскому специалисту в области систем искусственного интеллекта. Не исключено, что он действительно был первым, кто предложил (и зафиксировал это в письме коллегам) использовать символы :-). Использовались ли они раньше — кто знает?

В последнее время феномену смайликов посвящено большое количество исследований в различных областях, прежде всего — в социальной психологии и меньше — в лингвистике. В значительном количестве работ авторы так или иначе апеллируют к результатам классических экспериментов Альберта Мерабиана, который, начиная с 1960-х годов, экспериментально анализировал коммуникационные процессы. В частности, именно он установил классическую статистику эффективности устной коммуникации. Эта статистика стала одной из наиболее часто цитируемых. Выглядит она так:

- 7% значения передается произносимыми словами;
- 38% значения — паралингвистическими средствами (тем, как эти слова произносятся, т.е. интонацией, ритмом и пр.);

- 55% процентов передается мимикой, пантомимикой и жестиком.

Тут, правда, надо сразу оговориться, что указанная экспериментально подтвержденная статистика получена в отношении устной коммуникации, не письменной! Кроме того, модель Мерабиана точно не следует трактовать упрощенно — ведь, если следовать ей буквально, эффективность коммуникации, отличной от устного разговора с визуальным компонентом, должна быть очень невелика. А это не так — люди общаются и по телефону, и письменно — и ничего, как-то справляются. Одно из предположений — за счет того, что дополнительную нагрузку берут на себя другие каналы. При разговоре по телефону, например, — собственно слова и есть паралингвистическая составляющая.

Но в любом случае модель Мерабиана явно указывает на причину, по которой смайлики стали столь популярными. Посредством них частично восполняются два выпадающих в письменной речи канала коммуникаций — паралингвистический и “мимическо-жестиком-ориентированный”.

Отдельный интересный вопрос — это взрыв в использовании смайликов, который произошел с распространением мессенжеров. В обычном письме выпадающие каналы легче компенсировать смысловым — надо просто длиннее и точнее писать. В экспресс-переписке же, которая по смыслу и назначению предельно короткая, смайлики стали значимым инструментом общения.

Ну, вот... Хотел написать что-то легкое и веселое, а получилось вполне себе занудно ☺.

*Сергей Островский,  
гл. редактор*



## Кукарача снова рулит (теория и практика вступления в программирование)

4

**А.А. Дуванов,**  
г. Переславль-Залесский,  
[kurs@robotland.pereslavl.ru](mailto:kurs@robotland.pereslavl.ru)

**Н.Д. Шумилина,**  
г. Тверь,  
[nshumilina@yandex.ru](mailto:nshumilina@yandex.ru)

► Курс информатики для начальной и средней школы “Азбука Роботландии” известен читателям “Информатики” по публикациям, обозначенным в конце этой заметки.

Мы продолжаем рассказ о третьей части Азбуки, начатый в последней статье “Азбука 3: роботландский зоопарк исполнителей”.

“Зоопарк” венчает старый добрый Кукарача, неплохо сохранившийся со второй половины 80-х годов прошлого века! Известно ведь, что тараканы (а, скажем откровенно, Кукарача и есть таракан) очень выносливые животные! Есть мнение, что даже ядерная зима Кукараче нипочем, а что там какие-то 30 лет развития школьной информатики!



Если уж вспомнилось о тех древних годах, то стоит показать читателю обложку самой первой книжки с описанием нашего Таракана. Эта книжка была издана в 1986 году в г. Благовещенске, ее авторы: А.А. Дуванов, В.В. Немилостива и О.Г. Какаулин.

В 1991 году Кукарача вошел в курс информатики для начальной школы с говорящим названием “Роботландия”.



Из истории Кукарачи нового века непременно надо вспомнить три книги издательства БХВ — “Азы программирования (ученик + задачник + учитель), написанные тремя авторами: А.А. Дуванов, А.В. Рудь, В.П. Семенко. Эти книги были изданы в 2005 году, и если постараться, их еще можно найти (по крайней мере в электронном виде).

Из обращения к читателю в книге для учителя:

*Мы с вами любим информатику, а особенно — что греха таить! — программирование.*

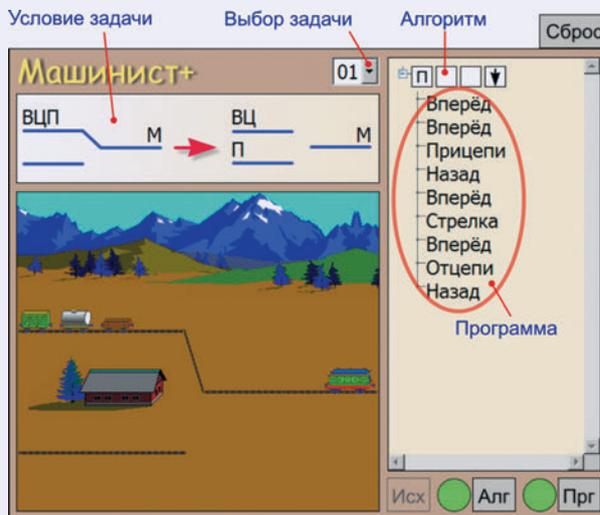
*Ведь программирование — это математика информатики: “ум в порядок приводит”, и ее музыка доставляет изысканное наслаждение!*

*Мы с вами, конечно, вкусили, вкушаем и, надо думать, будем вкушать яблочки с этого дерева, пусть не самые крупные, но не менее сладкие и полезные, насыщенные витамином увлеченности и азарта.*

*К сожалению, в последнее время интерес молодежи к программированию угасает. Как жаль! Ведь программирование — не только замена пустому времяпрепровождению. Программирование — это солидный багаж для вступления в успешную жизнь. Спрос на программистов только растет. И получают они за свою работу хорошие деньги. Гораздо большие, чем простые компьютерные пользователи.*

В “Азбуке Роботландии” Кукарача появляется после того, как ученики познакомились с поня-

тием “исполнитель”, поработали с разными исполнителями в командном режиме и освоили программное управление исполнителем Машинист+ :



Программа для Машиниста+ собирается при помощи контекстного меню из команд его СКИ (система команд исполнителя).

Программа для Кукарачи записывается в виде текста в текстовом редакторе с использованием процедурного языка программирования. То есть Кукарача открывает ученику мир настоящего “взрослого” программирования!

При помощи этого исполнителя решаются следующие образовательные задачи:

- 1. Формирование алгоритмического мышления** при помощи построения программ на языке программирования, содержащего основные алгоритмические структуры: процедуры, ветвления, повторения, условные циклы. Формализм языка программирования играет такую же роль, что и формулы в математике, — сформировать математическое мышление без абстракций невозможно, равно как невозможно сформировать алгоритмическое мышление без записи алгоритма в виде программы на языке программирования. Можно много говорить о важности математического мышления, но пока школьник на практике не пройдет через вычисление гипотенузы по формуле Пифагора, все слова будут красивы, но бесполезны. Так и в информатике. Можно много говорить об алгоритмическом мышлении, приводить примеры с алгоритмами перехода улицы и заварки чая, но пока школьник не напишет код с ветвлениями, циклами и рекурсией для конкретного формального исполнителя, не пройдет через этап тестирования и отладки, разговор об алгоритмическом мышлении будет красив, но бесполезен.

- 2. Освоение метода нисходящего процедурного проектирования** — эффективного метода преодоления сложностей, возникающих при решении больших задач в любых областях человеческой деятельности. В основу этой технологии положен

принцип иерархической нисходящей стратегии “от корня к листьям” — от проектирования крупных блоков к описаниям составляющих частей.

3. **Освоение метода восходящего проектирования** — он дополняет метод нисходящей разработки, когда есть готовые “детали”, которые можно (или нужно) использовать в новом проекте.

4. **Освоение принципов тестирования и отладки построенных решений:** проектирование оптимального набора тестов, позволяющих за приемлемое время наиболее полно проверить полученное решение, не забывая, помимо “типовых” наборов входных данных, рассматривать крайние, исключительные, не типичные, но допустимые варианты. Тестирование и отладка — это часть цепочки работы над проектом, цель которого — создать продукт, обладающий заданными свойствами. Цепочка начинается с формализации описания продукта, заданного заказчиком, превращения “художественного” текста в формальную постановку задачи. Следующий этап — построение модели, затем — разработка алгоритма, наконец — программирование. Но это не последний этап разработки! Начинается процесс отладки (тестирование и исправление ошибок), анализ результатов, уточнение модели и сопровождение программы (улучшение потребительских свойств, оптимизация и устранение дефектов после передачи продукта в эксплуатацию). Формализации и моделированию будет посвящено значительное место в следующей, четвертой части “Азбуки” (решение “постановочных” задач), в “Азбуке 3” основное внимание в цепочке проектирования уделяется разработке алгоритма, программированию, тестированию и отладке.

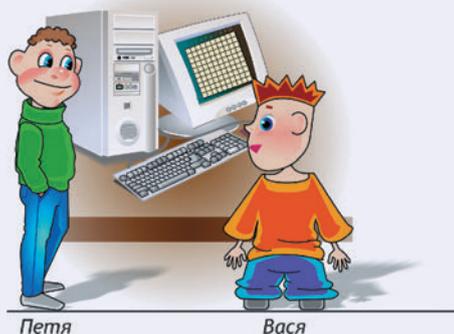
5. **Освоение рекурсивного мышления.** Рекурсия — мощный инструмент математики и, конечно, информатики. Часто рекурсивные решения настолько легки и красивы, насколько тяжелы и громоздки попытки обойтись циклами. Освоение рекурсии базируется не столько на освоении технологии, сколько на формировании специфического “рекурсивного” мышления, и этому вопросу “Азбука” уделит значительное место в четвертой части. Благодаря рекурсии Кукарачу можно научить выполнять подсчеты (если вы забыли удивиться в этом месте, то напоминаем, что в языке программирования Кукарачи нет переменных). Вы не поверите, но с помощью рекурсии в среде Кукарачи можно решать даже задачи лексического анализа текста (лексический анализатор — составная часть такого сложного программного продукта, как транслятор). Удивительные глубины скрыты в простецком на первый взгляд Кукараче!

6. **Формирование абстрактного и логического мышления** при решении “хитрых” задач.

Как живет Кукарача на страницах “Азбуки 3”? Давайте посмотрим!

## Старый знакомый

Приятно видеть, как начинают оживать старые герои! Прислушайтесь: с запыленных роботландских страниц по-прежнему доносятся голоса двух братьев. Что же так горячо обсуждают ребята?



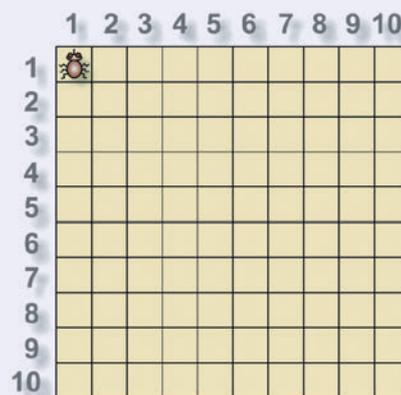
Петя

Вася

**Вася.** Петя, расскажи, пожалуйста, о Кукараче подробнее!

**Петя.** Кукарача может ползать по клетчатому полю и собирать из кубиков разные слова. Что ты видишь на экране?

**Вася.** Я вижу клетчатое поле и на нем одинокое существо в левом верхнем углу.



**Петя.** Это и есть среда исполнителя. А в углу в ожидании указаний стоит сам исполнитель — Кукарача. Вот его СКИ.



СКИ

ВПРАВО

ВЛЕВО

ВВЕРХ

ВНИЗ

СТОЯТЬ

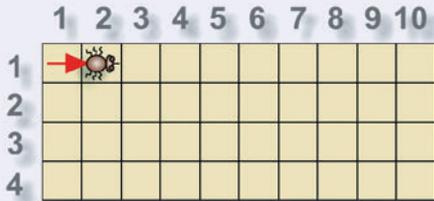
По командам **ВЛЕВО**, **ВПРАВО**, **ВВЕРХ**, **ВНИЗ** исполнитель переползает в соседнюю клетку в указанном направлении. **СТОЯТЬ** — “пустая” команда, исполнитель остается на месте.

**Вася.** А как задавать команды исполнителю?

**Петя.** Команды надо записывать в редакторе команд и нажимать экранную кнопку запуска, изображенную в виде зеленого треугольника.



**Вася.** Хорошо, напишу команду **ВПРАВО** и нажму кнопку запуска... Есть! Кукарача послушно сместился на одну клетку вправо.



А теперь скамандую **ВВЕРХ**. Ага, хитрый исполнитель даже не пошевелил лапками, он написал: Не могу!



**Петя.** Кукарача может сталкивать с поля кубики, но сам покинуть свою среду не может. Вот и сообщает об этом.

**Вася.** Но на поле нет никаких кубиков!

**Петя.** Их можно поставить. “Кубик” — это клетка, в которую записана буква, цифра или другой символ. Из кубиков можно складывать слова.



**Вася.** Я дал команду толкнуть кубик вверх, а исполнитель не понял!



**Петя.** Ты записал слово с ошибкой. Команды ВЕРХ нет в СКИ, Кукарача не понимает, что надо делать!

## Способы управления Кукарачей

Как известно, исполнители могут работать в двух режимах:

- командном;
- программном.

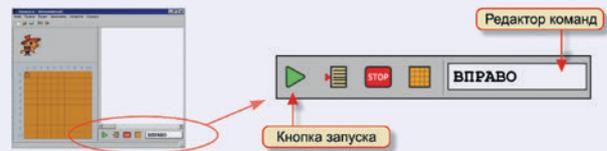
В командном режиме исполнитель получает команду, выполняет ее и ждет следующую.

В программном режиме исполнитель получает не одну команду, а целый список команд — программу и выполняет ее.

Кукарача может работать как в командном режиме, так и в программном.

### Командный режим

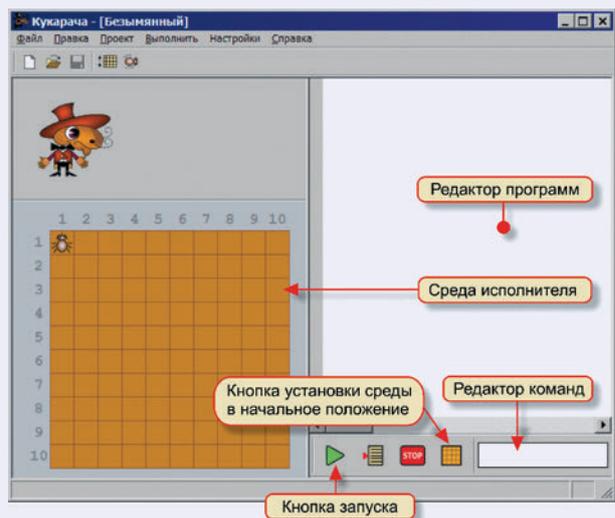
Команды записываются в редакторе команд и передаются исполнителю нажатием на кнопку запуска:



Исполнитель получает команду, выполняет ее и ждет следующую.

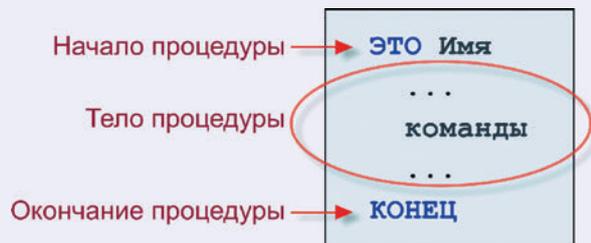
### Программный режим

Программа для Кукарачи записывается в **редакторе программ**, который (как и редактор команд) входит в состав среды разработки, дополняющей среду исполнителя. Все это существует в виде компьютерного приложения (работает в OS Windows):



Редактор программ настроен на разработку программ для Кукарачи: автоматически подсвечивает ключевые слова языка и команды из СКИ, предлагает контекстное меню для выбора языковых конструкций (сниппеты).

Программа состоит из процедур. Каждая процедура имеет вид, показанный на рисунке:



Чтобы запустить последовательность команд в процедуре на выполнение, нужно написать имя процедуры в редакторе команд и нажать экранную кнопку запуска:



Другой способ запуска процедуры — использование ее имени в качестве команды внутри другой процедуры.

Поясним сказанное на следующем примере.

**Задача.** Уж не нужен

Кукарача расположен где-то в первой строке своего поля. Справа от него плотно стоят два кубика. Удалить кубики с поля и вернуть Кукарачу на прежнее место. Пример возможного начального состояния среды показан на рисунке:

	1	2	3	4	5	6	7	8	9	10
1				🐻	У	Ж				
2										
3										
4										
5										

Запишем такую программу:

```
// Решение задачи "Уж не нужен"
ЭТО Уж
    Удаление // Удаляем кубики с поля
    Возврат  // Возвращаем исполнителя
              // на прежнее место
КОНЕЦ

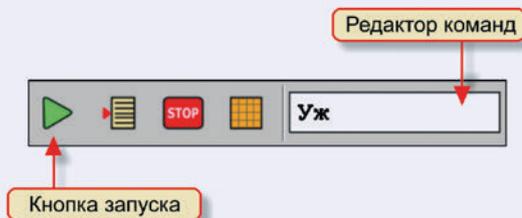
// Удаляем кубики с поля
ЭТО Удаление
    Удалить_кубик // Подходим к кубику У
                  // и удаляем его
    Удалить_кубик // Подходим к кубику Ж
                  // и удаляем его
КОНЕЦ

// Подходим к кубику и удаляем его
ЭТО Удалить_кубик
    ВНИЗ ВПРАВО // Подходим к кубику
    ВВЕРХ       // и удаляем его
КОНЕЦ
```

```
// Возвращаем исполнителя на прежнее место
ЭТО Возврат
    ВЛЕВО ВЛЕВО // Смещаем исполнителя
                // на две клетки влево
КОНЕЦ
```

Чтобы записанная программа выполнялась, нужно написать в редакторе команд имя *главной* (начальной, стартовой) процедуры, с которой нужно начать выполнение, и нажать кнопку запуска.

В нашем случае начальной является процедура Уж:



Во время своей работы процедура Уж сначала запустит на выполнение процедуру Удаление, а затем процедуру Возврат.

Работа процедуры Удаление сводится к выполнению двух процедур Удалить\_кубик, а работа процедуры Возврат — к выполнению двух команд ВЛЕВО из СКИ.

Запись программы рекомендуется сопровождать комментариями. Комментарий начинается с двух символов “//” и продолжается до конца строки. Комментарии не порождают никаких команд исполнителю, их можно располагать в любом месте программы, как внутри процедур, так и между ними.

**Процедура — это команда, дополняющая СКИ**

В СКИ исполнителя пять “низкоуровневых” команд: **ВПРАВО**, **ВЛЕВО**, **ВВЕРХ**, **ВНИЗ**, **СТОЯТЬ**.

Процедура — это возможность расширить СКИ новой командой. В решении задачи Уж не нужен мы объяснили исполнителю, как выполнять новую команду Уж:

```
ЭТО Уж
    Удаление
    Возврат
КОНЕЦ
```

Но в нашем объяснении мы снова использовали новые команды — Удаление и Возврат.

Конечно, Кукарача не сможет выполнить команду Уж, ведь команд Удаление и Возврат нет в его СКИ. Нам приходится объяснять исполнителю, как он должен выполнять и эти новые команды. Эти объяснения формально всегда укладываются в конструкцию: **ЭТО . . . КОНЕЦ**. Эта конструкция называется *описанием процедуры*. Если процедура описана, ее имя можно использовать как команду в редакторе команд или в теле других процедур.

Команда Уж запускается на выполнение через редактор команд, а команды Удаление и Возврат запускаются при выполнении команды Уж.

## Язык программирования Кукарачи

Язык Кукарачи весьма прост! В нем нет сложных конструкций, нет переменных и, соответственно, нет команды присваивания! Но алгоритмики в нем хватает! Мы это ясно увидим, решая задачи. Программа для Кукарачи — это набор описаний процедур. Процедуру можно выполнить, если ее имя использовать как команду, расширяющую набор команд из СКИ. Ниже приводится краткое описание всех команд языка программирования Кукарачи.

Команда языка	Как выполняется
<команда из СКИ>	Команда из СКИ исполнителя
<имя процедуры>	Команда-процедура (вызов процедуры). Выполняется процедура с указанным именем
<b>ПОВТОРИ</b> <число> <команда>	Цикл <b>ПОВТОРИ</b> . Повторение выполнения команды указанное число раз
<b>ПОКА</b> <условие> <команда>	Цикл <b>ПОКА</b> . Повторение выполнения команды, пока условие имеет значение <b>истина</b> . Проверка условия — перед выполнением команды
<b>ЕСЛИ</b> <условие> <b>ТО</b> <команда1> <b>ИНАЧЕ</b> <команда2>	Команда <b>ветвления</b> . Проверяется <b>условие</b> и выполняется либо <b>команда1</b> (при значении условия <b>истина</b> ), либо <b>команда2</b> (при значении условия <b>ложь</b> ). Часть <b>ИНАЧЕ</b> <команда2> может быть опущена
{ <список команд> }	Составная команда. Объединение нескольких команд в одну

Кукарача способен выполнять в своей среде проверки, задаваемые условиями, входящими в состав конструкций **ЕСЛИ** и **ПОКА**. Множество всех условий, допустимых в языке, приводится в следующей ниже таблице (запись <символ> обозначает шаблон, на месте которого может быть любой символ).

<символ>	Истина, если Кукарача толкнул кубик с указанным в условии символом, <b>ложь</b> в противном случае
<b>НЕ</b> <символ>	<b>Ложь</b> , если Кукарача толкнул кубик с указанным в условии символом, <b>истина</b> в противном случае
<b>ПУСТО</b>	Истина, если клетка, в которую сместился Кукарача, пуста, <b>ложь</b> в противном случае
<b>НЕ ПУСТО</b>	<b>Ложь</b> , если клетка, в которую сместился Кукарача, пуста, <b>истина</b> в противном случае
<b>ЦИФРА</b>	Истина, если Кукарача толкнул кубик с цифрой (один из символов 0, 1, 2, 3, 4, 5, 6, 7, 8, 9), <b>ложь</b> в противном случае
<b>НЕ ЦИФРА</b>	<b>Ложь</b> , если Кукарача толкнул кубик с цифрой (один из символов 0, 1, 2, 3, 4, 5, 6, 7, 8, 9), <b>истина</b> в противном случае

## Нисходящее процедурное программирование

Дробление программы на процедуры выполняем по принципу функциональности *методом нисходящей разработки программ* (сверху вниз; от общего к частному).

Материалы учебника и методички последовательно развивают иерархический стиль разработки: сначала проектируем крупные блоки, потом составляющие их более мелкие части, постепенно доходя до “винтиков” — низкоуровневых команд из СКИ.

В силу этой идеологии, в главной процедуре стараемся *не записывать* команды из СКИ, а формулируем код крупными командами-процедурами, отражающими логические действия (шаги алгоритма).

Пусть, например, нужно пройти лабиринт и установить исполнителя в заданную клетку. Останавливаем детей, которые начинают писать код командами из СКИ.

Согласно алгоритму решения, записываем две логические команды, подбирая для них “говорящие” имена:

**ЭТО** Лабиринт

Пройти\_лабиринт

В\_клетку\_назначения

**КОНЕЦ**

Замечаем далее, что траекторию движения исполнителя внутри лабиринта можно построить (например) из трех одинаковых частей:

**ЭТО** Пройти\_лабиринт

ПОВТОРИ 3 Пройти\_часть

**КОНЕЦ**

Команду Пройти\_часть описываем командами из СКИ:

**ЭТО** Пройти\_часть

...

**КОНЕЦ**

Приступая к описанию процедуры `В_клетку_назначения`, обнаруживаем (например), что ее код содержит всего одну команду из СКИ:

**ЭТО В\_клетку\_назначения**

**ВПРАВО**

**КОНЕЦ**

Не станем этим смущаться! И не будем заменять командой **ВПРАВО** команду `В_клетку_назначения` в главной процедуре. Ведь запись **ВПРАВО** не отражает логику действия (переместить исполнителя в клетку назначения).

Бывают и отступления от нисходящего метода, например, при использовании процедур, написанных ранее, но принцип логической функциональности процедур должен работать всегда. Не надо делить программу на процедуры механическим делением на части по объему кода.

### Таблицу в список. Мастер-класс Кукарачи

**Задача.** *Таблицу в список*

Из Роботландии получено задание: преобразовать таблицу в список и закончить работу в клетке (10,10):

**Дано**

1	2	3	4	5	6	7	8	9	10
1		1	2	3					
2		4	5	6					
3		7	8	9					
4									
5									
6									
7									
8									
9									
10									

**Надо**

1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9
2									
3									
4									
5									
6									
7									
8									
9									
10									

### Решение

Покажем на этой задаче, как быстро и правильно строить программы в среде Кукарачи.



Начинаем, конечно, с записи алгоритма.

1. Сдвинем строки таблицы по горизонтали:

1	2	3	4	5	6	7	8	9	10
1		1	2	3					
2				4	5	6			
3							7	8	9
4									

2. Установим кубики в первую строку:

1	2	3	4	5	6	7	8	9	10	
1		1	2	3	4	5	6	7	8	9
2										
3										
4										

3. Поставим исполнителя в клетку (10,10).

В соответствии с алгоритмом запишем программу в среде Кукарачи:

**ЭТО Список**

```
Подготовка // Сдвинем строки
              // по горизонтали
Установка   // Установим кубики
              // в первую строку
На_место    // Поставим исполнителя
              // в клетку (10,10)
```

**КОНЕЦ**

// Сдвинем строки по горизонтали

**ЭТО Подготовка**

**КОНЕЦ**

// Установим кубики в первую строку

**ЭТО Установка**

**КОНЕЦ**

// Поставим исполнителя в клетку (10,10)

**ЭТО На\_место**

**КОНЕЦ**

Главная процедура `Список` содержит три команды, не входящие в СКИ: `Подготовка`, `Установка`, `На_место`. Эти команды нужно объяснить Кукараче при помощи одноименных процедур.

Очень важно сразу записывать эти, пустые пока, *процедуры-заглушки* — это позволит в любой момент запускать код, не получая аварийного сообщения. Не описана процедура.

Продолжаем работу. Проектируем команду-процедуру `Подготовка`:

// Сдвинем строки по горизонтали

**ЭТО Подготовка**

Строка\_2

Строка\_3

**КОНЕЦ**

```

ЭТО Строка_2
  ВНИЗ
  ПОВТОРИ 3 ВПРАВО
  ПОВТОРИ 3 ВЛЕВО
КОНЕЦ

```

```

ЭТО Строка_3
КОНЕЦ

```

Запускаем программу **Список** (пустые процедуры-заглушки позволят нам это сделать). Среда после завершения работы процедуры **Строка\_2** выглядит так:

	1	2	3	4	5	6	7	8	9	10
1		1	2	3						
2					4	5	6			
3		7	8	9						
4										

Кукарача стоит в том положении, в котором его “возьмет” процедура **Строка\_3**, код которой нам теперь совершенно ясен:

```

// До: Кукарача в клетке (2,1)
ЭТО Строка_3

```

```

  ВНИЗ
  ПОВТОРИ 6 ВПРАВО
КОНЕЦ

```

Снова запускаем программу **Список** и видим состояние среды и положение исполнителя после выполнения написанного кода. Среда после завершения работы процедуры **Строка\_3**:

	1	2	3	4	5	6	7	8	9	10
1		1	2	3						
2					4	5	6			
3								7	8	9
4										

Это состояние среды и исполнителя является начальным для процедуры **Установка**, для написания которой наступил черед. Запишем:

```

// Установим кубики в первую строку
// До: Кукарача в клетке (3,7)
ЭТО Установка

```

```

  Установка_строки_2
  Установка_строки_3
КОНЕЦ

```

```

// До: Кукарача в клетке (3,7)
ЭТО Установка_строки_2
  ПОВТОРИ 3 Шаг1
КОНЕЦ

```

```

ЭТО Шаг1
  ВВЕРХ ВНИЗ ВЛЕВО
КОНЕЦ

```

```

ЭТО Установка_строки_3
КОНЕЦ

```

Запускаем программу и видим состояние среды и положение исполнителя перед входом в процедуру **Установка\_строки\_3**:

	1	2	3	4	5	6	7	8	9	10
1		1	2	3	4	5	6			
2										
3								7	8	9
4										

Код процедуры **Установка\_строки\_3** становится очевидным:

```

// До: Кукарача в клетке (3,4)
ЭТО Установка_строки_3

```

```

  Подход
  ПОВТОРИ 3 Шаг2
КОНЕЦ

```

```

ЭТО Подход
  ПОВТОРИ 3 ВПРАВО
  ВНИЗ
КОНЕЦ

```

```

ЭТО Шаг2
  ВПРАВО
  ПОВТОРИ 2 ВВЕРХ
  ПОВТОРИ 2 ВНИЗ
КОНЕЦ

```

Запускаем программу, видим состояние среды и положение исполнителя перед входом в последнюю процедуру **На\_место**:

	1	2	3	4	5	6	7	8	9	10
1		1	2	3	4	5	6	7	8	9
2										
3										
4										

Окончательный код программы:

```

ЭТО Список
  Подготовка // Сдвинем строки
               // по горизонтали
  Установка // Установим кубики
               // в первую строку
  На_место // Поставим исполнителя
               // в клетку (10,10)

```

```

КОНЕЦ

// Сдвинем строки по горизонтали
ЭТО Подготовка
  Строка_2
  Строка_3
КОНЕЦ

```

```

// До: Кукарача в клетке (1,1)
ЭТО Строка_2
  ВНИЗ

```

```

ПОВТОРИ 3 ВПРАВО
ПОВТОРИ 3 ВЛЕВО
КОНЕЦ

// До: Кукарача в клетке (2,1)
ЭТО Строка_3
  ВНИЗ
  ПОВТОРИ 6 ВПРАВО
КОНЕЦ

// Установим кубики в первую строку
// До: Кукарача в клетке (3,7)
ЭТО Установка
  Установка_строки_2
  Установка_строки_3
КОНЕЦ

// До: Кукарача в клетке (3,7)
ЭТО Установка_строки_2
  ПОВТОРИ 3 Шаг1
КОНЕЦ

ЭТО Шаг1
  ВВЕРХ ВНИЗ ВЛЕВО
КОНЕЦ

// До: Кукарача в клетке (3,4)
ЭТО Установка_строки_3
  Подход
  ПОВТОРИ 3 Шаг2
КОНЕЦ

ЭТО Подход
  ПОВТОРИ 3 ВПРАВО
  ВНИЗ
КОНЕЦ

ЭТО Шаг2
  ВПРАВО
  ПОВТОРИ 2 ВВЕРХ
  ПОВТОРИ 2 ВНИЗ
КОНЕЦ

// Поставим исполнителя в клетку (10,10)
// из положения в клетке (4,10)
ЭТО На_место
  ПОВТОРИ 6 ВНИЗ
КОНЕЦ

```

Этот код выглядит довольно внушительно, но благодаря программированию крупными командами-процедурами и промежуточным запускам кода, в котором записаны пустые процедуры-заглушки, результат получился как бы сам собой. Создавать программы таким образом не сложно.



## Стиль записи программ

Разделителем слов в языке программирования роботландских исполнителей служит пробел или конец строки. Посмотрите на такую запись программы:

```

ЭТО Выход Поиск_прохода На_край_доски
КОНЕЦ ЭТО Поиск_прохода ВПРАВО ПОКА НЕ
ПУСТО Шаг КОНЕЦ ЭТО Шаг ВЛЕВО ВНИЗ ВПРАВО
КОНЕЦ ЭТО На_край_доски ПОВТОРИ 8 ВПРАВО
КОНЕЦ

```

С точки зрения интерпретатора — это правильный код, и он будет успешно выполнен. Человек же читать плотно набитые строки не привык. Совсем другое дело, когда тот же самый код записывается так:

```

// Задача. Найти проход в стене и дойти
// до края поля (в десятом столбце)

```

```

ЭТО Выход
  Поиск_прохода
  На_край_доски
КОНЕЦ

ЭТО Поиск_прохода
  ВПРАВО
  ПОКА НЕ ПУСТО Шаг
КОНЕЦ

```

```

ЭТО Шаг
  ВЛЕВО // Вернуться в первый столбец
  ВНИЗ // Шагнуть на следующую строку
  ВПРАВО // Толкнуть кубик
КОНЕЦ

```

```

ЭТО На_край_доски
  ПОВТОРИ 8 ВПРАВО
КОНЕЦ

```

Что изменилось? Последний вариант кода содержит пустоты, структурирован и снабжен комментариями.

### Пустоты

Человек способен воспринимать информацию только маленькими порциями.

Поэтому обычные тексты в книгах не покрывают сплошь всю страницу. Пустоты помогают фиксировать внимание на небольших фрагментах, улучшая тем самым информационное пищеварение.

А ведь программы похожи на текст, они пишутся в текстовых редакторах, они выглядят как текст, они и есть текст!

В программном коде полезно отделять пустыми строками процедуры и в каждой строке записывать не более одной команды. Последнее правило можно нарушать лишь тогда, когда команды, записанные в одной строке, выполняют отдельный шаг алгоритма.

## Структурирование

Работая с отдельным фрагментом, необходимо понимать его место в общей ткани повествования. В силу этого книгу делят на части, части — на главы, главы — на параграфы, внутри параграфа записывают абзацы. Абзац содержит предложения, предложение — слова. Таким образом, наше восприятие настраивается на иерархическую структуру материала, связывая каждое слово с полным иерархическим деревом.

В программном коде выделить иерархическую структуру помогают отступы от левого края (структурная лесенка). Это общепринятое правило записи иерархических конструкций, и студенты Азбуки осваивали запись иерархий лесенкой в предыдущей части курса.

Иерархия, показанная на рисунке, представлена графически в виде дерева:



В текстовой записи эта иерархия выглядит так:

```
Личность
  фамилия
  Возраст
  Адрес
    Город
    Улица
    Дом
```

Смещение вправо означает переход от родителя к потомку.

Тот же принцип лежит в основе структурной лесенки в записи программных кодов. Ключевые слова **ЭТО** и **КОНЕЦ** записываются с первой позиции. Команды, составляющие тело процедуры и являющиеся ее прямыми потомками, размещаются со смещением (на две позиции) вправо.

Когда команда языка сама имеет иерархическую структуру, поступают подобным образом:

```
ЕСЛИ А
  ТО ВПРАВО
  ИНАЧЕ ВЛЕВО
```

Ветви **ТО** и **ИНАЧЕ** являются потомками команды **ЕСЛИ** (входят в нее).

В приведенном примере команды, расположенные на ветвях условной конструкции (**ВПРАВО** и **ВЛЕВО**), выровнены по вертикали, что дополнительно улучшает дизайн кода.

Отход от структурной лесенки допустим только в компактных конструкциях, которые лучше воспринимаются в одной строке:

```
ПОВТОРИ 8 ВНИЗ
ЕСЛИ ПУСТО ТО ВПРАВО
ПОКА НЕ А ВНИЗ
```

## Комментарии

Роль комментариев трудно переоценить. Обычно программирование сводится к записи шагов алгоритма на русском языке с последующим переводом их в конструкции языка программирования. При этом описания шагов алгоритма сохраняются в тексте программы в виде комментариев. Дополнительные комментарии могут пояснять отдельные важные или “тонкие” места в программе. Нужно сразу приучать учеников к комментариям. Труд, потраченный на их ввод, окупается экономией времени при отладке и сопровождении программы.

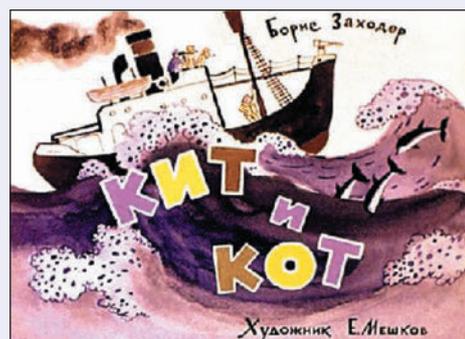
Комментарии не нужны лишь тогда, когда имена процедур содержательно их заменяют (“говорящие имена”).

## Команда ветвления

### Задача. Звери

Помогите Кукараче собрать два слова: название животного и место, где этому животному нравится жить. На перевернутом кубике (на рисунке он изображен знаком вопроса) или **О**, или **И**.

	1	2	3	4	5	6	7	8	9	10
1			О				К			
2			К				Р			
3			Е				Ы			
4			А				Ш			
5				Н	?	А				
6				К		Т				
7										
8										
9										
10										



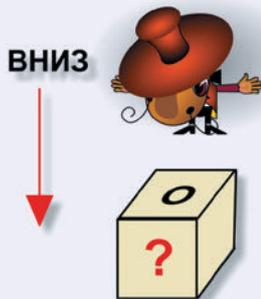
**Вася.** Я понял! Если это буква **О**, то надо собрать слова **КОТ** и **КРЫША**, а если буква **И** — **КИТ** и **ОКЕАН**. Давай перевернем кубик и посмотрим, что на нем написано.

**Петя.** Считаю, что поле Кукарачи недоступно. Например, оно подвергнуто радиоактивному заражению. Кукарача — робот, и он может там работать, а человек — нет. Но можно управлять исполнителем при помощи радио и компьютера.

**Вася.** Может ли исполнитель как-нибудь узнать, что написано на кубике?

**Петя.** Может. Когда Кукарача толкает кубик, кубик переворачивается, и Кукарача видит надпись.

**Вася.** Ну, тогда напишу команду **ВНИЗ**. Кукарача толкнет кубик, и я увижу надпись. После этого буду знать, как управлять исполнителем дальше.



**Петя.** Хотелось бы, чтобы робот работал автоматически без твоего присутствия, непрерывно, по заранее написанной программе. Представь, что Кукарачу отправили на Марс, связи с ним нет, и увидеть, как он переворачивает кубики, нельзя.

**Вася.** Даже если бы связь с Марсом была, она не быстрая, ведь сигнал от Марса идет так долго...

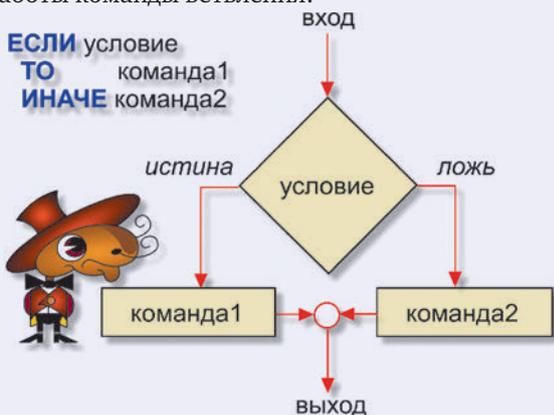
**Петя.** Верно! А роботу нужно принимать решение немедленно, он не должен ждать с Земли твоих сигналов.

**Вася.** Но я не знаю, как написать программу, которая бы принимала решение за меня!

**Петя.** Правильно, не знаешь. Это потому, что ты не знаком с командой ветвления. Посмотри, как она записывается:

```
ЕСЛИ условие
ТО команда1
ИНАЧЕ команда2
```

Слова **ЕСЛИ**, **ТО**, **ИНАЧЕ** — ключевые слова языка программирования. Они как цемент связывают команду ветвления воедино. Работает новая команда так: интерпретатор программ проверяет *условие* и, если условие верное, *то* выполняет первую команду, *иначе* (если условие ложное) — вторую. В любом случае работает только одна команда, либо **команда1**, либо **команда2**. Схема работы команды ветвления:



**Вася.** А как записывается условие?

**Петя.** Обычно в качестве условия записывают символ (букву, цифру или другой знак). Интерпретатор будет сравнивать этот символ с тем символом, который увидел Кукарача на перевернутом кубике.

**Вася.** Попробую написать программу для нашей задачи.

```
// Программа для местности,
// зараженной радиацией
ЭТО Звери
// Толкнуть кубик. Кубик перевернется,
// и Кукарача увидит надпись
ВНИЗ
ЕСЛИ И // На кубике буква И?
ТО ВЛЕВО // На кубике И —
// собираем ОКЕАН
ИНАЧЕ ВПРАВО // На кубике О —
// собираем КРЫША
```

**КОНЕЦ**

**Петя.** Написано правильно. Теперь расскажи, как интерпретатор выполнит эту программу.

**Вася.** Сначала он передаст исполнителю команду **ВНИЗ**. Кукарача толкнет кубик и увидит надпись на нем. А дальше...

**Петя.** Дальше интерпретатор, выполняя команду **ЕСЛИ**, пошлет Кукараче запрос: “Что на кубике?” Кукарача ответит, и в зависимости от ответа интерпретатор передаст исполнителю либо команду **ВЛЕВО**, либо команду **ВПРАВО**.



**Вася.** То есть интерпретатор будет работать за меня? Он спросит исполнителя, что тот видит на кубике, а затем, в соответствии с командой ветвления, задаст ему ту или иную работу.

**Петя.** Да. Только интерпретатор, как и Кукарача, не боится радиации. Он может даже на Марсе выполнять ту работу, которую ты запрограммировал в своем уютном кресле.

**Вася.** Вот она — сила программирования вообще и робототехники в частности!

**Петя.** Как ты думаешь, сколько запусков программы **Звери** нужно сделать, чтобы убедиться, что программа работает правильно?

**Вася.** Думаю, что два. Сначала надо проверить программу, когда на перевернутом кубике записана буква **О**, и второй раз, когда на перевернутом кубике — буква **И**.

## Экран

Простекие и скучноватые задачки у Кукарачи. Вы уверены? Ну, вот вам головоломочка про экран.

### Задача. Экран

Во второй строке злоумышленник перемешал буквы в слове ЭКРАН. Восстановить испорченное слово.



	1	2	3	4	5	6	7	8	9	10
1										
2	#	?	?	?	?					
3										
4										

### Решение

#### Описание алгоритма

Решение основано на следующей идее: переместить буквы в строки 3–7, а затем собрать слово в десятом столбце. При этом буква Э перемещается в третью строку, буква К — в четвертую, и так далее, наконец, буква Н перемещается в седьмую строку. Таким образом, независимо от того, как стояли буквы изначально, они будут стоять в правильном порядке по строкам. Остается только сместить их в один столбец.

Идею решения демонстрирует следующая схема (Кукарача изображен символом “#”):

1	.....	.....	.....	.....
2	#КРЭНА....	.....#.....	#.....	.....
3	.....	...Э.....	...Э.....	.....Э
4	.....	-> .К.....	-> .К.....	-> .....К
5	.....	..Р.....	..Р.....	.....Р
6	.....	...А.....	...А.....	.....А
7	.....	...Н.....	...Н.....	.....Н

### Программа

ЭТО Экран

```
ПОВТОРИ 5 Установить_букву // в нужную
// строку
```

```
Исполнитель_в_первый_столбец
```

```
Установка_слова // в 10-м столбце
```

КОНЕЦ

...

Полный код не приводим — он не сложен, а идея рассказана!

### Тесты

Для проверки именно этого алгоритма достаточно запустить программу один раз с произвольным расположением букв Э, К, Р, А, Н на кубиках — ведь для этого алгоритма безразлично, на каком кубике какая буква находится.

Для проверки же всех возможных вариантов начальных состояний пришлось бы выполнить  $5! = 120$  проверочных запусков.

## Переключатель

### Задача. Установка цифры

Под Кукарачей, во второй строке расположена цифра. Ее значение больше двух. Определить, какая это цифра, и поставить ее в строку с соответствующим номером.

	1	2	3	4	5	6	7	8	9	10
1	#									
2	?									
3										
4										

### Решение

ЭТО Установка\_цифры

```
ВНИЗ
```

```
ЕСЛИ 4
```

```
ТО
```

```
ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 5
```

```
ТО ПОВТОРИ 2 ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 6
```

```
ТО ПОВТОРИ 3 ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 7
```

```
ТО ПОВТОРИ 4 ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 8
```

```
ТО ПОВТОРИ 5 ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 9
```

```
ТО ПОВТОРИ 6 ВНИЗ
```

КОНЕЦ

Вася. Выглядит жутковато.

Петя. Обычно такие конструкции (их называют переключателями) записывают так:

ЭТО Установка\_цифры

```
ВНИЗ
```

```
ЕСЛИ 4 ТО ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 5 ТО ПОВТОРИ 2 ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 6 ТО ПОВТОРИ 3 ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 7 ТО ПОВТОРИ 4 ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 8 ТО ПОВТОРИ 5 ВНИЗ
```

```
ИНАЧЕ ЕСЛИ 9 ТО ПОВТОРИ 6 ВНИЗ
```

КОНЕЦ

Вася. Да, это совсем другое дело.

Петя. Хотя в это и трудно поверить, но в последней программе записано то же самое, что и в предыдущей, но код по-другому отформатирован, то есть по другим правилам сформированы строчки и расставлены пробелы.

Дополним беседу братьев еще одним замечанием. Сравните фрагменты кода в двух записях:

Запись 1	Запись 2
ЭТО Цифру_в_нужную_строку	ЭТО Цифру_в_нужную_строку
ЕСЛИ 3 ТО Цифра_3	ЕСЛИ 3 ТО Цифра_3
ИНАЧЕ ЕСЛИ 4 ТО Цифра_4	ЕСЛИ 4 ТО Цифра_4
ИНАЧЕ ЕСЛИ 5 ТО Цифра_5	ЕСЛИ 5 ТО Цифра_5
ИНАЧЕ ЕСЛИ 6 ТО Цифра_6	ЕСЛИ 6 ТО Цифра_6
ИНАЧЕ ЕСЛИ 7 ТО Цифра_7	ЕСЛИ 7 ТО Цифра_7
ИНАЧЕ ЕСЛИ 8 ТО Цифра_8	ЕСЛИ 8 ТО Цифра_8
ИНАЧЕ ЕСЛИ 9 ТО Цифра_9	ЕСЛИ 9 ТО Цифра_9
КОНЕЦ	КОНЕЦ

Запись 1 — это переключатель, запись 2 — нет, хотя многие начинающие программисты записывают переключение по образцу записи 2. Ведь эта запись проще!

Дело в том, что вторая запись может привести к трудноуловимой ошибке.

Допустим, надо увеличить  $x$  на 1, если  $x$  равен 3, и на 2, если он равен 4. Напишем так (этот код не имеет отношения к Кукараче):

```
// запись 2
ЕСЛИ x = 3 ТО x := x + 1
ЕСЛИ x = 4 ТО x := x + 2
```

Предположим, что перед вычислениями  $x$  равнялся 3. Чему будет равен  $x$  после работы этих двух команд? Ведь не четырем, как хотелось, правда? Он станет равным 6.

А если записать так:

```
// запись 1
ЕСЛИ x = 3 ТО x := x + 1
ИНАЧЕ ЕСЛИ x = 4 ТО x := x + 2
```

то все будет работать правильно.

Итак, запись 2 на роль переключателя не годится: она может привести к ошибке. Кроме того, запись 2, несмотря на свою простоту, приводит к неэффективным программам. В таких записях все проверки работают всегда, независимо от значения переменной. В записи 1 количество выполненных проверок зависит от значения переменной и может равняться одному.

Запись 1 не эквивалентна записи 2 и для Кукарачи.

Рассмотрим два варианта кода:

Запись 1	Запись 2
ЭТО Код1	ЭТО Код2
ВНИЗ	ВНИЗ
ЕСЛИ 1 ТО ВПРАВО	ЕСЛИ 1 ТО ВПРАВО
ИНАЧЕ ЕСЛИ 2	ЕСЛИ 2
ТО ПОВТОРИ 2 ВПРАВО	ТО ПОВТОРИ 2 ВПРАВО
ВВЕРХ	ВВЕРХ
КОНЕЦ	КОНЕЦ

На рисунках показаны результаты выполнения этих кодов при одном и том же начальном состоянии среды. Результаты совершенно разные.

Результат выполнения кода 1:

**До выполнения**

	1	2	3	4	5	6	7	8	9	10
1	☀									
2	1	2								
3										
4										

**После выполнения**

	1	2	3	4	5	6	7	8	9	10
1										
2		☀	2							
3	1									
4										



Результат выполнения кода 2:

**До выполнения**

	1	2	3	4	5	6	7	8	9	10
1	☀									
2	1	2								
3										
4										

**После выполнения**

	1	2	3	4	5	6	7	8	9	10
1										
2				☀	2					
3	1									
4										

### Тестирование

Как-то Вася решил такую задачу.

**Задача.** *Расставь цифры*

Исполнитель стоит в левом верхнем углу поля. Во второй строке, начиная с первой клетки, находится плотный ряд кубиков с цифрами. Число кубиков не более 9. Цифры в ряду расположены в произвольном порядке, и значение каждой из них больше 2. Поставить кубики в строки с номерами, равными значениям цифр. На рисунке показан пример начального состояния среды и соответствующий ему результат.

**Дано**

	1	2	3	4	5	6	7	8	9	10
1	☀									
2	3	6	4	7	8	5	4	9		
3										
4										
5										
6										
7										
8										
9										
10										

**Надо**

	1	2	3	4	5	6	7	8	9	10
1									☀	
2										
3	3									
4			4				4			
5						5				
6		6								
7				7						
8					8					
9								9		
10										

Вася написал программу, запустил ее один раз (программа сработала верно) и хотел было перейти к решению другой задачи, но брат его остановил!

**Петя.** После того как программа написана, ее начинают *тестировать*, то есть проверять, правильно ли она работает. Чтобы убедиться в правильности работы программы на 100%, нужно запустить ее при *всех* возможных начальных состояниях среды. На рисунке показано несколько таких вариантов, допустимых по условию задачи:

	1	2	3	4	5	6	7	8	9	10
1	☀									
2	3									
3										
4										
5										
6										
7										
8										
9										
10										

	1	2	3	4	5	6	7	8	9	10
1	☀									
2	3	4	5	6	7	8	9			
3										
4										
5										
6										
7										
8										
9										
10										

	1	2	3	4	5	6	7	8	9	10
1	☀									
2	7	5	5	7	8	6	7	3	3	
3										
4										
5										
6										
7										
8										
9										
10										

Интересно, сколько всего начальных состояний среды у задачи *Расставь цифры?*

Вы не поверите, но количество вариантов здесь измеряется числом 47 079 207, то есть вариантов более 47 миллионов! Если выполнять проверку со скоростью один запуск варианта в минуту, потребуется примерно 90 лет непрерывной работы!



Когда Вася услышал это, он был шокирован, впал в отчаяние, сказал, что ему, возможно, не хватит и жизни, чтобы проверить правильность всего одной программы. Стоит ли тогда вообще заниматься программированием?

Петя привел брата в чувство следующими словами.

**Петя.** Не беспокойся, брат! Ни один программист не станет проверять свою программу 90 лет. На проверку берем лишь то, что **предположительно** заставит работать все участки нашего программного кода при типичных и особых начальных состояниях среды.

Процесс проверки программы называют *тестированием*, а наборы начальных состояний — *тестами*. Вот необходимый набор тестов для нашей задачи (которую мы еще даже не начинали решать):



Исходная запись	Комментарий
3	Особый случай: одна цифра
3456789	Промежуточный вариант между крайними. Присутствуют все возможные цифры (проверим правильность установки каждой цифры)
755786733	Особый случай: максимальное число цифр

А что делать, если обнаружен тест, на котором программа работает неверно? Нужно найти в тексте программного кода ошибку и исправить ее. Для поиска места ошибки можно запускать выполнение программы по шагам (среда разработки программ для Кукарачи это позволяет).

Процесс тестирования и исправления ошибок программисты называют *отладкой* программы. Тестирование устанавливает только факт наличия ошибки, а для исправления нужно найти место ошибки в программном коде, исправить и убедиться, что непрошедший тест теперь проходит правильно.

### Цикл ПОКА

**Петя.** Хочу предложить необычную задачу. Вот посмотри на рисунок:





	1	2	3	4	5	6	7	8	9	10
1										А
2										
3										
4										

Буква А стоит в углу, в клетке (1,10). Ее надо удалить с поля.

**Вася.** А где же Кукарача?

**Петя.** Где-то в первой строке.

**Вася.** Как же я напишу программу, не зная положения исполнителя?

**Петя.** В этом-то и состоит задача: программа должна работать для *любого* положения Кукарачи в первой строке.

**Вася.** Если напишу **ПОВТОРИ 9 ВПРАВО**, то где бы ни был Кукарача, он всегда выбросит букву с поля!

**Петя.** Да, так поступить можно. Но чаще всего выполнение этой команды будет заканчиваться аварийным сообщением Не могу.

**Вася.** Понятно. Но как же поступить? Ведь Кукарача должен двигаться вправо, пока не толкнет букву А. Число повторений заранее неизвестно, значит, команда **ПОВТОРИ** здесь не годится.

**Петя.** Верно подметил: надо двигаться вправо, пока не обнаружится буква А! Именно так, с ключевого слова **ПОКА** и начинается еще одна команда повторения. Замечу, что программисты обычно называют команды повторения **циклами**. Вот как записывается команда цикла **ПОКА** для нашей задачи:

**ПОКА НЕ А ВПРАВО**

**Вася.** Замечательно! Это то, что нужно.

**Петя.** В общем случае эта команда имеет вид:

**ПОКА условие команда**

Условие записывается точно так же, как и в команде ветвления.

Работает команда так:

1. Сначала проверяется условие.
2. Если условие ложно, выполнение цикла заканчивается.
3. Если условие истинно, выполняется команда, записанная после условия, и все начинается сначала: проверяется условие, выполняется команда и так далее.

Схема работы цикла **ПОКА**:



**Вася.** Значит, если условие ложно с самого начала, то команда, записанная после условия, не выполнится ни разу?

**Петя.** Правильно. В связи с этим хочу заметить, что код

**ПОКА НЕ А ВПРАВО**

для нахождения кубика А в некоторых случаях может оказаться неверным.

**Вася.** Почему?

**Петя.** Попробуй решить такую задачу.

**Задача. Две буквы**

Кукарача стоит в клетке (1,1). На поле расположены две буквы А. Одна — сразу под Кукарачей, другая — где-то во второй строке. Переместить первую букву в третью строку, а Кукарачу поставить на место второй. Пример начального состояния среды показан на рисунке.

	1	2	3	4	5	6	7	8	9	10
1	☀									
2	А						А			
3										
4										

**Вася.** Очень просто! Вот мое решение:

**ЭТО Вход**

**ВНИЗ**

**ПОКА НЕ А ВПРАВО**

**КОНЕЦ**

**Петя.** Теперь запусти эту программу.

**Вася.** Ничего не понимаю!

Кукарача толкает кубик вниз и останавливается. Почему он не идет вправо?



**Петя.** Причина в том, что условие цикла ложно с самого начала — ведь перед циклом Кукарача толкнул букву А. Команда **ВПРАВО**, записанная в цикле, не выполнится ни разу — условие цикла ложно, цикл работать не будет.

**Вася.** Как же быть?

**Петя.** Надо, чтобы Кукарача “забыл”, что толкнул первую букву А. Переместим для этого Кукарачу в направлении второй буквы А:

**ЭТО Вход**

**Первый\_кубик\_вниз**

**На\_место\_второго\_кубика**

**КОНЕЦ**

**ЭТО Первый\_кубик\_вниз**

**ВНИЗ**

**КОНЕЦ**

**ЭТО На\_место\_второго\_кубика**

**ВПРАВО // Шаг в направлении кубика**

**ПОКА НЕ А ВПРАВО**

**КОНЕЦ**

## Как проверяются условия

Петя уточняет механизм проверки условий.

**Петя.** В среде есть специальная *ячейка памяти* (ее не видно на экране). В начальный момент (перед выполнением программы) в эту ячейку помещается значение **ПУСТО**.

**ПУСТО**

Затем значение ячейки меняется каждый раз, когда Кукарача передвигается в соседнюю клетку. Если в клетке был кубик А (исполнитель толкнул его), в ячейку записывается символ с кубика, в противном случае в ячейку записывается **ПУСТО**.

**А**

Роботландцы называют эту ячейку индикатором состояния клетки, или просто *индикатором*.

Проверка условия состоит в сравнении индикатора с тем, что записано после ключевого слова **ЕСЛИ** или **ПОКА**. Результат проверки: **истина** или **ложь**.

## Сквозь стену

**Задача.** *Сквозь стену*

Найти проход в стене и дойти до края поля (в десятом столбце). На рисунке показан пример начального состояния среды.

	1	2	3	4	5	6	7	8	9	10
1		Ж								
2		Ж								
3		Ж								
4		Ж								
5		Ж								
6		Ж								
7										
8		Ж								
9		Ж								
10		Ж								

**Петя.** Здесь нельзя использовать команду **ПОВТОРИ**, потому что расположение прохода неизвестно (на рисунке он изображен приблизительно). Мы не знаем, сколько надо сделать шагов до прохода, значит, неизвестно, какое число повторений надо указать в этой команде. Что же делать? А вот что. Надо идти вдоль стены (кубики с буквой Ж), **пока** стена есть. Потом пройти в обнаруженный проход.

**Вася.** Попробую записать это так:

```
ЭТО Сквозь_стену
  Поиск_прохода
  На_край_поля
КОНЕЦ
```

Первая процедура написана, а как искать проход?

**Петя.** Толкать кубики, спускаясь вниз, пока не обнаружится пустое место.

**Вася.** Попробую так:

```
ЭТО Поиск_прохода
  ПОКА НЕ ПУСТО Шаг
КОНЕЦ
```

Поиск\_прохода

	1	2	3	4	5	6	7	8	9	10
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										

На\_край\_поля

**Петя.** Команда **Шаг** не выполнится ни разу. Ведь в начальный момент Кукарача не толкал кубики, поэтому условие в команде **ПОКА** — ложно.

**Вася.** Хорошо, тогда напишу так:

```
ЭТО Поиск_прохода
  ВПРАВО
  ПОКА НЕ ПУСТО Шаг
КОНЕЦ
```

**Петя.** Это другое дело. Продолжай дальше.

	1	2	3	4	5	6	7	8	9	10
1			Ж							
2			Ж							
3			Ж							
4			Ж							
5			Ж							
6			Ж							
7										
8			Ж							
9			Ж							
10			Ж							

**Вася.** Теперь напишу процедуру **Шаг**:

```
ЭТО Шаг
  ВЛЕВО // В первый столбец
  ВНИЗ // На следующую строку
  ВПРАВО // Толкнуть кубик
КОНЕЦ
```

**Петя.** Очень хорошо.

	1	2	3	4	5	6	7	8	9	10
1			Ж							
2			Ж							
3			Ж							
4			Ж							
5			Ж							
6			Ж							
7		☀								
8		Ж								
9		Ж								
10		Ж								

Вася. Теперь осталось написать совсем простую процедуру **На\_край\_поля**:

**ЭТО** **На\_край\_поля**  
**ПОВТОРИ 8 ВПРАВО**  
**КОНЕЦ**

**Никто не нужен**

**Задача.** *Никто не нужен*

Кукарача расположен где-то в первой строке своего поля. Справа от него плотно стоят несколько кубиков (не занимая последний столбец). Удалить кубики с поля. Примеры возможных начальных состояний среды показаны на рисунке.



	1	2	3	4	5	6	7	8	9	10
1		☀	Д	Р	А	К	О	Н		
2										
3										
4										
5										

	1	2	3	4	5	6	7	8	9	10
1					☀	Т	И	Г	Р	
2										
3										
4										
5										

**Решение**

**Обсуждение решения**

Ранее были решены похожие задачи *Уж не нужен* и *Бегемот не нужен*. В решениях этих задач была задействована процедура **Шаг**:

**ЭТО** **Шаг**  
**ВНИЗ**  
**ВПРАВО**  
**ВВЕРХ**  
**КОНЕЦ**

	1	2	3	4	5	6	7	8	9	10
1				☀	У	Ж				
2										
3										
4										
5										

	1	2	3	4	5	6	7	8	9	10
1		☀	Б	Е	Г	Е	М	О	Т	
2										
3										
4										
5										

Процедура **Шаг** удаляет одну букву и устанавливает исполнителя на ее место. **УЖ** был удален при помощи конструкции **ПОВТОРИ 3 Шаг**, а **БЕГЕМОТ** — при помощи конструкции **ПОВТОРИ 7 Шаг**.

В задаче *Никто не нужен* тоже будем опираться на процедуру **Шаг**, но число повторений заранее неизвестно, значит, нужен цикл **ПОКА**. Будем выполнять **ПОКА НЕ ПУСТО Шаг**. Чтобы удалить из индикатора начальное состояние **ПУСТО**, один **Шаг** выполним перед циклом (иначе тело цикла не выполнится ни разу).

Окончание цикла наступает, когда исполнитель “толкает” пустую клетку. Вот почему в условии введено требование пустого 10-го столбца.

Отметим еще одну деталь. В условиях задач *Уж не нужен* и *Бегемот не нужен* требовалось вернуть исполнителя в исходное положение после удаления всех кубиков. Здесь мы не можем этого потребовать. С циклом **ПОКА** такую задачу не решить. Цикл **ПОКА** не умеет считать число совершенных оборотов цикла, такое под силу только рекурсии, изучение которой предстоит в следующем году, в “Азбуке 4”.

**Программа**

**ЭТО** **Вход**  
**Шаг**  
**ПОКА НЕ ПУСТО Шаг**  
**КОНЕЦ**

**ЭТО** **Шаг**

**ВНИЗ ВПРАВО ВВЕРХ**

**КОНЕЦ**

**Рекурсивная программа**

Раз уж речь зашла о рекурсии, приведем рекурсивный вариант программы (пока без объяснений, рекурсивный Кукарача — одна из тем следующей части курса “Азбука Роботландии”), который не только удаляет кубики, но и возвращает Кукарачу в исходную позицию (ту, с которой он начинает работу):

ЭТО Вход

Шаг

ЕСЛИ НЕ ПУСТО ТО Вход

ВЛЕВО // Обеспечивает возврат  
// в исходную точку

КОНЕЦ

ЭТО Шаг

ВНИЗ ВПРАВО ВВЕРХ

КОНЕЦ

Тесты

Проверки должны охватить как слова разной длины (от 0 до 8 кубиков), так и разные начальные положения исполнителя. Можно предложить такой набор тестов (символом “#” обозначен исполнитель, а точками — пустые клетки):

```
#КРОКОДИЛ.  
#ТИГР.....  
...#ТИГР.  
..#ТИГР...  
....#.....
```

## Азбука Роботландии. Часть III. Алгоритмы + Графика

Эта заметка познакомила читателя с Кукарачей, который поселился в 3-й части “Азбуки Роботландии”. Но в этой же части курса ученики осваивают теорию и практику конструирования рисунка в растровом графическом редакторе (на базе бесплатного Paint.NET). Этой теме мы посвятим отдельную заметку, а здесь приведем несколько ссылок.

1. ДемOVERсия “Азбуки Роботландии”:

<http://robotlandia.ru/abc.htm>.

2. 15-дневная ознакомительная версия среды разработчика Кукарача Windows:

<ftp://anonymous@robotlandia.ru@robotlandia.ru/cockroach/cockroach-1.0.8.38.exe> (если спросят пароль, просто нажмите Enter).

3. Задачи турниров Роботландского университета: <ftp://anonymous%40robotlandia.ru@robotlandia.ru/cockroach/azhtml.zip>.

Запуск гипертекстового приложения (после распаковки архива) — файл [index.htm](#).

4. Заказ продуктов Роботландии:

<http://www.botik.ru/~robot/sale>.

Для тех, кто первый раз слышит об “Азбуке Роботландии”, сообщаем, что это курс информатики для начальной и средней школы, который пози-

ционируется авторами как курс, закладывающий основы компьютерной, информационной, алгоритмической и коммуникационной грамотности школьников.

В “Азбуке” запланировано пять ключевых разделов:

- Азбука 1: Компьютер
- Азбука 2: Информация + Текст
- Азбука 3: Алгоритмы + Растровая графика
- Азбука 4: Алгоритмы 2 + Векторная графика
- Азбука 5: Интернет + Медиа

Мы хотим научить детей конкретным приемам работы с информацией, но самое главное — мы хотим научить детей мыслить алгоритмически, ведь основа информатики — это теория и практика *составления алгоритмов* для обработки информации. Развитие алгоритмического мышления — наша основная цель.

## “Азбука Роботландии” в журнале “Информатика”

№	Часть	Название статьи	Выходные данные
9	Азбука 3	Азбука 3: роботландский зоопарк исполнителей	№ 10, октябрь, 2014 год
8	Азбука 2	“Азбука Роботландии”: работаем с буфером обмена, строим и читаем деревья	№ 12, декабрь, 2013 год
7	Азбука 2	“Азбука Роботландии”: информатика для началки должна быть фундаментальной	№ 4, апрель, 2013 год
6	Азбука 2	Структуры данных	№ 9, октябрь, 2012 год
5	Азбука 2	Алгоритмы редактирования текста	№ 8, сентябрь, 2012 год
4	Азбука 2	Азбука Роботландии. Информация	№ 7, август, 2012 год
3	Азбука 1	Устройства ввода и вывода. Теория относительности	№ 14, сентябрь, 2011 год
2	Азбука 1	Алгоритмы и программы “Азбуки Роботландии”	№ 14, сентябрь, 2011 год
1	Азбука 1	Азбука Роботландии	№ 14, сентябрь, 2011 год



**Роботландский сетевой университет** — работает с октября по май каждого учебного года.

В Университет принимаются коллективные, а на курс “42. Web-конструирование” — коллективные и индивидуальные ученики.

*Коллективный студент* — это группа детей, работающая под руководством одного или нескольких наставников (наставник, как правило — школьный учитель).

*Индивидуальный студент* — это учитель, желающий пройти обучение индивидуально, без группы детей.

В конце года обучаемые получают удостоверения от негосударственного образовательного учреждения “Роботландия+” и грамоты за победы в курсовых конкурсах.

Занятия в Университете платные (цены — на странице [www.botik.ru/~robot/ru/price.htm](http://www.botik.ru/~robot/ru/price.htm)). Они начинаются 10 октября текущего года и продолжаются в течение двух семестров до мая следующего года. **Для подписчиков “Информатики” предусмотрена 10%-ная скидка за обучение.**

Характерные черты Роботландской школы:

1. Совместное обучение учителя и школьников в рамках одной команды.
2. Турнирный цикл обучения.
3. Моделирование коллективной деятельности.
4. Реальная практическая польза детских проектов.
5. Перекрестные проверки работ.
6. Развитые горизонтальные связи.

**Заявки принимаются по адресу:** [kurs@robotland.pereslavl.ru](mailto:kurs@robotland.pereslavl.ru) в сентябре.

#### Краткое описание курсов

Номер и название курса	Возраст детей	Куратор курса	Описание курса
10. Азбука Роботландии. Компьютер	1–3-е классы	Первин Юрий Абрамович	Введение в информатику на базе курса “Азбука Роботландии”, часть I “Компьютер”
11. Азбука Роботландии. Информация + Текст	2–4-е классы	Первин Юрий Абрамович	Продолжение курса “Азбука Роботландии”
12. Азбука Роботландии. Алгоритмы. Графика (растр)	3–5-е классы	Первин Юрий Абрамович	Продолжение курса “Азбука Роботландии”
13. Азбука Роботландии. Алгоритмы 2. Графика (вектор)	4–6-е классы	Первин Юрий Абрамович	Продолжение курса “Азбука Роботландии”
31. Азы программирования-I. Плюстик и Кукарача	6–8-е классы	Садовая Ирина Владимировна	Введение в программирование
32. Азы программирования-II. Корректор	7–9-е классы	Садовая Ирина Владимировна	Продолжение курса 31 на базе исполнителя Корректор
42. Web-конструирование	старшие классы	Дуванов Александр Александрович	Основы создания сайтов и гипертекстовых приложений на базе HTML+CSS+JavaScript (последнее факультативно). Основы проектирования, веб-дизайна и юзабилити



Педагогический университет  
**«Первое сентября»**

## ДИСТАНЦИОННЫЕ КУРСЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ

(с учетом требований ФГОС)

До 30 сентября ведется прием заявок на первый поток 2015/16 учебного года

### образовательные программы:

- НОРМАТИВНЫЙ СРОК ОСВОЕНИЯ – 108 УЧЕБНЫХ ЧАСОВ  
Стоимость – 4990 руб.

- НОРМАТИВНЫЙ СРОК ОСВОЕНИЯ – 72 УЧЕБНЫХ ЧАСА  
Стоимость – от 3990 руб.

По окончании выдается удостоверение о повышении квалификации  
установленного образца

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

Пожалуйста, обратите внимание:

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



## Применение карт взаимосвязи понятий учебного материала

### Аннотация

► В статье предлагается обратить внимание на составление карт взаимосвязи между базовыми понятиями изучаемого материала. Обсуждаются достоинства таких карт для изучения, систематизации и анализа изучаемого материала. Рассмотрены конкретные примеры различных типов карт, продемонстрированы их основные возможности.

Эта статья не претендует на роль полного обзора в обсуждаемой области и не содержит каких-то бесспорных методических рекомендаций. Напротив, ее цель состоит в том, чтобы, используя конкретные примеры карт понятий, вызвать коллег на обсуждение материала.

Различного рода схемы, изображающие взаимосвязи между терминами изучаемого материала, появились гораздо раньше, чем компьютеризированное обучение (цветных карандашей и бумаги для этой цели вполне достаточно). Тем не менее внедрение компьютеров дает педагогам новые возможности в этой области: переход к компьютерной графике делает процесс создания и редактирования схем значительно удобнее, а его результаты — качественнее.

Способы рисования схем необычайно разнообразны. Чтобы убедиться в этом, достаточно посмотреть обзоры [1, 2]. В последнее время стало заметно

**Е.А. Еремин,**  
к. ф.-м. н., доцент кафедры  
мультимедийной дидактики  
и ИТО ПГГПУ, г. Пермь,  
[eremin@pspu.ru](mailto:eremin@pspu.ru),

**К.Ю. Поляков,**  
д. т. н., учитель информатики  
ГБОУ СОШ № 163,  
г. Санкт-Петербург,  
[kpolyakov@mail.ru](mailto:kpolyakov@mail.ru)

стремление представить в виде структур из базовых понятий содержание не только отдельных тем, но даже целых учебников, причем учебники информатики стали использовать эти технологии одними из первых. В частности, как только появился учебник пермских авторов под редакцией И.Г. Семакина (первое издание [3] вышло в 1998 году), в его методическое сопровождение был включен сборник структурированных конспектов [4], изображавших базовые понятия и их взаимосвязь для каждой темы курса. Другим учебником, в котором существенное внимание уделено картам взаимосвязей, является учебник С.А. Бешенкова и Е.А. Ракиной [5]. Немного позднее система структурированных схем, “раскрывающих различные аспекты изучаемых тем предмета «Информатика и ИКТ» и показывающих основные понятия и их взаимосвязи”, была опубликована в виде небольшого альбома [6].

Имеются успешные попытки построения схем взаимосвязи понятий и по другим предметам. Так, в статье [7] подробно описывается структурно-логическая модель, включающая в себя термины из всего школьного курса физики.

### Карты понятий как основа для глубоких знаний

Чем привлекают учителей карты взаимосвязи понятий? Прежде всего наглядностью и систематичностью представления материала в сочетании с компактностью изображения. Подобно тому, как географическая карта позволяет увидеть место, где мы находимся, и что располагается неподалеку, карта понятий дает возможность оценить место изучаемого понятия среди других связанных с ним понятий.

Согласно педагогической теории, представление знаний в виде определенной взаимосвязанной системы имеет большое значение: “сформировавшаяся система знаний — важнейшее средство предотвращения их забывания. Забытые знания быстро восстанавливаются в системе, без нее — с большим трудом” [8]. Известный психолог Дж. Брунер даже утверждал, что в обучении “изложение структуры знаний, овладение этой структурой, а не просто усвоение фактов и технических приемов является центральным моментом” [9].

Имеются прямые экспериментальные доказательства важности составления карт понятий. Так испанская группа “CIBERDIDACT” выдвинула и развила “теорию ядерных понятий” [10], которая базируется на предположении о существовании у студентов некоторых структур знаний, составленных из взаимосвязанных понятий. Новые знания при обучении должны как-то присоединяться к этой структуре из уже

усвоенных концепций. Проведя эксперименты с усвоением знаний 440 студентами, исследователи пришли к довольно неожиданному выводу. Оказалось, что новые понятия стихийно присоединяются у студентов не к важным обобщенным понятиям высокого уровня, а, как кажется со стороны, к случайным — некоторые из которых вообще относятся к иллюстративным примерам. Эти-то концепции и получили название *ядерных*, так как они служат ядрами для концентрации вокруг себя новых понятий.

Авторы утверждают, что студенты в основном строят свои системы знаний, руководствуясь не столько оценкой важности или общности новых терминов, сколько простотой их присоединения к существующему кругу знаний. В итоге в головах студентов часто создаются весьма причудливые и запутанные сети из терминов. Авторы назвали такой способ формирования структур знаний термином “path of low-cost”, смысл которого великолепно передается русской фразой “идти по линии наименьшего сопротивления”.

Таким образом, для улучшения качества знаний надо не усложнять, а упрощать систему понятий, стихийно формирующуюся в головах учащихся, и рассматриваемые карты могут существенно помочь в этом.

Дополнительное объяснение роли карт во взаимосвязи понятий дается в публикации [11]. Она напрямую связывает присоединение новых понятий к существующей внутренней структуре знаний с *пониманием* учебного материала: “ученик понимает новый материал, если в его сознании известные и новые предметы мышления могут быть соединены с помощью известных ученику видов связей”. В статье также обосновывается важность умения выделять разнообразные связи между понятиями при усвоении знаний.

### Виды карт

Объем учебного материала, для которого составляется карта понятий, может быть различен. В соответствии с ним карты получаются разные по сложности и объему, причем большие карты, как правило, требуют при построении значительной работы, зато приобретают в результате более мощные дидактические возможности. Стоит отметить, что применение сложной карты потребует от педагога более тщательной методической подготовки: недостаточно сразу предложить ученику готовое “масштабное полотно”.

Наиболее естественно выделить следующие разновидности карт понятий:

- конспект конкретного параграфа;
- карта темы (раздела или главы);
- карты, связывающие разные темы;
- карта курса в целом.

Далее будут показаны примеры для этих случаев и рассмотрены их основные возможности.

### Карты понятий как конспект конкретного параграфа

Простейшая карта взаимосвязи понятий получается, если графически изобразить основные обсуждающиеся в данном параграфе термины и их взаимосвязи. Пример подобной схемы для параграфа “Редактирование текста” приведен на рис. 1.

Вследствие простоты и небольшого объема составление такой карты можно смело поручить ученикам. Тем не менее желательно, чтобы учитель тоже заранее глубоко проанализировал материал, составил свою карту и помог ученикам добавить пропущенные ими существенные связи.

Подобная карта позволяет повторить пройденный на уроке материал и лучше запомнить его. В педагогике подобного рода схемы часто называют *опорным конспектом*.

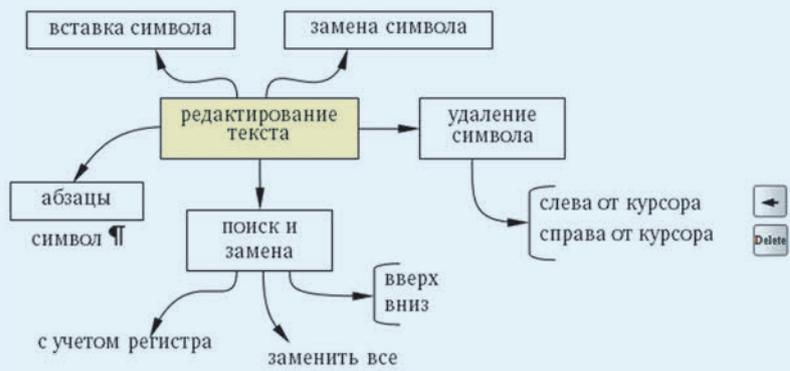


Рис. 1. Карта понятий для параграфа

### Карты понятий для целой темы

Карта для конкретного параграфа получается простой и наглядной, но она, как правило, не отражает связей материала данного параграфа с остальными. Особенно это касается ссылок на материал, который еще только предстоит изучить, ибо невозможно как-то отразить на карте то понятие, которое пока не изучалось.

Без специальных указаний учителя есть риск, что учащийся усвоит отраженный на схеме пара-

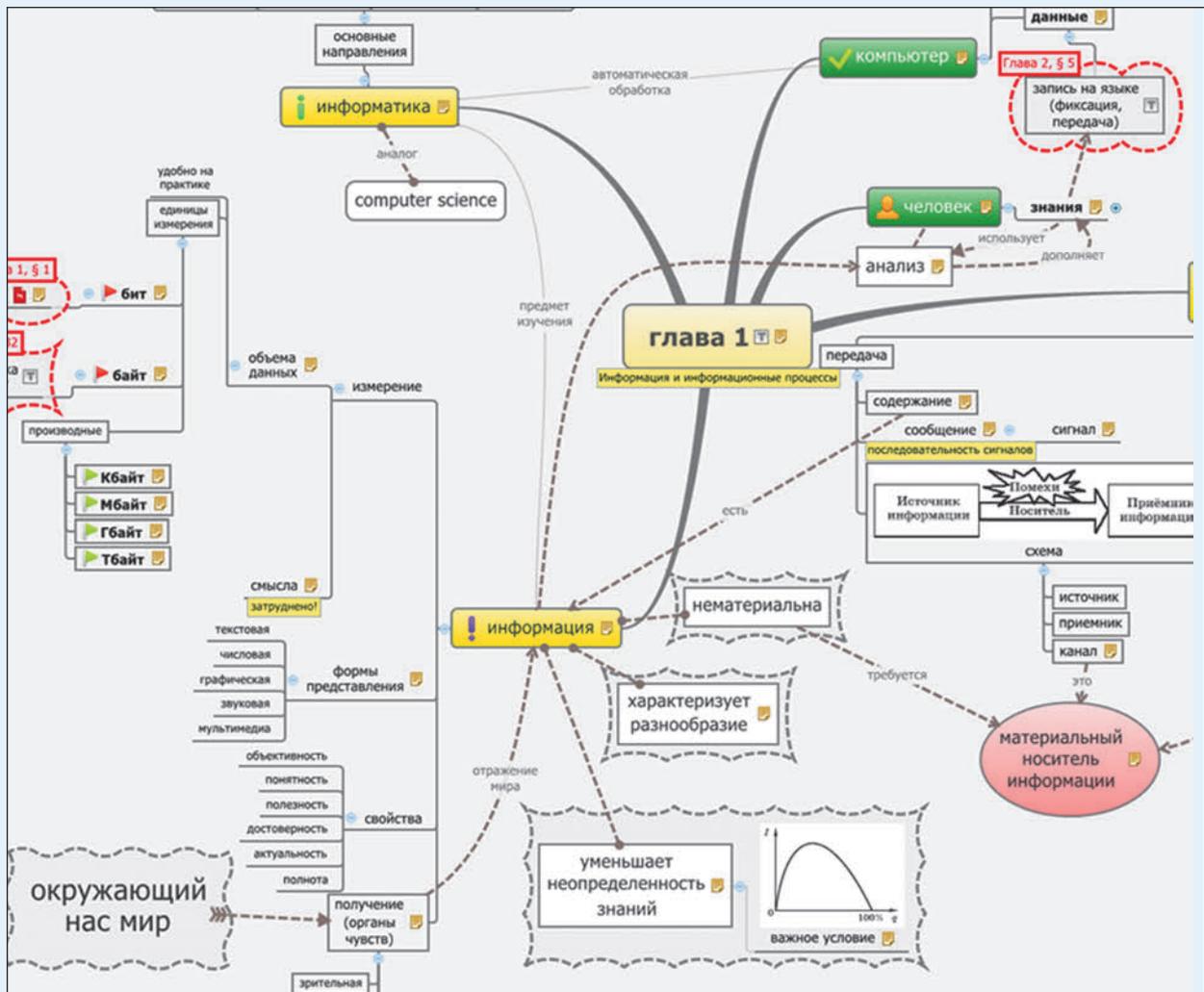


Рис. 2. Фрагмент карты главы

графа материал изолированно от остальных тем. С этих позиций можно усложнить схему, объединив на ней термины более крупного фрагмента учебника, например, главы. Такие карты для каждой главы нашего учебника [12, 13] выложены на сайте поддержки [14]. Карты имеют достаточно большой размер, поэтому по полиграфическим соображениям здесь не будут приведены.

Достоинством такого рода карт является систематизация полученных знаний. Попытка оглянуться назад и, повторяя пройденное, объединить полученные на отдельных уроках знания в некоторую общую картину имеет самостоятельную педагогическую ценность.

У обсуждаемых карт есть и еще одна интересная особенность. При тщательном построении карта “сама” начинает подсказывать смысловые связи. Так, на карте самой первой главы учебника “Информация и информационные процессы” сформировалась непредусмотренная заранее цепочка: *окружающий мир — получение информации через органы чувств — собственно информация — ее анализ, который проводится человеком с использованием знаний*. А далее видно, что знания человека для обработки на компьютере превращаются в данные путем записи на некотором языке (центральная часть карты вынесена на рис. 2). Дополнительно поясним, что о получении информации говорится в одном разделе параграфа, а о превращении информации в знание и фиксации знаний — в другом, причем между ними находится еще один небольшой раздел о формах представления информации. Следовательно, отследить целиком указанную выше цепочку без карты удастся не каждому.

Построить карту главы самостоятельно ученику трудно. По-видимому, здесь не удастся обойтись без предварительной подготовительной работы учителя. Дополнительную сложность в процесс составления таких крупных карт вносит субъективность наших знаний и представлений. В связи с этим хочется порекомендовать учителям не воспринимать полученную схему догматически и по

возможности дорабатывать ее. Стоит внимательно проверять карты, составленные учениками, глядя на них под углом поиска новых существенных связей для своей схемы.

По-видимому, можно построить работу с картой главы так: разбить ее на части, выделяя отдельные параграфы; при этом постоянно держать в голове, что в конце изучения темы отдельные карты (или их фрагменты) планируется собирать вместе.

## Карты, связывающие разные темы

Увидеть связи между материалом, изученным в разное время, ученикам без помощи учителя еще труднее. Поэтому для отдельных особо важных понятий и терминов можно рекомендовать еще один класс карт — карты связи данного понятия с остальными темами курса.

В качестве примера на рис. 3 приведена схема связей для понятий “бит” и “байт”. Из нее видно, что понятия эти имеют отношение к самым разным частям курса информатики. Во-первых, бит и байт — это *единицы измерения информации*, которые формально вводятся в теоретической информатике, причем бит — это *часть байта*. Во-вторых, бит и байт имеют самое непосредственное отношение к вычислительной технике: бит — это *минимальный элемент хранения информации* (электронная схема под названием *триггер* хранит 1 бит), а байт — это *ячейка памяти* (минимальная адресуемая часть информации; т.е. байт имеет *адрес*, а бит — нет). В-третьих, бит — это *двоичная цифра*, т.е. существует связь этого термина с двоичной системой счисления. Наконец, в-четвертых, наличие битовой организации данных существенным образом связано с их *дискретностью*, так как нет никаких других значений, кроме 0 или 1.

Таким образом мы видим, что фундаментальные понятия “бит” и “байт” используются в самых разных разделах информатики. Подобные связи “цементируют” учебный курс, делая его целостным.

Заметим, что на рис. 3 дополнительно показано, что если “вовремя не остановиться”, то схема начинает обрастать разными деталями, которые постепенно будут “размывать” суть карты. Поэтому можно посоветовать выносить на рисунок только те понятия, которые действительно требуются для иллюстрации изучаемого материала.

Приведенную на рисунке схему в литературе по представлению знаний принято называть *семантической сетью*. Для математического описания таких структур знаний используется *теория графов*.

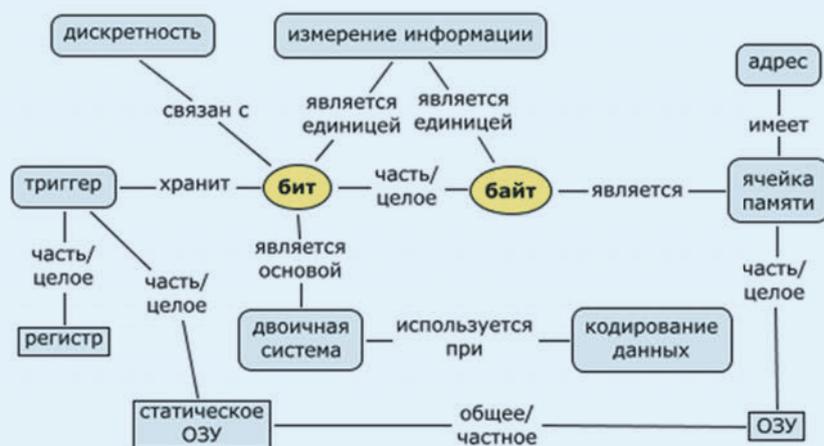


Рис. 3. Схема связей понятий “бит” и “байт”

## Карта всего курса

По-видимому, данный вид карт предназначен не столько для работы по ним с учениками, сколько для детального методического анализа содержания учебного курса и при составлении рабочих программ. Не случайно реальные карты курсов в книгах [4, 6] разбиты на отдельные темы. Если захочется увидеть пример единой схемы, на которой объединены сразу все базовые понятия школьного курса информатики, можно обратиться к публикации [15].

Заметим, что предложенная в [7] структурная схема школьного курса физики придумана главным образом ради удобства навигации по электронным учебным материалам. Можно предположить, что с увеличением количества электронных учебников роль подобных карт будет возрастать.

## Роль карт в освоении важнейших типов связей

Одной из важных задач курса информатики является развитие у учеников способностей к систематизации знаний. В связи с этим обратимся к двум наиболее часто используемым типам связей (см., например, учебник [16]): *часть/целое* и *общее/частное* (часто связь этого типа называют *класс/подкласс*). Первая всегда возникает, когда вы рассказываете ученикам об устройстве того или иного сложного составного объекта; она наиболее удачно характеризуется фразой “*X входит в состав Y*”. Вторая связь служит основой любой классификации и описывается словами “*X является разновидностью Y*”.

О важности этих связей говорит тот факт, что они подробно изучаются в курсе информатики начальной школы.

На рис. 4 показана карта терминов, использующихся при описании графического интерфейса. Связи, соответствующие отношению *часть/целое*, помечены небольшим разделенным на части кружком ⊗, а *общее/частное* — развилкой <. Интересной особенностью данной схемы является тот факт, что на ней нет никаких иных связей, кроме двух обсуждаемых.

Как свидетельствует личный опыт одного из авторов, проверившего умение своих студентов

различать эти два типа связей [17], результаты неутешительны: большая часть студентов их путает. Это обстоятельство только подчеркивает полезность упражнений с указанным типом карт.

Подводя итоги, можно сказать, что применение карт взаимосвязи учебных понятий принесет пользу как учителю, так и ученикам. Наиболее отчетливо эта польза заметна при планировании учебного материала учителем и при повторении пройденного учениками; систематизация материала одинаково полезна обеим сторонам учебного процесса.

## Литература

1. Босова Л.Л. Об использовании графических схем в курсе информатики и ИКТ. Информатика и образование, 2008, № 5, с. 16–25.
2. Fisher K.M. Overview of Knowledge Mapping. In: Mapping Biology Knowledge. Science and Technology Education Library, 2002, vol. 11, pp. 5–23.
3. Семакин И.Г., Залогова Л.А., Русаков С.В., Шестакова Л.В. Информатика. Базовый курс. Учебник для 7–9-х классов. М.: Лаборатория Базовых Знаний, 1998.
4. Семакин И.Г., Вараксин Г.С. Информатика. Структурированный конспект базового курса. М.: Лаборатория Базовых Знаний, 2001.
5. Бешенков С.А., Ракитина Е.А. Информатика. Систематический курс: Учебник для 10-го класса. М.: БИНОМ. Лаборатория знаний, 2004.
6. Афанасьева Н.Е., Гаврилова С.А., Ракитина Е.А., Вязовова О.В. Информатика в схемах. М.: БИНОМ. Лаборатория знаний, 2006.
7. Баяндин Д.В., Мухин О.И. Структурно-логическая модель школьного курса физики в электронных средствах образовательного назначения. Вестник ПГГПУ, серия “Информационные компьютерные технологии в образовании”, выпуск 9. Пермь: ПГГПУ, 2013, с. 28–45.
8. Подласый И.П. Педагогика: Новый курс. Кн. 1: Общие основы. Процесс обучения. М.: Гуманит. изд. центр ВЛАДОС, 2002.
9. Брунер Дж. Процесс обучения. М.: АПН РСФСР, 1962.
10. Luengo R., Casas L.M., Mendoza M., Arias J. Possibilities of “Nuclear Concepts Theory” on Educational Research, a Review. Proceedings of International Conference “The Future of Education”, edition 1. [http://conference.pixel-online.net/edu\\_future/common/download/Paper\\_pdf/SOE33-Gonzalez,Garcia,Garcia,Masa.pdf](http://conference.pixel-online.net/edu_future/common/download/Paper_pdf/SOE33-Gonzalez,Garcia,Garcia,Masa.pdf).
11. Бершадский М. Педагогическая диагностика уровня понимания. Педагогические измерения, 2012, № 3, с. 60–88.
12. Поляков К.Ю., Еремин Е.А. Информатика. Углубленный уровень: учебник для 10-го класса: в 2 ч. М.: БИНОМ. Лаборатория знаний, 2013.
13. Поляков К.Ю., Еремин Е.А. Информатика. Углубленный уровень: учебник для 11-го класса: в 2 ч. М.: БИНОМ. Лаборатория знаний, 2013.
14. <http://kpolyakov.spb.ru/school/probook/mindmaps.htm>.
15. Семакин И.Г. Пермской версии школьной информатики 20 лет: с чего начинали и до чего дошли. “Информатика”, 2013, № 10, с. 28–42.
16. Суворова Н.И. Информационное моделирование. Величины, объекты, алгоритмы. М.: Лаборатория Базовых Знаний, 2002.
17. Еремин Е.А. Разрозненные факты или единое целое: экспериментальная оценка концептуальных знаний студентов. “Информатика и образование”, 2012, № 10, с. 90–96.

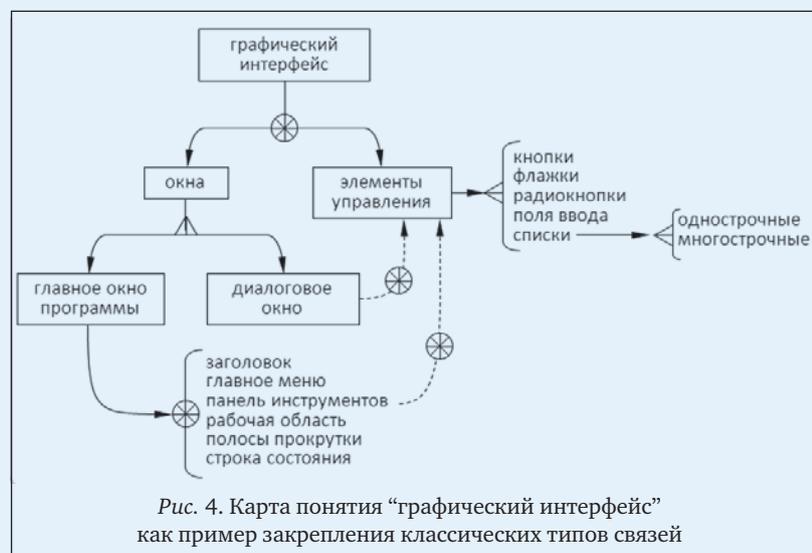


Рис. 4. Карта понятия “графический интерфейс” как пример закрепления классических типов связей



Общероссийский проект  
**Школа цифрового века**

---

**6 тысяч рублей от школы  
за весь 2015/16 учебный год  
независимо от количества учителей  
в образовательной организации**

Каждому учителю:

- 24 предметных ежемесячных журнала
- десятки курсов повышения квалификации

**Не забудьте принять  
или продлить участие!**

Подробности и форма заявки на сайте:

**[digital.1september.ru](http://digital.1september.ru)**



## Табличные формулы, или Действия над диапазонами

Д.М. Златопольский,  
Москва

► В электронной таблице Microsoft Excel и других аналогичных программах имеется замечательная<sup>1</sup> возможность проводить расчеты не только над данными, записанными в отдельных ячейках, но и в диапазоне ячеек. “Но это не новость!” — скажете вы, и будете правы. Конечно, именно это делают многие функции, имеющиеся в программе: СУММ (определяет количество числовых значений в заданном диапазоне), СРЗНАЧ (возвращает среднее арифметическое чисел в диапазоне), СЧЕТЕСЛИ (подсчитывает количество значений, удовлетворяющих некоторому условию) и другие, у которых в качестве аргументов указываются диапазоны ячеек. А если функции использовать нельзя?

<sup>1</sup> Автор надеется, что после знакомства с данной статьей читатель согласится с такой оценкой.

Рассмотрим пример.

*Пример 1.* На листе (см. рис. 1) представлена информация о цене товаров в условных единицах. Подготовить лист для получения в столбце С цены каждого товара в рублях в соответствии с курсом валюты, указываемым в ячейке В102.

	А	В	С
1	Товар	Цена, у.е.	Цена, руб.
2	Тележка	23	764,29
3	Люк	3	99,69
4	Двигатель	124	4120,52
...			
100	Трансформатор	76	2525,48
101			
102	Курс	33,23	

Рис. 1

### Решение

Можно в ячейку C2 ввести одну формулу:

$$=B2*\$B\$102 \quad (1)$$

и затем распространить (скопировать) ее на остальные ячейки столбца C.

Но можно поступить и по-другому — получить во всех ячейках столбца C одну и ту же формулу:

$$\{=B2:B100*B102\} \quad (2)$$

Смысл формулы понятен — надо умножить содержимое каждой ячейки диапазона B2:B100 на значение в ячейке B102. Такие формулы, в которых, так сказать, “участвует” в расчетах не отдельная ячейка, а диапазон ячеек, называют “табличными формулами”, или “формулами массива”.

Обратим внимание на то, что в последней формуле использована “обычная” (относительная) ссылка на ячейку B102, в то время как в формуле (1) должна быть применена так называемая “абсолютная” ссылка (можно было записать и смешанную ссылку в виде B\$102).

Чтобы получить в некотором диапазоне табличную формулу, необходимо выполнить следующие действия.

1. Выделить диапазон ячеек, в который необходимо ввести формулы (в нашем случае — C2:C100).

2. Щелкнуть мышью в строке формул и ввести в нее знак равенства — признак формулы.

3. Выделить диапазон B2:B100 — он отразится в строке формул:

$$=B2:B100$$

4. Ввести в строку формул знак “\*”.

5. Ввести в строку формул адрес B102.

6. Закончить ввод формулы особым способом — нажать одновременно клавиши **Shift**, **Ctrl** и **Enter**<sup>2</sup>.

В результате, если все сделано правильно, во все ячейки диапазона будет введена одна и та же формула (2), которая в строке формул будет взята в фигурные скобки. Они говорят о том, что это табличная формула. Эти скобки нельзя набирать вручную (формула будет воспринята как текст). Заметим, что если бы мы нажали **Enter**, то формула была бы введена только в активную ячейку диапазона — C2.

Заметим, что на этапе 3 диапазон B2:B100 можно в формулу ввести и “вручную”, что особенно удобно при больших диапазонах.

Обладает ли новый способ какими-либо другими преимуществами перед “традиционным” в данном случае — решать вам, уважаемый читатель (к преимуществам применения табличных формул мы еще неоднократно вернемся ниже).

Можно в формулах использовать два и более диапазонов.

**Пример 2.** Фирма имеет 10 магазинов. Подготовить лист для определения прибыли каждого магазина по известной информации о доходах и расходах (см. рис. 2).

	A	B	C	D
1	Магазин №	Доходы	Расходы	Прибыль
2	1			
3	2			
...				
11	10			

Рис. 2

### Решение

1. Выделить диапазон ячеек D2:D11.

2. Щелкнуть мышью в строке формул и ввести в нее знак равенства.

3. Выделить или указать диапазон B2:B11.

4. Ввести в строку формул знак “=”.

5. Выделить или указать диапазон C2:C11 — он также отразится в строке формул.

6. Нажать аккорд **Shift** + **Ctrl** + **Enter**.

В результате во всех ячейках диапазона будет использована табличная формула  $\{=B2:B11-C2:C11\}$ .

Рассмотренные задачи относятся к одному типу задач, который можно описать так: “Проведение расчетов по одной и той же формуле для диапазона ячеек”.

Прежде чем предлагать задания для самостоятельной работы учащихся, а потом идти дальше, расскажем еще кое-что.

Диапазоны в табличных формулах можно “умножать” и т.п. При этом скобки использовать не обязательно:  $\{=2*N14:N162-O14:O162/2\}$ .

Если планируется копирование табличной формулы, то при необходимости следует применить абсолютные или смешанные ссылки.

Для изменения табличной формулы сразу во всех ячейках с ней:

1) выделите ячейку (ячейки), содержащую (содержащие) формулу; это удобно сделать, нажав аккорд **Ctrl** + <“/”>;

2) щелкните в строке формул — фигурные скобки в табличной формуле исчезнут (можно также перейти в режим редактирования формул, нажав функциональную клавишу **F2**);

3) измените формулу;

4) нажмите одновременно клавиши **Shift**, **Ctrl** и **Enter**.

Удаляются табличные формулы как “обычные” — с помощью клавиши **Delete** ☹. Но одну формулу (или несколько) из диапазона удалить нельзя — это можно сделать только для всех ячеек с данной табличной формулой. Если это все же сделать необходимо — например, на листе, показанном на рис. 2, нужно удалить ячейку D11, — то следует поступить так:

1) выделить диапазон D2:D11 (нажав аккорд **Ctrl** + <“/”>);

2) войти в режим редактирования (**F2**) или строка формул);

3) добавить в начале формулы апостроф (') — формула превратится в текст;

<sup>2</sup> Комбинацию из нескольких клавиш, нажимаемых одновременно, называют “аккорд”.

- 4) нажать аккорд **Ctrl** + **Enter** — табличная формула прекратит существование;
- 5) очистить последнюю строку таблицы;
- 6) выделить диапазон D2:D10;
- 7) ввести в него новую табличную формулу.

Если понадобится *добавить* табличную формулу еще в одну или несколько ячеек в дополнение к имеющимся, то путем копирования имеющейся формулы это сделать нельзя. Например, если на листе, показанном на *рис. 2*, необходимо добавить еще один магазин (см. *рис. 2а*), то после копирования в ячейку D12 имеющейся табличной формулы в ней будет получена формула с другим диапазоном:

$$\{=B12:B21-C12:C21\}$$

— что вовсе не входит в наши планы.

	A	B	C	D
1	Магазин №	Доход	Расход	Прибыль
2	1			
3	2			
...				
11	10			
12	11			

Рис. 2а

Нужный результат можно получить так:

- 1) выделить диапазон D2:D12 (сразу весь или сначала D2:D11, нажав аккорд **Ctrl** + <“/”> и увеличив его на одну ячейку);
- 2) изменить в формуле используемый диапазон на D2:D12;
- 3) нажать аккорд **Shift** + **Ctrl** + **Enter**.

Конечно, процедуры уменьшения и увеличения размеров диапазона с данной табличной формулой непростые и неприятные<sup>3</sup>. Но применение табличных формул для решения ряда задач, как мы еще убедимся, дает такие дополнительные возможности, что с неудобствами, связанными с изменениями этих формул, придется смириться.

### Задания для самостоятельной работы учащихся<sup>4</sup>

1. Известны радиусы 20 окружностей. Подготовить лист для определения:
  - а) длины каждой из них;
  - б) площади соответствующих кругов.
2. На листе (см. *рис. 3*) записана информация о площади территории и численности населения 12 государств Европы.

Подготовить лист для определения плотности населения, выраженной в количестве человек на 1 км<sup>2</sup>.

**Примечание.** Данные по населению стран к моменту публикации статьи могли измениться.

	A	B	C
1	Название	Площадь территории, тыс. кв. км	Численность населения, тыс. чел.
2	Австрия	83,9	8015
3	Албания	28,7	3411
4	Андорра	0,5	62
5	Белоруссия	207,6	10 367
6	Бельгия	30,5	10 072
7	Болгария	111,0	8469
8	Босния и Герцеговина	51,1	4422
9	Ватикан	0,0004	0,8
10	Великобритания	244,1	58 041
11	Венгрия	93,0	10 262
12	Германия	357,0	81 338
13	Греция	132,0	10 350

Рис. 3

3. Известны масса и объем восьми предметов (*рис. 4*):

Номер предмета	Объем, см <sup>3</sup>	Масса, кг
1	10130	23,3
2	1067	1,6
3	1207	3,5
4	3111	2,8
5	12222	8,8
6	1583	1,9
7	1537	10,3
8	2333	5,6

Рис. 4

Подготовить лист для определения плотности материала каждого из них, выраженной в г/мм<sup>3</sup>.

4. В диапазоне A1:A20 записаны числа. Определить в ячейках B2:B20 сумму соседних чисел.
5. Установите, что произойдет, если размер диапазона, в котором записываются табличные формулы, будет отличаться от размера диапазона в формуле. Рассмотрите два случая:
  - а) первый упомянутый диапазон меньше второго;
  - б) первый диапазон больше второго.

Рассмотрим еще один пример.

**Пример 3.** В диапазоне A1:M2 записаны числа (см. *рис. 5*).

	A	B	C	D	...	M
1	4	23	4	13		3
2	5	6	10	13		5
3						
4						

Рис. 5

Необходимо в ячейке A4 получить среднее значение частных от деления значений в первой строке на значения в соответствующем столбце второй строки.

<sup>3</sup> Можно, конечно, очищать имеющиеся диапазоны с табличной формулой и вводить новую формулу в новый диапазон. Но и это задачу не упрощает.

<sup>4</sup> Все задачи для самостоятельной работы должны быть решены с использованием табличных формул.

Традиционный способ решения задачи такой:

- 1) в ячейку A3 вводится формула =A1/A2;
- 2) она распространяется (копируется) на остальные ячейки диапазона A3:M3; в результате в них будут получены соответствующие частные:

	A	B	C	D	...	M
1	4	23	4	13		3
2	5	6	10	13		5
3	0,8	3,833333	0,4	1		0,6
4						

Рис. 6

3) в ячейке A4 находится искомое среднее значение по формуле =СРЗНАЧ(A3:M3).

При использовании табличных формул задача решается так. В ячейку A4 вводится формула, начинающаяся функцией СРЗНАЧ, а в качестве ее аргументов указывается (“вручную” или путем выделения соответствующих диапазонов): A3:M3/A4:M4. После нажатия аккорда [Shift] + [Ctrl] + [Enter] будет получена табличная формула {=СРЗНАЧ(A3:M3/A4:M4)}, естественно, с требуемым результатом. Обратите внимание — хотя формула возвращает значение только в одной ячейке, поскольку в ней проводятся операции над диапазонами, она должна вводиться как табличная (если ввести эту формулу как обычную, то она вернет сообщение об ошибке #ЗНАЧ!).

Легко убедиться, что в данном случае использование табличной формулы имеет явное преимущество по сравнению с традиционным способом решения задачи с точки зрения количества используемых ячеек. Кроме того, при этом не требуется копирование формулы, что особенно важно при больших диапазонах для копирования.

Важно понять, как работает приведенная табличная формула. В результате выполнения операции деления одного диапазона на другой (можно сказать — одного массива чисел на другой) A3:M3/A4:M4 будет сформирован (в памяти компьютера) массив частных того же размера, что и массивы-операнды. После этого с помощью функции СРЗНАЧ будет рассчитано среднее значение элементов массива частных.

Общая схема алгоритма решения рассмотренной и подобных задач такая:

Этап 1. Расчеты по одной и той же формуле для диапазона (диапазонов) ячеек.

Этап 2. Расчеты по результатам расчетов (извините за “масляное масло” ☺) на этапе 1.

Схематически такие задачи можно будет обозначить в виде:

Расчеты2 (Расчеты1 (Диапазоны) )

При “внешних” (на этапе 2) расчетах, кроме функции СРЗНАЧ, можно использовать также функции СУММ, СЧЕТ, МАКС, МИН, КОРЕНЬ, СТЕПЕНЬ и т.п.

Сразу же сделаем важное замечание. Функции СЧЕТЕСЛИ и СУММЕСЛИ (подсчитывает сумму значений в диапазоне, удовлетворяющих некоторому

условию) на этапе 2 применять нельзя. Поэтому для данных, показанных на рис. 5, нельзя определить количество частных от деления, равных 1, по формуле: {=СЧЕТЕСЛИ(A1:M1/A2:M2;1)}.

### Задания для самостоятельной работы учащихся

6. Известны радиусы 20 окружностей. Подготовить лист для определения:

а) среднего арифметического площадей соответствующих кругов;

б) максимальной из площадей соответствующих кругов.

7. Известны размеры 15 прямоугольников. Подготовить лист для определения:

а) минимальной из их площадей;

б) максимальной из диагоналей заданных прямоугольников.

8. Известны размеры 18 параллелепипедов. Подготовить лист для определения:

а) среднего арифметического их объемов;

б) суммарного объема всех параллелепипедов.

9. На листе (см. рис. 7) записаны координаты  $(x_i, y_i)$  на плоскости 10 пар точек.

	A	B	C	D	E	F
1	№	x1	y1	x2	y2	R
2	1	2,3	4,8	4,5	3,5	
3	2	-12,7	-5,2	0	2,8	
...						
11	10	-5,0	3,1	4,6	-3,7	

Рис. 7

Определить расстояние R между точками каждой пары.

10. На листе (см. рис. 8) записаны координаты  $(x_i, y_i)$  на плоскости вершин треугольника ABC.

	A	B	C	D	E
1	Сторона	x1	y1	x2	y2
2	AB	2,3	4,8	4,5	3,5
3	BC	4,5	3,5	0	2,8
4	AC	2,3	4,8	0	2,8

Рис. 8

Определить периметр указанного треугольника. Кроме арифметических операций, над диапазонами можно выполнять и операции отношения.

**Пример 4.** В столбцах A и B записаны числа a и b (см. рис. 9). Получить в столбце C ответ на вопрос, соблюдается ли неравенство  $a > b$ .

	A	B	C
1	a	b	a > b?
2	14	3	ИСТИНА
3	5	26	ЛОЖЬ
4			
...			
21	7	7	ЛОЖЬ

Рис. 9

Соответствующая табличная формула:  
 {=A2:A21>B2:B21}.

Это означает, что диапазоны можно использовать в качестве аргумента логической функции электронных таблиц ЕСЛИ. В качестве примера решим задачу, упоминавшуюся чуть выше, — для данных, показанных на рис. 5, определим количество частных от деления, равных 1. Но сначала приведем решение без применения табличных формул. В этом случае используется функция СЧЕТЕСЛИ<sup>5</sup>. Она подсчитывает количество ячеек внутри диапазона, удовлетворяющих заданному критерию. Ее формат:

СЧЕТЕСЛИ (диапазон;критерий)

Аргумент критерий в форме числа, текста или адреса ячейки определяет, какие ячейки надо подсчитывать. Например, критерий может быть выражен следующим образом: 10, “10”, “яблоки”, D6. Можно также использовать критерий со знаком сравнения и числом: “=10”; “>10” и т.п.

Учитывая это, для решения задачи следует после расчета частных от деления значений в первой строке на значения в соответствующем столбце второй строки (см. рис. 6) определить искомую величину по формуле =СЧЕТЕСЛИ(A3:M3;1). Общее число используемых ячеек (кроме ячеек с исходными данными) — 14. (В дальнейшем это число будем называть “число ячеек для решения задачи”.)

А как же решить задачу с применением табличной формулы? О том, что функцию СЧЕТЕСЛИ использовать нельзя, мы уже говорили (проверьте!). Выход можно найти при следующих рассуждениях.

Можно сравнить каждое частное с 1, используя функцию ЕСЛИ, и в зависимости от результата проверки получить 1 (при истинности условия) и 0 (при ложности). После этого искомое значение может быть найдено с помощью функции СУММ. Итак, задача решается путем ввода формулы только в одну (!) ячейку:  
 {=СУММ(ЕСЛИ(A1:F1/A2:F2=1;1;0))}

Описанный подход может быть использован для решения широкого круга задач. При рассмотрении соответствующих задач значения, возвращаемые функцией ЕСЛИ (в рассмотренном примере — 1 или 0), в целях краткости будем называть “индикаторами” выполнения того или иного условия.

**Пример 5.** В диапазоне ячеек A2:A25 записаны целые числа. Определить, сколько из них четных.

*Решение без использования табличных формул (Функцию СЧЕТЕСЛИ применить нельзя.)*  
 1. Во “вспомогательном” диапазоне M2:M25 получить индикаторы в зависимости от того, является ли соответствующее число в столбце A четным. Для проверки четности числа удобно использовать функцию ОСТАТ. Формула в ячейке M2:  
 =ЕСЛИ(ОСТАТ(A2;2)=0;1;0)  
 — распространяется (копируется) на остальные ячейки столбца M.  
 2. Искомое значение определяется по формуле:  
 =СУММ(M2:M25)  
 Число ячеек для решения задачи — 25.

**Задания для самостоятельной работы учащихся**

11. Решите задачу из примера 5 без использования функции ЕСЛИ.

12. На листе (см. рис. 10) записана информация о результатах десяти игр баскетбольной команды (число очков, набранных ею и командой-соперником).

	A	B	C
1	<b>Номер игры</b>	<b>Очки</b>	
2		<b>Набраны ею</b>	<b>Набраны соперником</b>
3	1	86	68
4	2	56	60
...			
12	10	91	79
13	<b>Всего выигрышей</b>		

Рис. 10

Определить количество выигрышей данной команды.

13. На листе (см. рис. 11) представлена информация о результатах десяти игр футбольной команды (число забитых и пропущенных мячей).

	A	B	C
1	<b>Номер игры</b>	<b>Забито</b>	<b>Пропущено</b>
2	1	2	1
3	2	3	0
...			
11	10	1	1
12	<b>Всего выигрышей</b>		
13	<b>Всего ничьих</b>		
14	<b>Всего проигрышей</b>		

Рис. 11

Определить:

- а) общее количество выигрышей данной команды;
- б) общее количество ничьих;
- в) общее количество проигрышей.

14. В диапазоне ячеек A2:N2 записаны целые числа. Определить, сколько из них оканчиваются нулем. Функцию, с помощью которой можно определить последнюю цифру любого целого числа, установите самостоятельно.

*Решение с использованием табличных формул {=СУММ(ЕСЛИ(ОСТАТ(A2:A25;2)=0;1;0))}*  
 Число ячеек для решения задачи — 1.

<sup>5</sup> Можно именно так, а не СЧЕТЕСЛИ.

15. На листе (см. рис. 12) представлены итоговые отметки учеников по алгебре и геометрии:

	A	B	C	D
1	№	Фамилия, имя	Алгебра	Геометрия
2	1	Абызов Павел	4	4
3	2	Бойко Сергей	5	5
...				
28	27	Яновская Мария	5	4

Рис. 12

Определить количество учеников, имеющих отметку “5” по обоим предметам.

16. В диапазоне A1:A20 записаны числа. Определить, сколько из них принимают:

- наибольшее значение;
- наименьшее значение.

17. В диапазоне A1:N1 записаны числа. Определить количество чисел, являющихся квадратами целых чисел. Условие, по которому можно проверить указанную особенность числа, определите самостоятельно.

18. В диапазоне A1:A20 записаны числа. Определить количество пар соседних чисел, сумма которых равна некоторому числу, задаваемому в ячейке C1.

19. В диапазоне A1:N1 записаны числа. Определить количество чисел, меньших, чем среднее арифметическое их “соседей”.

20. В диапазоне A1:A10 записана числовая последовательность. Определить, является ли она возрастающей (при просмотре сверху вниз).

21. Решите предыдущую задачу для случая, когда в диапазоне A1:A10 может быть записана последовательность из менее чем десяти чисел.

На некоторое время “забудем” о табличных формулах и вернемся к функции СЧЕТЕСЛИ. Несмотря на кажущуюся простоту используемого в ней критерия для подсчета количества значений, здесь имеется ряд “подводных камней”.

Обсудим ряд примеров.

**Пример 6.** На листе (см. рис. 13) записаны отметки учеников по информатике.

	A	B
1	Успеваемость по информатике	
2	Фамилия, имя	Отметка
3	Антонов Егор	4
...		
27	Щукина Алла	5
28		
29	Количество отметок “5”	

Рис. 13

Определить количество пятятерок.

**Решение**

Искомое значение в ячейке B29 определяется по любой из формул:

$$=СЧЕТЕСЛИ(В3:В27;"=5") \quad (3)$$

$$=СЧЕТЕСЛИ(В3:В27;5) \quad (4)$$

$$=СЧЕТЕСЛИ(В3:В27;"5") \quad (5)$$

Как уже отмечалось, в качестве критерия в функции СЧЕТЕСЛИ может быть использован так-

же адрес некоторой ячейки. Для иллюстрации несколько изменим условие задачи: “Известны отметки по информатике каждого из 25 учеников класса. Оформить лист для определения количества одной из отметок 5, 4, 3 или 2. Значение отметки, количество которой должно быть определено, указывается в отдельной ячейке (см. рис. 14)”.

**Решение**

	A	B
1	Успеваемость по информатике	
2	Фамилия, имя	Отметка
3	Антонов Егор	4
...		
27	Щукина Анна	5
28		
29	Введите отметку	
30	Количество отметок	

Рис. 14

Искомое значение находится по формуле:  $=СЧЕТЕСЛИ(В3:В27;В29)$ .

Если же по аналогии с критерием, являющимся числом (см. формулы (3) и (5)), записать последнюю формулу в виде  $=СЧЕТЕСЛИ(В3:В27;"=В29")$  или  $=СЧЕТЕСЛИ(В3:В27; "В29")$ , то в обоих случаях она вернет значение 0 при любой оценке в ячейке B29. Попытка же убрать в первой из только что приведенных формул кавычки в записи критерия приведет к появлению сообщения об ошибке.

**Пример 7.** Известен рост каждого из 26 учеников класса. Определить, сколько учеников имеют рост больше 165 см.

**Решение**

Здесь возможен единственный вариант записи критерия: “>165”. Все другие способы записи (>165, >B29<sup>6</sup>, “>B29”) приведут к неправильным результатам или к появлению сообщения об ошибке. Между тем имеется обширный класс задач, в которых значение критерия для подсчета не известно заранее, как в примере 7, а указывается или рассчитывается по формуле в отдельной ячейке. Приведем несколько примеров.

**Пример 8.** Известен рост каждого из 24 учеников класса. Определить, сколько учеников имеют рост больше среднего по классу.

**Пример 9.** В диапазоне ячеек C2:C20 записаны числа. Определить, сколько из них меньше среднего арифметического максимального и минимального из чисел.

Оформим лист для решения задачи из примера 8 (см. рис. 15):

	A	B
1	Сведения о росте учащихся	
2	Фамилия, имя	Рост, см
3	Антонов Егор	167
...		
27	Щукина Анна	162
28		
29	Средний рост, см	164,7
30	Количество учеников, имеющих рост больше среднего	

Рис. 15

<sup>6</sup> Имеется в виду, что в ячейке B29 указано значение 165.

Как отмечалось чуть выше, формула в ячейке В30 не может быть оформлена в виде =СЧЕТЕСЛИ(В3:В27;>В29) или =СЧЕТЕСЛИ(В3:В27;>В29). Как и при решении задачи о количестве частных от деления, равных 1 (см. выше), приходится “хитрить”. Для каждого ученика определим, больше ли его рост среднего значения или нет, и ответ запишем в одном из столбцов (лучше — вне зоны видимости листа, например, в столбце М) в виде чисел 1 и 0<sup>7</sup>:

	А	В	...	М
1	<b>Сведения о росте учащихся</b>			
2	<b>Фамилия, имя</b>	<b>Рост, см</b>		
3	Антонов Е.	167		1
...				
27	Щукина А.	162		0
28				
29	Средний рост, см	164,7		
30	Количество учеников, имеющих рост больше среднего			

Рис. 16

В этом случае значение в ячейке В30 находится суммированием значений в диапазоне М3:М27.

А теперь покажем, как задача решается с использованием табличной формулы — в этом случае потребуется ввести формулу только в одну (!) ячейку В30:

{=СУММ(ЕСЛИ(В3:В27>СРЗНАЧ(В3:В27);1;0))}

Заметим, что до приобретения достаточного опыта применения табличных формул можно рекомендовать некоторые используемые в ней значения рассчитывать отдельно в каких-то ячейках (например, в рассмотренном примере — значение СРЗНАЧ(В3:В27)).

### Задания для самостоятельной работы учащихся

**22.** Задачу из примера 8 можно решить и по-другому. Можно получить в одном из столбцов, например в том же столбце М, только те значения роста учеников, которые больше среднего роста (использовав для этого функцию ЕСЛИ). В случае ложности условия в ячейке записывается “пустое” значение (“”). Тогда искомая величина определяется по формуле: =СЧЕТ(М3:М27). Используйте эти рассуждения в табличной функции.

**23.** Решите задачу из примера 9, приведенного выше, с использованием табличной функции (для нахождения наибольшего и наименьшего значений в диапазоне ячеек применяются, соответственно, функции МАКС и МИН).

Преимущества табличных формул ярко проявляются также в задачах, в которых функция СЧЕТЕСЛИ не может быть применена из-за того, что в ней в качестве критерия не могут быть использованы так называемые “сложные условия”, записываемые с

помощью логических функций И, ИЛИ, НЕ. Рассмотрим следующий пример.

**Пример 10.** Известно количество осадков (в мм), выпавших за каждый месяц года:

Месяц	Кол-во осадков, мм
Январь	37,2
Февраль	11,4
Март	16,5
Апрель	19,5
Май	11,7
Июнь	129,1
Июль	57,1
Август	43,8
Сентябрь	8,7
Октябрь	86,0
Ноябрь	12,5
Декабрь	21,2

Подготовить лист для определения числа месяцев, в которых выпадало от 20 до 100 мм осадков.

#### Комментарии к решению

Если переписать данные этой таблицы на лист в ячейки А1:В13 (см. рис. 17), а затем в ячейке с результатом записать формулу =СЧЕТЕСЛИ(В2:В13; И(">=20"; "<=100")), то она вернет результат, равный нулю, хотя месяцы, в которые выпало соответствующее количество осадков, имеются. Правильный результат здесь можно получить так, как при решении задачи из примера 8, использовав в функции ЕСЛИ сложное условие (логическую функцию И), а затем — функцию СУММ для определения искомого значения:

	А	В	...	М
1	<b>Месяц</b>	<b>Кол-во осадков</b>		
2	Январь	37,2		1
3	Февраль	11,4		0
4	Март	16,5		0
5	Апрель	19,5		0
6	Май	11,7		0
7	Июнь	129,1		0
8	Июль	57,1		1
9	Август	43,8		1
10	Сентябрь	8,7		0
11	Октябрь	86		1
12	Ноябрь	12,5		0
13	Декабрь	21,2		1
14				
15	Ответ:	5		

Рис. 17

С использованием табличных формул результат может быть получен при записи формулы только в одну ячейку. Вот эта формула:

{=СУММ(ЕСЛИ(ЕСЛИ(В2:В13>20;1;0)+ЕСЛИ(В2:В13<100;1;0)=2;1;0))}

Она работает так. Для каждого значения в диапазоне В2:В13 отдельно проверяются два “простых” условия и в зависимости от результата проверки

<sup>7</sup> Напомним, что мы условились называть такие значения “индикаторами”.

определяются индикаторы 1 или 0. Эти индикаторы суммируются. Затем каждая из 12 сумм проверяется на равенство числу 2, и также в каждом случае запоминается результат, равный 1 или 0. Последние 12 индикаторов суммируются.

Для решения аналогичных задач приведем общий вид формулы:

$$\{=\text{СУММ}(\text{ЕСЛИ}(\text{ЕСЛИ}(\dots)+\text{ЕСЛИ}(\dots)=2;1;0))\} \quad (6)$$

Заметим, что попытка использовать вместо двух “внутренних” (с индикаторами) функций ЕСЛИ одну с логической функцией И:

$$\{=\text{СУММ}(\text{ЕСЛИ}(\text{И}(\text{В2:В13}>20;\text{В2:В13}<100);1;0)=1;1;0))\} \quad (7)$$

— не приведет к требуемому результату.

Если в условии задачи подразумевается не функция И (как в рассмотренной задаче), а ИЛИ, то в последней формуле при определении индикаторов для суммирования вместо условия “=2” должно использоваться следующее: “>0”. Общий вид табличной формулы в этом случае такой:

$$\{=\text{СУММ}(\text{ЕСЛИ}(\text{ЕСЛИ}(\dots)+\text{ЕСЛИ}(\dots)+\dots>0;1;0))\} \quad (7)$$

### Задания для самостоятельной работы учащихся

**24.** В диапазоне A1:L1 записаны целые числа, среди которых есть и отрицательные. Определить количество четных положительных чисел.

**25.** В диапазоне A1:N1 записаны целые числа. Определить количество чисел, кратных трем и заканчивающихся нулем.

**26.** Для условия задачи примера 10 подготовить лист для определения числа месяцев, в которых выпадало менее 10 или более 90 мм осадков.

Аналогично решаются задачи на определение количества значений, удовлетворяющих некоторому сложному условию, проверяемому в двух и более *разных* диапазонах. Приведем пример задачи.

**Пример 11.** На листе записаны отметки 26 учеников класса по алгебре и геометрии (см. рис. 18):

	A	B	C	D	E
1		<b>Фамилия, имя</b>	<b>Алгебра</b>	<b>Геометрия</b>	
2	1	Аксенова Анна	4	3	
3	2	Бойко Петр	5	5	
...					
27	26	Яновский Игорь	4	5	
28					
29		Ответ:			

Рис. 18

Определить число учеников, имеющих отметку “4” по обоим предметам.

*Решение*

Здесь получить искомое значение по сумме отметок нельзя (убедитесь в этом!). Следует использовать формулу (6):

$$\{=\text{СУММ}(\text{ЕСЛИ}(\text{ЕСЛИ}(\text{В2:В26}=4;1;0)+\text{ЕСЛИ}(\text{С2:С26}=4;1;0)=2;1;0))\}$$

### Задание для самостоятельной работы учащихся

**27.** Для условий задачи примера 11 определить:

- число учеников, имеющих хотя бы одну отметку “5”;
- число учеников, не имеющих ни одной отметки “3” (принять, что двоек в таблице нет);
- число учеников, имеющих только одну пятёрку.

Во всех последних задачах в качестве индикаторов мы использовали значения 1 и 0. Можно также после проверки условия (условий) в функции ЕСЛИ возвращать значения из ячеек. Это позволяет определять не только количество, но и сумму значений, удовлетворяющих некоторому условию.

**Пример 12.** В диапазоне A2:A31 записаны целые числа. Определить сумму нечетных чисел.

<p><i>Решение без использования табличных формул</i> (Функцию СУММЕСЛИ применить нельзя.)</p> <p>1. Во “вспомогательном” диапазоне M2:M31 получить значения нечетных чисел из столбца A. Формула в ячейке M2:</p> $=\text{ЕСЛИ}(\text{ОСТАТ}(\text{A2};2)=1;\text{A2};0)$ <p>— распространяется (копируется) на остальные ячейки столбца M.</p> <p>2. Искомое значение определяется по формуле:</p> $=\text{СУММ}(\text{M2:M31})$ <p>Число ячеек для решения задачи — 31.</p>	<p><i>Решение с использованием табличных формул</i> {=СУММ(ЕСЛИ(ОСТАТ(A2:A31;2)=1; A2:A31;0))}</p> <p>По результатам проверки чисел из диапазона A2:A31 формируется массив, в котором элементы равны нечетным числам из указанного диапазона либо нулю.</p> <p>Число ячеек для решения задачи — 1.</p>
---	--

**Задание для самостоятельной работы учащихся**

28. В диапазоне A1:L1 записаны числа, среди которых есть и отрицательные. Определить сумму таких чисел.

Если для суммирования должно использоваться сложное условие, то применяется формула, аналогичная формулам (6) или (7) (см. выше).

**Задание для самостоятельной работы учащихся**

29. В диапазоне A1:L1 записаны целые числа. Определить сумму чисел, кратных семи и одновременно заканчивающихся единицей.

Можно также условие проверять в одном диапазоне, а значения использовать из другого.

**Пример 13.** На листе записаны годовые отметки по информатике 26 учеников класса (см. рис. 19):

	A	B	C
1		<b>Фамилия, имя</b>	<b>Отметка</b>
2	1	Аксенова Анна	4
3	2	Бойко Петр	5
...			
27	26	Яновский Игорь	4
28			
29		Введите фамилию и имя	
30		Его отметка	

Рис. 19

Подготовить лист для определения отметки ученика, фамилия и имя которого вводятся в ячейку C29.

*Решение*

Без применения табличных формул задача может быть решена аналогично задаче примера 12 (см. рис. 20).

	A	B	C	...	M
1		<b>Фамилия, имя</b>	<b>Отметка</b>		
2	1	Аксенова Анна	4		0
3	2	Бойко Петр	5		5
...					
27	26	Яновский Игорь	4		0
28					
29		Введите фамилию и имя	Бойко Петр		
30		Его отметка			

Рис. 20

После этого искомое значение определяется суммой значений в столбце M.

Табличная формула для решения задачи:  
 {=СУММ(ЕСЛИ(B2:B27=C29;C2:C27;0))}

**Задания для самостоятельной работы учащихся**

30. На листе записано расстояние от районного центра до 15 населенных пунктов (см. рис. 21).

	A	B
1	<b>Населенный пункт</b>	<b>Расстояние, км</b>
2	Абрамцево	14,5
3	Бекасово	22,8
...		
16	Ясиноватая	7,4
17		
18	Введите населенный пункт	
19	Расстояние от него до районного центра, км	

Рис. 21

Подготовить лист для определения расстояния от районного центра до населенного пункта, название которого вводится в ячейку B18.

31. Известен рост каждого из 20 учеников класса. Подготовить лист для определения роста ученика, фамилия которого вводится в одну из ячеек.

32. На листе (см. рис. 22) записана численность каждого из 25 классов школы.

	A	B	C	D
1		<b>Численность классов</b>		
2		Класс	Номер параллели	Численность класса
3	1	1-й "А"	1	29
4	2	1-й "Б"	1	27
5		2-й "А"		25
...				
27	25	11-й "В"	11	20
28				

Рис. 22

Подготовить лист для определения общего числа учащихся в задаваемой параллели. Количество классов в каждой параллели заранее не известно.

Аналогично можно получить сумму, произведение и другие величины для значений из двух и более столбцов (или строк).

**Пример 14.** Когда-то при поступлении в высшее учебное заведение подсчитывалась сумма баллов (отметок), полученных абитуриентом на каждом вступительном экзамене. На листе будут записаны отметки на трех экзаменах 86 абитуриентов (см. рис. 23):

	A	B	C	D	E
1		<b>Отметки на экзаменах</b>			
2		Фамилия, имя	Экзамен 1	Экзамен 2	Экзамен 3
3	1				
4	2				
...					
88	86				
89					
90		Введите фамилию и имя абитуриента			
91		Его сумма баллов			

Рис. 23

Подготовить лист для определения суммы баллов абитуриента, фамилия и имя которого вводятся в ячейку C29.

*Решение*

Следует найти сумму значений в трех столбцах C, D и E, соответствующих условию  $V3:B88=C90$ :

$$\{=СУММ(ЕСЛИ(V3:B88=C90;C3:C88;0)+ЕСЛИ(V3:B88=C90;D3:D88;0)+ЕСЛИ(V3:B88=C90;E3:E88;0))\}$$

Можно существенно сократить длину и “читаемость” формулы, если “складывать” диапазоны:

$$\{=СУММ(ЕСЛИ(V3:B88=C90;C3:C88+D3:D88+E3:E88;0))\}$$

### Задания для самостоятельной работы учащихся

33. На листе (рис. 24) представлена информация о размерах 12 параллелепипедов.

	A	B	C	D
1	№	Длина, см	Ширина, см	Высота, см
2	1	12,4	10,3	4,9
3	2	45,5	22,1	12,1
...				
13	12	33,7	25,4	15,8
14				
15	Введите номер параллелепипеда			
16	Его объем			

Рис. 24

Подготовить лист на нахождение в ячейке C16 объема параллелепипеда, номер которого указывается в ячейке D15.

34. На листе (рис. 25) представлена информация о самых больших странах Европы.

	A	B	C	D	E		
1	Наибольшая площадь					Наибольшее население	
2	№	Страна	Площадь, кв. км			№	Страна
3	1	Россия	17 102 345			1	Россия
4	2	Украина	576 604			2	Германия
5	3	Франция	547 030			3	Франция
6	4	Испания	497 304			4	Великобритания
7	5	Швеция	449 964			5	Италия
8	6	Германия	357 021			6	Испания
9	7	Финляндия	337 030			7	Украина
10	8	Норвегия	324 220			8	Польша
11	9	Польша	312 685			9	Румыния
12	10	Италия	301 340			10	Нидерланды
13	11	Великобритания	244 820			11	Греция
14	12	Румыния	237 500			12	Бельгия
15	13	Белоруссия	207 595			13	Португалия
16	14	Греция	131 940			14	Чехия
17	15	Болгария	110 910			15	Венгрия
18	16	Исландия	103 000			16	Сербия
19	17	Венгрия	93 030			17	Швеция
20	18	Португалия	92 082			18	Белоруссия
21							
22	Укажите страну						
23	Плотность населения в этой стране						

Рис. 25

В ячейке C23 определить плотность населения (чел./кв. км) в стране, название которой указывается в ячейке C22.

**Примечания**

1. Естественно, что указываемая страна должна присутствовать в обеих таблицах.
2. Данные по населению стран к моменту публикации статьи могли измениться.

Задачу решите тремя способами:

- 1) без использования табличных формул;

2) используя две табличные формулы;

3) используя одну табличную формулу.

35. На листе (рис. 26) представлена информация о размерах 22 треугольников.

	A	B	C	D
1	№	Основание, см	Высота, см	
2	1	12,4	10,3	
3	2	45,5	22,1	
...				
23	12	33,7	25,4	
24				
25	Введите номер треугольника			
26	Его площадь			

Рис. 26

Подготовить лист для нахождения в ячейке C26 площади треугольника, номер которого указывается в ячейке D25.

36. Выскажите свое мнение о том, когда целесообразно использовать табличные формулы, и о их преимуществах и недостатках.

37. Сравните размеры файлов для двух способов решения каких-нибудь рассмотренных задач (без использования табличных формул и с ними). Рассмотрите большие массивы данных.

И в заключение приведем ряд задач, которые вообще не могут быть решены (!) без использования табличных формул.

1. На листе (см. рис. 27) приведен список финансовых документов — счетов-фактур с указанием суммы счета и фамилии и имени продавца, его выписавшего ([1]).

	A	B	C	D
1	Порядковый номер	Сумма	Продавец	Общая сумма
2	1	7	Михайлова Нина	
3	2	5	Кошев Владимир	
4	3	6	Мазин Александр	
5	4	1	Кошев Владимир	
6	5	2	Михайлова Нина	
7	6	3	Шевчук Иван	
8	7	4	Кошев Владимир	
9	8	5	Кошев Владимир	
10	9	7	Михайлова Нина	
11	10	2	Мазин Александр	
12	11	12	Мазин Александр	

Рис. 27

Необходимо оформить лист так, чтобы в столбце D получить общую сумму счетов, выписанных каждым продавцом.

**Примечание.** Приведены условные значения.

**Решение**

Если бы требовалось определить общую сумму счетов, например, Михайловой Нины, то это можно было бы сделать по формуле:

$$={СУММ(ЕСЛИ(С2:С12=С2;В2:В12;0))}$$

Если бы для нее требовалось определить сумму ее счетов нарастающим итогом, то диапазон суммирования должен быть для каждой строки разный. Этого можно добиться, используя для адресов ячеек C2 и B2 абсолютную ссылку:

$$={СУММ(ЕСЛИ($C$2:С2=$C$2;$B$2:В2;0))}$$

При этом последняя формула должна быть введена сначала только в ячейку D2, а потом распространена (скопирована) на остальные ячейки столбца D. Результат показан на рис. 28.

	A	B	C	D
1	Порядковый номер	Сумма	Продавец	Общая сумма
2	1	7	Михайлова Нина	7
3	2	5	Кошев Владимир	7
4	3	6	Мазин Александр	7
5	4	1	Кошев Владимир	7
6	5	2	Михайлова Нина	9
7	6	3	Шевчук Иван	9
8	7	4	Кошев Владимир	9
9	8	5	Кошев Владимир	9
10	9	7	Михайлова Нина	16
11	10	2	Мазин Александр	16
12	11	12	Мазин Александр	16

Рис. 28

Для решения основной задачи осталось заменить ссылку \$C\$2 на относительную C2 (чтобы для каждой строки использовалась очередная фамилия и суммирование происходило именно для нее):

$$={СУММ(ЕСЛИ($C2:С2=С2;$B$2:В2;0))}$$

Эта формула вводится только в ячейку D2, а потом распространяется (копируется) на остальные ячейки столбца D. Результат представлен на рис. 29.

	A	B	C	D
1	Порядковый номер	Сумма	Продавец	Общая сумма
2	1	7	Михайлова Нина	7
3	2	5	Кошев Владимир	5
4	3	6	Мазин Александр	6
5	4	1	Кошев Владимир	6
6	5	2	Михайлова Нина	9
7	6	3	Шевчук Иван	3
8	7	4	Кошев Владимир	10
9	8	5	Кошев Владимир	15
10	9	7	Михайлова Нина	16
11	10	2	Мазин Александр	8
12	11	12	Мазин Александр	20

Рис. 29

2. На листе (см. рис. 30), в столбце D, будут записываться результаты соревнований по прыжкам в длину 12 спортсменов из трех школ.

	А	В	С	Д	Е
1	Регистрационный номер	Фамилия	Номер школы	Результат, см	Лучший результат по школе
2	21	Белозеров	1		
3	42	Тымчук	3		
4	3	Алешин	2		
5	14	Крюков	2		
6	15	Пак	1		
7	26	Зозуля	3		
8	7	Волков	3		
9	18	Баков	1		
10	9	Петренко	3		
11	19	Яковлев	2		
12	11	Потапов	1		
13	5	Приходько	2		

Рис. 30

Подготовить лист так, чтобы после ввода результата очередного спортсмена в столбце Е показывался лучший результат среди уже записанных участников соревнований из его школы.

*Решение*

Задача решается аналогично предыдущей, с той разницей, что вместо суммирования проводится поиск максимальных значений. “Базовая” для копирования формула в ячейке E2:

=МАКС(ЕСЛИ(\$C\$2:C2=C2;\$D\$2:D2;0))

3. На листе, аналогичном показанному на рис. 30, в столбце D будут записываться результаты соревнований по бегу на 800 м двенадцати спортсменов из трех школ.

Подготовить лист так, чтобы после ввода результата очередного спортсмена в столбце Е показывался лучший результат среди уже записанных участников соревнований из его школы.

*Решение*

Используется функция МИН. “Базовая” для копирования формула в ячейке E2:

=МИН(ЕСЛИ(\$C\$2:C2=C2;\$D\$2:D2;0))

4. На листе (см. рис. 31) представлена информация об участниках районных олимпиад.

	А	В	С	Д	Е	
1	№	Фамилия, имя	Номер школы	Место	Предмет	
2	1	Агасарян Арман		32	2	Математика
3	2	Борисенко Степан		12	–	Биология
4	3	Волкова Анна		32	1	Химия
...						
218	217	Якубовский Лев		44	1	Математика
219						
220	<b>Статистика</b>					
	Номер школы	Всего участников	1-е место	2-е место	3-е место	
221						
222	3					
223	12					
224	44					
...						

Рис. 31

Подготовить лист для определения по каждой школе общего числа участников олимпиад и числа участников, занявших призовые места.

*Решение*

Формула в ячейке C222:

{=СУММ(ЕСЛИ(ЕСЛИ(\$C\$2:\$C\$218=A222;1)+ЕСЛИ(\$D\$2:\$D\$218=1;1)=2;1;0))}

распространяется (копируется) на остальные ячейки столбца С (для других школ).

Аналогично — с формулами в ячейках D222 и E222:

{=СУММ(ЕСЛИ(ЕСЛИ(\$C\$2:\$C\$218=A222;1)+ЕСЛИ(\$D\$2:\$D\$218=2;1)=2;1;0))}

{=СУММ(ЕСЛИ(ЕСЛИ(\$C\$2:\$C\$218=A222;1)+ЕСЛИ(\$D\$2:\$D\$218=3;1)=2;1;0))}

Значение в ячейке B222 может быть получено без использования табличной формулы (можно применить функцию СЧЕТЕСЛИ).

Литература

1. *Ботт Эд, Леонард Вуди.* Использование Microsoft Office XP. М.: Издательский дом “Вильямс”, 2002.

2. *Лавренов С.М.* Excel. Сборник примеров и задач. М.: Финансы и статистика, 2001.

Ответы к некоторым задачам для самостоятельной работы учащихся

4. Надо использовать два диапазона (A1:A9 и A2:A10) и получить массив, состоящий из сумм соседних чисел:

$$\{=A1:A19+A2:A20\}$$

(Здесь A1 складывается с A2, A2 — с A3, ..., A19 — с A20.)

9.  $\{=\text{КОРЕНЬ}((D2:D11-B2:B11)^2+(E2:E11-C2:C11)^2)\}$

10. После расчета длин сторон (как в предыдущей задаче) находится их сумма:

$$\{=\text{СУММ}(\text{КОРЕНЬ}((D2:D4-B2:B4)^2+(E2:E4-C2:C4)^2))\}$$

11. Идея решения заключается в том, чтобы четным числам сопоставить 1, а нечетным — 0. Один из возможных вариантов — расчет абсолютной величины разности остатка от деления числа на 2 и единицы:

$$\{=\text{СУММ}(\text{ABS}(\text{ОСТАТ}(A2:A25;2)-1))\}$$

12.  $\{=\text{СУММ}(\text{ЕСЛИ}(B3:B12>C3:C12;1;0))\}$

13.

а)  $\{=\text{СУММ}(\text{ЕСЛИ}(B2:B11>C2:C11;1;0))\}$ ;

б)  $\{=\text{СУММ}(\text{ЕСЛИ}(B2:B11=C2:C11;1;0))\}$ ;

в)  $\{=\text{СУММ}(\text{ЕСЛИ}(B2:B11<C2:C11;1;0))\}$ .

14.  $\{=\text{СУММ}(\text{ЕСЛИ}(\text{ОСТАТ}(A2:N2;10)=0;1;0))\}$

15.  $\{=\text{СУММ}(\text{ЕСЛИ}(C2:C27+D2:D27=10;1;0))\}$

16.

а)  $\{=\text{СУММ}(\text{ЕСЛИ}(A1:A20=\text{МАКС}(A1:A20);1;0))\}$ ;

б)  $\{=\text{СУММ}(\text{ЕСЛИ}(A1:A20=\text{МИН}(A1:A20);1;0))\}$ .

**Примечание.** Задачи могут быть решены без использования табличной функции при числе ячеек для решения, равном 1.

17. Число является квадратом целого числа, если его квадратный корень и целая часть последнего равны. Результат сравнения этих значений даст величину индикатора для каждого числа.

$$=\text{СУММ}(\text{ЕСЛИ}(\text{КОРЕНЬ}(A1:N1)=\text{ЦЕЛОЕ}(\text{КОРЕНЬ}(A1:N1));1;0))$$

18. Надо получить массив, состоящий из сумм соседних чисел: A1:A19+A2:A20 (см. задачу 4 для самостоятельной работы). После этого каждый его элемент сравнивается с числом в ячейке C1 с использованием индикаторов, которые складываются:

$$\{=\text{СУММ}(\text{ЕСЛИ}(A1:A19+A2:A20=C1;1;0))\}$$

19. Следует использовать три диапазона:

$$\{=\text{СУММ}(\text{ЕСЛИ}(B1:M1<(A1:L1+C1:N1)/2;1;0))\}$$

20. Основные этапы решения задачи:

1. Получить девять чисел — разностей соседних чисел.

2. Определить количество положительных из них.

3. В зависимости от этого количества получить ответ.

Фрагмент листа, на котором задача решается без использования табличных функций, показан на ниже.

	A	B	C	D
1	1		Разность	Больше 0?
2	2			1 1
3	3			1 1
4	8			5 1
5	12			4 1
6	13			1 1
7	5			1 0
8	15			1 1
9	19			4 1
10	20			1 1
11			Всего положительных разностей	8
12			Ответ:	Нет, не является

Число ячеек для решения задачи — 19.

При использовании табличных функций требуется только одна (! — еще раз ☺) формула:

$$\{=\text{ЕСЛИ}(\text{СУММ}(\text{ЕСЛИ}(A2:A10-A1:A9>0;1;0))=9; \text{“Да, является”}; \text{“Нет, не является”})\}$$

В ней формируется массив, состоящий из разностей соседних чисел: A2:A10-A1:A9 (здесь из A2 вычитается A1, из A3 – A2, ..., из A10 – A9). После этого получается массив индикаторов положительных разностей (ЕСЛИ(A2:A10-A1:A9>0;1;0)), подсчитывается количество ненулевых индикаторов в массиве индикаторов (СУММ(ЕСЛИ(A2:A10-A1:A9>0;1;0))). Если это количество равно девяти (общее количество чисел последовательности, уменьшенное на 1), то последовательность — возрастающая, иначе — нет.

Так как приведенная формула достаточно громоздкая и трудночитаемая, то ее целесообразно получить не сразу, а, например, предварительно подсчитав количество положительных разностей:

$$\{=\text{СУММ}(\text{ЕСЛИ}(A2:A10-A1:A9>0;1;0=9))\}$$

после чего скопировать последнюю формулу (без знака равенства) в качестве первого аргумента “наружной” функции ЕСЛИ.

21. Вместо числа девять в формуле следует использовать функцию СЧЕТ, возвращающую количество чисел в диапазоне (возвращаемое ею значение уменьшается на 1):

$$=\text{ЕСЛИ}(\text{СУММ}(\text{ЕСЛИ}(A2:A10-A1:A9>0;1;0))=\text{СЧЕТ}(A1:A10)-1; \text{“Да, является”}; \text{“Нет, не является”})$$

22.  $\{=\text{СЧЕТ}(\text{ЕСЛИ}(B3:B27>\text{СРЗНАЧ}(B3:B27); B3:B27;””))\}$

23.  $\{=\text{СУММ}(\text{ЕСЛИ}(C2:C8>(\text{МАКС}(C2:C8)+\text{МИН}(C2:C8))/2;1;0))\}$

24.  $\{=\text{СУММ}(\text{ЕСЛИ}(\text{ЕСЛИ}(A1:L1>0;1;0)+\text{ЕСЛИ}(\text{ОСТАТ}(A1:L1;2)=0;1;0)=2;1;0))\}$

25.  $\{=\text{СУММ}(\text{ЕСЛИ}(\text{ЕСЛИ}(\text{ОСТАТ}(A1:N1;3)=0;1;0)+\text{ЕСЛИ}(\text{ОСТАТ}(A1:N1;10)=0;1;0)=2;1;0))\}$

26.  $\{=\text{СУММ}(\text{ЕСЛИ}(\text{ЕСЛИ}(B2:B13<10;1;0)+\text{ЕСЛИ}(B2:B13>90;1;0)>0;1;0))\}$

27.

a) {=СУММ(ЕСЛИ(ЕСЛИ(B2:B26=5;1;0)+  
ЕСЛИ(C2:C26=5;1;0)>0;1;0))}

См. формулу (7);

б) {=СУММ(ЕСЛИ(ЕСЛИ(B2:B26>3;1;0)+  
ЕСЛИ(C2:C26>3;1;0)=2;1;0))}

См. формулу (6);

в) {=СУММ(ЕСЛИ(ЕСЛИ(B2:B26=5;1;0)+  
ЕСЛИ(C2:C26=5;1;0)=1;1;0))}

Обратим внимание на условие ... = 1.

28. {=СУММ(ЕСЛИ(A1:L1=1; A1:L1;0))}

29. {=СУММ(ЕСЛИ(ЕСЛИ(ОСТАТ(A1:L1;7)=0;1;0)+  
ЕСЛИ(ОСТАТ(A1:L1;10)=1;1;0)=2;1;0))}

30. {=СУММ(ЕСЛИ(A2:A16=B18;B2:B18;0))}

32. {=СУММ(ЕСЛИ(C3:C27=C29;D3:D27;0))}

— где C29 — адрес ячейки, в которой задается номер параллели.

33. {=СУММ(ЕСЛИ(A2:A13=D15;B2:B13\*C2:C13\*D2:D13;0))}

34. {=СУММ(ЕСЛИ(A2:A23=D25;B2:B23\*C2:C23/2;0))}

## Табличные формулы и имена диапазонов

В электронных таблицах в формулах можно использовать не только адреса отдельных ячеек или диапазонов, но и их имена. Это облегчает понимание формул. Например, формула нахождения корня линейного уравнения, оформленная в виде:

=ЕСЛИ(a<>0;"";-b/a),

где *a* и *b* — имена ячеек, в которых задаются значения коэффициентов уравнения *a* и *b* соответственно, выглядит гораздо понятнее, чем

=ЕСЛИ(B2<>0;"";-B3/B2),

а формулу =СУММ(Продано\_в\_первом\_квартале) легче понять, чем такую: =СУММ(C20:C138).

Указанную возможность, естественно, можно использовать и в табличных формулах, благодаря чему их размер сокращается, а “читаемость” улучшается.

Вспомним задачу из примера 2: “Фирма имеет десять магазинов. Подготовить лист для определения прибыли каждого магазина по известной информации о доходах и расходах (см. рис. П1.1)”.

	A	B	C	D
1	Магазин №	Доход	Расход	Прибыль
2	1			
3	2			
...				
11	10			

Рис. П1.1

Если диапазону B2:B11 присвоить имя *Доход*, а диапазону C2:C11 — имя *Расход*, то табличная формула, с помощью которой решается задача, примет вид:

{=Доход-Расход}

Самый простой способ присвоения имени ячейке или диапазону ячеек:

- 1) выделить требуемую ячейку или диапазон ячеек;
- 2) щелкнуть мышью в поле *Имя*, расположенном у левого края строки формул (где обычно отображается адрес ячейки);
- 3) ввести в нем имя для назначения выделенным ячейкам;
- 4) нажать клавишу .

После этого при установке указателя активной ячейки на ячейку с именем в поле *Имя* будет отображаться соответствующее имя.

Чтобы найти ячейку или диапазон по имени, следует нажать кнопку выпадающего списка справа от поля *Имя* и щелкнуть мышью на нужном имени в списке.

*Некоторые требования к именам:*

- 1) первым символом имени должна быть буква;
- 2) в именах нельзя использовать пробелы (рекомендуется заменять символом подчеркивания);
- 3) можно использовать точку (например, Квартал.1);
- 4) нельзя присваивать имя, совпадающее с адресами ячеек;
- 5) регистр букв не различается.

Использовать имя ячейки/диапазона можно так:

- 1) при вводе формулы на вкладке **Формулы** в группе **Определенные имена** выбрать — **Использовать** в формуле (см. рис. П1.2);
- 2) в появившемся окне выбрать нужное имя.

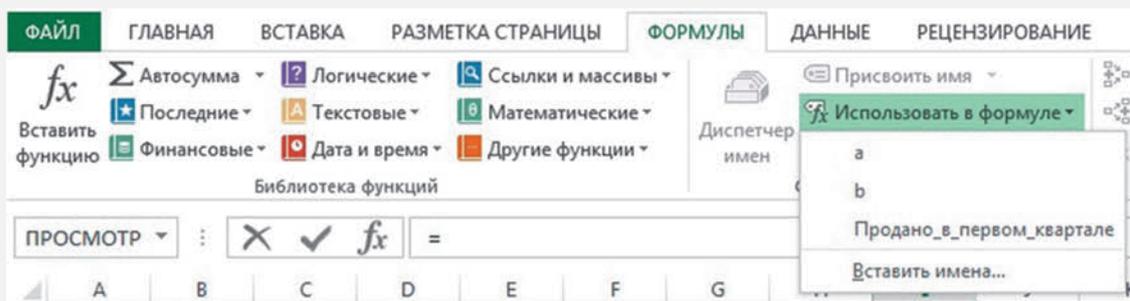


Рис. П1.2

Более быстрый способ — использование вместо меню “горячей” клавиши **F3**. При этом появится окно **Вставка имени** (см. рис. П1.3).

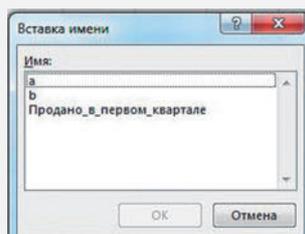


Рис. П1.3

Например, для ввода формулы (условной)

$$=-b/a$$

после набора начала формулы:

$$=-$$

можно нажать **F3**, выбрать имя *b*, щелкнуть на кнопке <ОК>, набрать знак “/”, нажать **F3**, выбрать имя *a*, щелкнуть на кнопке <ОК>, после чего закончить ввод формулы нажатием клавиши **Enter**.

В заключение еще раз скажем, что использование имен сокращает длину формулы, повышает наглядность и уменьшает риск ошибок. Для подтверждения этого приведем ответы к ряду задач без использования имен диапазонов и с ними.

**Пример 1**

{=B2:B100\*B102}  
{=цена\*курс}

**Задача 9**

{=КОРЕНЬ((D2:D11-B2:B11)^2+(E2:E11-C2:C11)^2)}  
{=КОРЕНЬ((x2- x1)^2+(y2-y1)^2)}

**Задача 10**

{=СУММ(КОРЕНЬ((D2:D4-B2:B4)^2+(E2:E4-C2:C4)^2))}  
{=СУММ(КОРЕНЬ((x2- x1)^2+(y2-y1)^2))}

**Задача 12**

{=СУММ(ЕСЛИ(B3:B12>C3:C12;1;0))}  
{=СУММ(ЕСЛИ(она>соперн;1;0))}

**Задача 13**

а)  
{=СУММ(ЕСЛИ(B2:B11>C2:C11;1;0))}  
{=СУММ(ЕСЛИ(заб>проп;1;0))}

**Задача 15**

{=СУММ(ЕСЛИ(C2:C27+D2:D27=10;1;0))}  
{=СУММ(ЕСЛИ(алг+geo=10;1;0))}

**Задача 19**

{=ЕСЛИ(СУММ(ЕСЛИ(A2:A10-A1:A9>0;1;0))=9; “Да, является”; “Нет, не является”)}  
{=ЕСЛИ(СУММ(ЕСЛИ(очер-пред;1;0))=9; “Да, является”; “Нет, не является”)}

**Пример 8**

{=СУММ(ЕСЛИ(B3:B27>СРЗНАЧ(B3:B27);1;0))}  
{=СУММ(ЕСЛИ(числа>СРЗНАЧ(числа);1;0))}

**Задача 21**

{=СУММ(ЕСЛИ(C2:C8>(МАКС(C2:C8)+МИН(C2:C8))/2;1;0))}  
{=СУММ(ЕСЛИ(числа>(МАКС(числа)+МИН(числа))/2;1;0))}

**Пример 10**

{=СУММ(ЕСЛИ(ЕСЛИ(B2:B13>20;1;0)+ЕСЛИ(B2:B13<100;1;0)=2;1;0))}  
{=СУММ(ЕСЛИ(ЕСЛИ(осад>20;1;0)+ЕСЛИ(осад<100;1;0)=2;1;0))}

**Пример 11**

{=СУММ(ЕСЛИ(ЕСЛИ(B2:B26=4;1;0)+ЕСЛИ(C2:C26=4;1;0)=2;1;0))}  
{=СУММ(ЕСЛИ(ЕСЛИ(алг=4;1;0)+ЕСЛИ(гео=4;1;0)=2;1;0))}

**Пример 13**

{=СУММ(ЕСЛИ(B2:B27=C29;C2:C27;0))}  
{=СУММ(ЕСЛИ(фам=кто;ответ;0))}

**Задача 31**

{=СУММ(ЕСЛИ(A2:A13=D15;B2:B13\*C2:C13\*D2:D13;0))}  
{=СУММ(ЕСЛИ(номер=D15;I\*b\*h;0))}

**Задача 32**

{=СУММ(ЕСЛИ(A2:A23=D25;B2:B23\*C2:C23/2;0))}  
{=СУММ(ЕСЛИ(номер=D25;a\*h/2;0))}

**Когда табличные формулы не нужны**

Имеются задачи, которые на первый взгляд должны решаться с применением табличных формул, однако на самом деле это не обязательно. Приведем ряд примеров.

1. На листе записаны длины сторон 20 квадратов:

	A	B
1	Длина, см	
2	12,4	
3	45,5	
...		
21	33,7	
22		
13	Общая площадь	

Подготовить лист на нахождение в ячейке B13 общей площади всех фигур.

Здесь вместо табличной формулы:  
{=СУММ(A2:A21\*A2:A21)}

— можно применить функцию СУММКВ (она возвращает сумму квадратов аргументов):  
=СУММКВ(A2:A11;B2:B11).

2. На листе записаны размеры десяти прямоугольников:

	A	B
1	Длина	Высота
2	12,4	4,9
3	45,5	12,1
...		
11	33,7	15,8
12		
13	Общая площадь	

Подготовить лист на нахождение в ячейке B13 общей площади всех прямоугольников.

Вместо табличной формулы:  
{=СУММ(A2:A11\*B2:B11)}

— можно применить функцию СУММПРОИЗВ:  
=СУММПРОИЗВ(A2:A11;B2:B11).

3. На листе записаны размеры двенадцати параллелепипедов:

	A	B	
1	Длина, см	Ширина, см	Высота, см
2	12,4	10,3	4,9
3	45,5	22,1	12,1
...			
13	33,7	25,4	15,8
14			
15	Общая площадь		

Подготовить лист на нахождение в ячейке B15 общего объема всех тел.

Задача также решается с помощью СУММПРОИЗВ:  
=СУММПРОИЗВ(A2:A13;B2:B13;C2:C13).

4. В теории вероятностей и статистике наиболее распространенный показатель рассеивания значений случайной величины относительно ее математического ожидания называют “стандартным отклонением”. При ограниченных массивах выборок значений вместо математического ожидания используется среднее арифметическое совокупности значений. При наличии ряда случайных величин x стандартное отклонение вычисляется по следующей формуле:

$$\sqrt{\frac{\sum (x - \bar{x})^2}{(n - 1)}}$$

— где  $\bar{x}$  — среднее арифметическое значений.

При выполнении расчетов в электронных таблицах вместо табличной формулы, соответствующей только что приведенной:

=КОРЕНЬ(СУММ((A1:110-CPЗНАЧ(A1:110))^2)/(СЧЁТ(A1:110)-1)),

— где A1:110 — диапазон обрабатываемых значений, можно использовать функцию СТАНДОТКЛОН.В (=СТАНДОТКЛОН.В(A1:110)).

В электронных таблицах имеются и другие стандартные функции для расчета статистических показателей.



## СЕМИНАР

Двоичная система  
и “Ханойские башни”

► Головоломка “Ханойские башни” известна уже довольно давно. Ее автором принято считать французского математика Э.Люка, создавшего головоломку на основе древних легенд. В русской литературе она впервые появилась в 1902 году в книге Е.Игнатьева “В царстве смекалки”.

Если вы возьмете детскую пирамиду (диски располагаются в порядке возрастания: верхний — самый маленький, а нижний — самый большой) и еще два стержня от таких же детских пирамид, то головоломка уже у вас в руках (рис. 1).

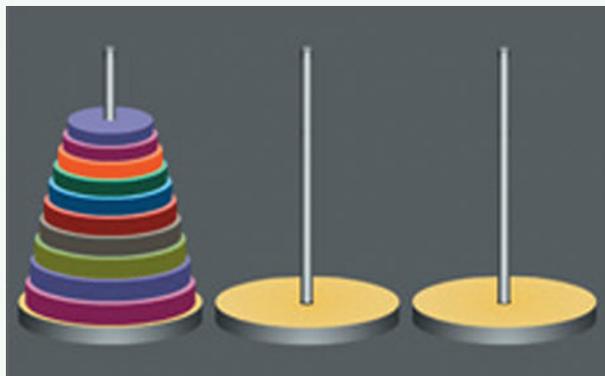


Рис. 1

Пронумеруем стержни: тот, на котором находятся диски, получит номер I, другие — номера II и III. Задача состоит в том, чтобы перенести диски со стержня I на стержень III, используя стержень II как промежуточный. При этом должны соблюдаться три условия:

- 1) за один ход можно переносить лишь один диск;
- 2) нельзя класть больший диск на меньший;
- 3) снятый диск нельзя отложить в сторону — он должен быть надет на один из стержней.

Согласно легенде, в древнем Ханое стоял храм, в котором на одном из трех стержней были надеты 64 золотых диска, и монахи без устали, сменяя друг друга, переносят диски с одного стержня на другой в соответствии с описанными правилами. Когда жрецы перенесут все диски с первого стержня на третий, — гласит легенда, — наступит конец света.

Давайте попробуем справиться с такой головоломкой.

При четырех дисках задача решается легко — ее можно решить, используя, например, монеты разного диаметра и сложив из них пирамидку. Если пронумеровать диски, начиная с самого маленького, 1, 2, 3, 4 и записывать свои действия, то можно получить следующий алгоритм (примем, что диски-монеты нужно перенести с первого “стержня” на третий):

1. 1 → I<sup>1</sup>.
2. 2 → III.
3. 1 → III.
4. 3 → II.
5. 1 → I.
6. 2 → II.
7. 1 → II.
8. 4 → III.
9. 1 → III.
10. 2 → I.
11. 1 → I.
12. 3 → III.
13. 1 → II.
14. 2 → III.
15. 1 → III.

Конечно, можно решить задачу и при большем числе дисков.

А каким должен быть алгоритм не в конкретном, а в общем случае, когда количество дисков равно некоторому числу  $k$ ?

Чтобы ответить на этот вопрос, давайте разберемся, как же перенести диски с первого стержня на третий, а заодно выясним, скоро ли наступит обещанный конец света.

<sup>1</sup> Смысл этого действия, конечно, понятен.

Если бы в пирамиде был только один диск, то решение очевидно — перенесем его на третий стержень, и дело с концом. Мы выполнили требуемое задание за один ход. А если бы было два диска? Тогда положим сначала меньший диск на второй стержень, затем положим второй диск на третий стержень, затем перенесем и меньший диск на третий стержень, положив его поверх второго. Все. За три действия мы смогли переложить оба диска на третий стержень. Отметим, что  $1 = 2^1 - 1$ , а  $3 = 2^2 - 1$ . При трех дисках мы можем сначала перенести два верхних на второй стержень (такую задачу мы решили только что), потом оставшийся нижний (самый большой) диск — на третий стержень, после чего на него же можно перенести те два диска, которые находятся на втором стержне.

Теперь предположим, что мы умеем перекладывать на третий стержень пирамиду из  $n$  дисков за  $2^n - 1$  действий. Покажем, что в таком случае можно перенести на третий стержень и пирамиду из  $n + 1$  дисков, притом за  $2^{n+1} - 1$  действий [1].

Пусть на пирамиде лежит  $n + 1$  дисков. Сначала мы можем перенести  $n$  верхних дисков с первого стержня на второй, произведя  $2^n - 1$  действий. На первом стержне остался лишь один диск. Перенесем его на свободный третий стержень. Теперь у нас на втором стержне лежит  $n$  дисков, первый — свободен, а на третьем лежит самый большой диск. Осталось перенести со второго стержня на третий  $n$  дисков (используя второй стержень как вспомогательный), что мы умеем делать за  $2^n - 1$  операций. Все. Мы собрали на третьем стержне все  $n + 1$  дисков, совершив  $(2^n - 1) + 1 + (2^n - 1) = 2^{n+1} - 1$  действий.

Отсюда, в соответствии с принципом математической индукции, вытекает, что для любого натурального числа  $k$  можно, имея пирамиду с  $k$  дисками, перенести их с первого стержня на третий, соблюдая правила, за  $2^k - 1$  действие. Нетрудно показать, что меньше чем за  $2^k - 1$  действие перенести  $k$  дисков с первого стержня на третий невозможно. Поэтому легендарным жрецам понадобится  $2^{64} - 1$  действий, чтобы исполнить свою работу. Если тратить на каждое действие лишь по одной секунде, то понадобится 18 446 744 073 709 551 615 секунд, или более 500 миллиардов лет (!), то есть волноваться не надо ☺.

Читатели, знакомые с таким понятием программирования, как “рекурсия” [2], конечно, увидели ее здесь. Рекурсивная схема решения задачи следующая:

**если**  $n > 1$

**то**

- переложить  $(n - 1)$  дисков на промежуточный стержень
- переложить самый большой из дисков
- переложить  $(n - 1)$  дисков

на нужный стержень

**иначе**

- переложить этот единственный диск на нужный стержень

**все**

“Но при чем здесь двоичная система счисления?” — спросите вы. Дело в том, что, оказываясь, последовательность перекладывания дисков со стержня на стержень (см. рис. 1) может быть найдена при использовании именно этой системы.

Обозначим диски, начиная с самого маленького, соответственно, 1, 2, 3, ... .

Далее запишем в таблицу десятичные числа от 1 до 7, переведенные в двоичную систему (рис. 2).

$k$	Цифры		
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Рис. 2

Если обозначить разряды двоичных чисел, начиная с крайнего правого, соответственно, 1, 2 и 3 (рис. 3), то последовательность перекладывания дисков, обеспечивающую решение головоломки при трех дисках, можно получить следующим образом: при  $k$ -м ходе перекладывается тот из дисков, номер которого совпадает с номером разряда, в котором появляется “новая”<sup>2</sup> цифра 1 (см. рис. 4).

$k$	Цифры		
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

3 2 1

Рис. 3

$k$	Цифры			Перекладывается диск
1	0	0	1	1
2	0	1	0	2
3	0	1	1	1
4	1	0	0	3
5	1	0	1	1
6	1	1	0	2
7	1	1	1	1

3 2 1

Рис. 4

Предлагаем читателям убедиться в правильности сделанного вывода, а также определить по-

<sup>2</sup> Можно также сказать — самая правая.

следовательность переключивания дисков в случае, когда их 5 (ответы присылайте в редакцию).

У внимательного читателя наверняка возник вопрос — а куда переключивать самый маленький диск, ведь его можно разместить на любом из двух других стержней? (Для других дисков двух вариантов для размещения нет — убедитесь в этом!)

Имеется стратегия выбора хода, всегда приводящая к правильному решению: если общее число дисков нечетно, то последовательность перемещений наименьшего диска имеет вид  $I \rightarrow K \rightarrow P \rightarrow I \rightarrow K \rightarrow P \rightarrow \dots$  (где  $I$  — стержень, на котором диски находятся в исходном положении,  $K$  — стержень,

на котором нужно собрать диски,  $P$  — оставшийся стержень), а если  $n$  четно, то  $I \rightarrow P \rightarrow K \rightarrow I \rightarrow P \rightarrow K \rightarrow \dots$

Компьютерные программы, моделирующие решение головоломки “Ханойские башни”, описаны в рубрике “Школа программирования” в данном выпуске.

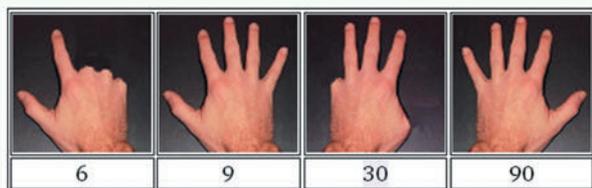
### Литература

1. Савин А.П. Ханойская башня. / Журнал “Квант”, 1991, № 11.
2. Рекурсия — эффективно, но не всегда эффективно. / “В мир информатики” № 188–189 (“Информатика” № 7–8/2014).

## “ЛОМАЕМ” ГОЛОВУ

### Передача числовой информации на пальцах

Иногда требуется передать числовую информацию другому человеку, но беззвучно (например, так называемым “спецназовцам”). Одним из способов решения этой задачи является использование пальцев рук. Существуют различные системы кодирования чисел таким образом. Один из вариантов кодирования для некоторых чисел показан на фотографиях ниже.



А как будут выглядеть в этой системе числа 0, 3, 8, 10, 70? Ответ оформите в виде схематичного рисунка или опишите словесно.

Задача подготовлена по материалам проектной работы Полины Тастиевой, ученицы средней школы села Сейтяково Балтачевского р-на, Республика Башкортостан (научный руководитель — учитель информатики **А.Ф. Загафуранова**). Полностью проект представлен в электронных материалах к данному номеру журнала.

### Переправа

На одном берегу реки собралась компания: отец с двумя сыновьями, мать с двумя дочерьми и шериф с заключенным. Все они хотят переплыть на противоположный берег. При этом:

- 1) детишки не могут одни находиться на плоту;
- 2) шериф не может оставлять заключенного с остальными;
- 3) мужчина не может оставлять ни одного из двух сыновей одних с женщиной, а женщина — своих дочерей с мужчиной;

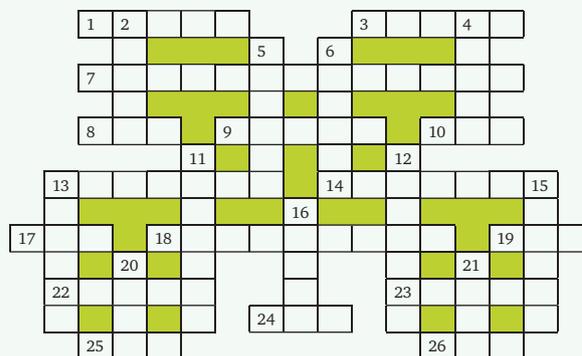
4) естественно, что плот не может плыть сам по себе, а на плоту могут находиться не более двух человек.

Какое минимальное количество раз плот причалит к противоположному берегу, чтобы перевезти всю компанию? Ответ обоснуйте.

По материалам сайта [diofant.ru](http://diofant.ru)

### Кроссворд

Решите, пожалуйста, кроссворд:



По горизонтали

1. Буква греческого алфавита.
3. Последовательность символов, предназначенная для чтения человеком.
7. Специалист, осуществляющий настройку локальной сети и управление ею.
8. Характеристика файла или (в программировании) переменной величины.
9. Цифра девятеричной системы счисления.
10. Задний план, на котором изображается символ на экране или элементы слайда презентации.
13. Множество закономерно связанных между собой объектов, представляющее собой определенное целостное образование.
14. Была милиция, а стала...
17. Цифра шестнадцатеричной системы счисления.
18. Круговая или столбчатая...
19. Структура данных — двусторонняя очередь.
22. Сторона прямоугольного треугольника.

23. Уменьшительное название цифры двоичной системы счисления.

24. Название буквы древнерусского алфавита, напоминающей твердый знак.

25. Буква греческого алфавита, которой, как правило, обозначают неизвестную величину.

26. ...координат.

*По вертикали*

2. Так называют устройство, осуществляющее модулирование и демодулирование данных.

4. Последовательность букв и цифр, ограниченная с обоих концов пробелами, запятыми, точками, дефисами и т.п.

5. Символ, знак.

6. Одна из характеристик абзаца.

11. Место хранения информации в процессоре.

12. Часть массива.

13. Часть таблицы.

15. Элемент электронной таблицы.

16. Комплект символов, воспроизводящий знаки алфавита на экране или на принтере.

20. Структура данных, в которых применен принцип “последним пришел — первым вышел”.

21. Знак арифметической операции.

Ответы (можно не ко всем терминам) присылайте в редакцию.

## Необычная фигура на шахматной доске

**Д.М. Златопольский,**  
Москва

Обсудим такую задачу.

В левом нижнем углу шахматной доски<sup>3</sup> (см. рис. 1) стоит шахматная фигура, которую назовем “полукороль”, так как она, в отличие от “настоящего” шахматного короля, ходит только на одно поле строго вправо, или строго вверх, или строго по диагонали вправо-вверх. Сколько существует способов перемещения фигуры на поле, выделенное желтым цветом?

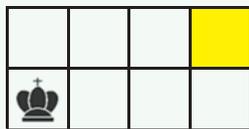


Рис. 1

Эту задачу можно решить, нарисовав “дерево” всех возможных ходов нашей фигуры.

Пронумеруем поля:



Рис. 2

<sup>3</sup> На всех рисунках статьи поля доски черного цвета условно не выделены.

Тогда указанное “дерево” будет иметь вид (числа в кружочках — номера полей):

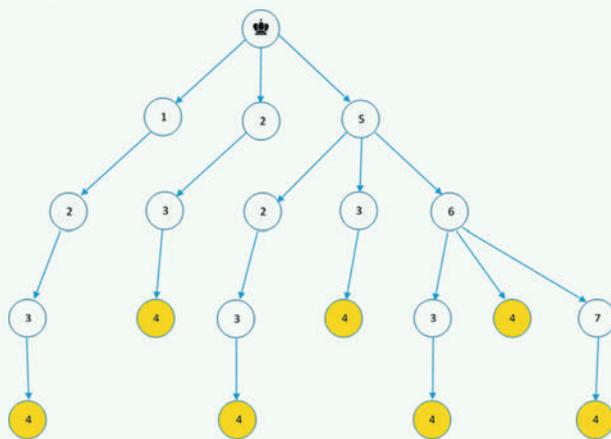


Рис. 3

Видно, что искомое общее число маршрутов достижения желтого (номер 4) поля равно 7.

Однако такой способ решения задачи не очень подходит для случаев, когда полей больше.

Имеются более рациональные способы решения задачи, которые можно использовать для любого числа полей. Один из них заключается в анализе маршрутов, так сказать, “с конца”.

С полей, выделенных на рис. 4 синим цветом, до “цели” полукороль может переместиться единственным способом. Запишем это на рисунке:

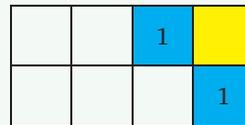


Рис. 4

С красных полей путь также единственный:

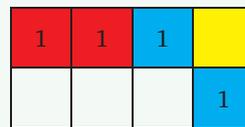


Рис. 5

Обсудим теперь число маршрутов с зеленого поля (рис. 6). С него на желтое поле можно попасть непосредственно или через синие поля. Так как число маршрутов с синих полей нам уже известно (оно записано в этих полях), то общее число путей с зеленого поля до цели равно 1 (непосредственно) + 1 (через верхнее синее поле) + 1 (через правое синее поле) = 3. Запишем результат:

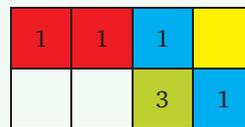


Рис. 6

Продолжая аналогичные рассуждения и записывая рассчитываемые значения, можно получить следующую картину:

1	1	1	
7	5	3	1

Рис. 7

Из рис. 7 видно, что искомое число маршрутов равно 7.

Можно также провести анализ, начиная с полей, соседних с исходным полем:

1	3		
	1		

Рис. 8

Предлагаем читателям провести решение задачи таким способом самостоятельно.

### Задания для самостоятельной работы

1. Нарисуйте граф, моделирующий возможные маршруты фигуры, описанной в статье (граф — конечное число точек на плоскости, соединенных отрезками линий). Используйте номера полей, приведенные на рис. 2. Номера полей, в отличие от рис. 3, на графе повторяться не должны.

2. В правом нижнем углу фрагмента шахматной доски (см. рис. 9) стоит шахматная фигура, которую назовем “полукороль”, так как она, в отличие от “настоящего” короля, ходит только на одно поле строго влево, или строго вверх, или строго по диагонали влево-вверх. Сколько существует способов перемещения фигуры на поле, выделенное желтым цветом?

Рис. 9

3. Подготовьте лист электронной таблицы Microsoft Excel или другой подобной программы, с помощью которого можно определить число способов перемещения полукороля со всех полей шахматной доски на левое верхнее поле (в шахматной нотации — поле a8). Постарайтесь вручную вводить как можно меньше формул (используйте их копирование).

Ответ и лист присылайте в редакцию.

Программа для решения задачи описана в рубрике “Школа программирования” в этом выпуске.

### Числовой ребус с “МАССИВОМ”

Решите, пожалуйста, числовой ребус:

$$MA^c = CIB,$$

в котором, как принято в таких головоломках, одинаковыми буквами зашифрованы одинаковые цифры, разными буквами — разные цифры.

### Ли Ван и Ван Ли

В некоторой стране, жители которой говорят или только правду, или только лгут, имеются два семейства — Ли Ван и Ван Ли, отличающиеся от обычных жителей. Семейство Ли Ван говорит правду только утром и лжет вечером, а семейство Ван Ли — правду только вечером, а ложь утром.

Вы встретили двух человек — А и В, по одному из каждого семейства, но не знаете, кто из них к какому семейству принадлежит. Каждый сделал по одному утверждению:

А: “Сейчас вечер”;

В: “Моя фамилия Ван Ли”.

Сможете ли вы определить, какие семейства представляет каждый из встреченных? Если да, то присылайте ответ в редакцию.

### Число кратчайших путей. Задача 1

На поле шахматной доски, выделенном зеленым цветом (см. рисунок ниже), стоит “настоящий” шахматный король, то есть фигура, которая, в отличие от “полукороля” из предыдущей задачи, ходит по правилам шахмат. Сколько существует маршрутов перемещения короля на желтое поле доски за наименьшее число ходов?


### Число кратчайших путей. Задача 2

Для каждого поля шахматной доски (см. рисунок ниже) определите число кратчайших маршрутов перемещения шахматного короля с этого поля на поле a8 (выделенное желтым цветом). Например, для полей a7, b8 и b7 это число равно 1.


Ответ оформите в виде таблицы 8 × 8.

## Решаем задачу о полукороле средствами программирования

Разработаем программу, с помощью которой можно определить число способов перемещения полукороля (см. статью в рубрике «Ломаем» голову») со всех полей шахматной доски на левое верхнее поле (в шахматной нотации — поле *a8*). Используем метод, основанный на анализе с конца маршрута.

Искомые значения будем хранить в двумерном массиве из восьми строк и восьми столбцов с именем *доска*.

Для полей первого столбца и для первой строки число возможных маршрутов равно 1:

```
|1-й столбец
нц для i от 2 до 8
  доска[i, 1] := 1
кц
|1-я строка
нц для j от 2 до 8
  доска[1, j] := 1
кц
```

Как показано в статье, для остальных полей доски это число может быть определено как сумма значений для соседнего левого поля, для соседнего верхнего поля и для соседнего левого верхнего поля. Исключение составляет второе поле второй строки (убедитесь в этом самостоятельно), поэтому для него значение запишем как известное:

```
доска[2, 2] := 3
Значения для остальных полей доски:
|- во 2-й строке
нц для j от 3 до 8
  доска[2, j] := доска[2, j - 1] +
    доска[1, j] + доска[1, j - 1]
кц
|- в остальных строках
нц для i от 3 до 8
  нц для j от 2 до 8
    доска[i, j] := доска[i, j - 1] +
      доска[i - 1, j] + доска[i - 1, j - 1]
  кц
кц
```

Всю программу на языке программирования, который вы изучаете, соберите самостоятельно (естественно, предусмотрите в ней ввод координат исходного поля и вывод ответа). Программу, пожалуйста, пришлите в редакцию — фамилии всех приславших правильную программу будут опубликованы.

## Программа, моделирующая решение головоломки «Ханойские башни»

Разработаем не рекурсивную программу, в которой на экран выводится только последовательность номеров перекладываемых дисков (см. статью «Двоичная система и «Ханойские башни»» в этом выпуске).

В статье было показано, что такую последовательность можно получить на основе таблиц на рис. 3 и 4 и подобных. Первый вариант нашей программы и будет основан на этом методе.

Сначала научимся определять номер разряда, в котором в двоичной записи того или иного числа появляется «новая» цифра 1 при увеличении номера хода на 1 (см. только что указанные таблицы).

Для хранения двоичных цифр очередного числа (номера очередного перекладывания) в программе используем массив с именем *мас\_нов*. Размер этого массива зависит от количества дисков. Как? Составим таблицу:

Число дисков <i>n</i>	2	3	4	...
Число перекладываний	3	7	15	...
Номера перекладываний	1-3	1-7	1-15	...
Число двоичных разрядов для хранения максимального номера	2	3	4	...

Из нее следует, что размер массива равен общему количеству дисков *n*.

Аналогичный массив потребуется и для хранения двоичных цифр предыдущего числа. Его имя в программе — *мас\_ст* (а имя числа — *ст*).

Если номер очередного числа обозначить *нов*, то фрагмент, относящийся к заполнению указанных массивов, может быть оформлен так:

```
нов := ...
ст := нов - 1
нц для i от 1 до n
  |Выделяем последние двоичные цифры
  |и записываем их в соответствующие
  |массивы (в их начало!)
  мас_нов[i] := mod(нов, 2)
  мас_ст[i] := mod(ст, 2)
  |Определяем целочисленное частное
  нов := div(нов, 2)
  ст := div(ст, 2)
кц
```

После заполнения массивов искомый номер разряда может быть найден с применением оператора цикла с условием:

```
i := 1 |С начала массива, но с конца
  |двоичной записи
нц пока мас_нов[i] = 0
  i := i + 1
кц
```

```
|Значение i - искомое
```

Обращаем внимание на условие, использованное в операторе.

Теперь — об основной программе. В ней надо найти и вывести на экран соответствующие номера разрядов для каждого номера перекладывания:

```
цел n
n := ...
алг Решение_головоломки_Ханойские_башни
  |Вывод номеров перекладываемых дисков
нач цел нов, ст, нов2, i,
```

```

цел таб мас_нов[1:n], мас_ст[1:n]
вывод нс, "Последовательность
    переключиваний дисков: "
|Для каждого номера переключивания
нц для нов от 1 до 2 ** n - 1
    ст := нов - 1
    нов2 := нов |Копия значения величины нов
    |Заполняем массивы мас_нов и мас_ст
нц
    ... (см. выше)
кц
|Ищем номер разряда
i := 1
... (см. выше)
|Выводим его
вывод i, " "
кц
кон

```

**Примечание.** Так как значение величины *нов* в теле оператора цикла меняется, а она является параметром цикла, менять который нельзя, то в программе используется “копия” этой величины (*нов2*).

Размер программы существенно сокращается, если для определения номера разряда, в котором в двоичной записи того или иного числа появляется “новая” цифра 1, применить логические и сдвиговые операции над числами [1]. С их использованием фрагмент, относящийся к нахождению искомого номера разряда для некоторого числа *нов*, может быть оформлен следующим образом:

```

ст := нов - 1
i := 1 |Нумерация разрядов справа
нц пока нов И 1 = 0
    |Отбрасываем последнюю двоичную цифру
    нов := нов СдвигВправоНа 1
    ст := ст СдвигВправоНа 1
    i := i + 1
кц
|Значение i - искомое

```

— где И — логическая операция, СдвигВправоНа — сдвиговая операция<sup>4</sup>, а вся программа, моделирующая переключивание дисков, имеет вид:

```

n := ...
алг Решение_головомки_Ханойские_башни
    |Вывод номеров переключиваемых дисков
нач цел нов, ст, нов2, i
    вывод нс, "Последовательность
        переключиваний дисков:"
нц для нов от 1 до 2 ** n - 1
    ст := нов - 1
    |Копия значения величины нов
    нов2 := нов
    i := 1; |Нумерация разрядов справа
нц пока нов И 1 = 0
    нов2 := нов2 СдвигВправоНа 1

```

<sup>4</sup> В программе на школьном алгоритмическом языке операции И и СдвигВправоНа использованы условно; в языке программирования Паскаль их аналогами являются, соответственно, операции AND и SHR.

```

ст := ст СдвигВправоНа 1
i := i + 1
кц
|Выводим номер найденного разряда
вывод i, " "

```

**кц**

**кон**

И, наконец, самый компактный (и понятный) вариант программы дает прием, называемый “рекурсией” [2]. Напомним, что в программировании рекурсией называют прием, когда какая-то задача сводится к такой же задаче, но с другими исходными данными (в нашем случае — с другим количеством дисков).

Учитывая рассуждения в статье, связанные с переключиванием ( $n + 1$ ) дисков, можем так оформить общую схему алгоритма решения задачи при  $n$  дисках:

```

если n = 1
то
    Переносим единственный диск
    на нужный стержень
иначе
    |Используем этот же алгоритм
    | (рекурсию),
    | но с n - 1 верхними дисками
    Перемещение_дисков((n - 1) дисков)
    Переносим самый большой диск
    |Еще раз используем этот же алгоритм
    | с n - 1 верхними дисками
    Перемещение_дисков((n - 1) дисков)

```

**все**

На школьном алгоритмическом языке соответствующая рекурсивная (использующая рекурсию) процедура выглядит так:

```

алг Перемещение (арг цел n)
    |Вывод номеров переключиваемых дисков
нач
если n = 1
то
    вывод n, " " |Вывод номера 1
иначе
    |Рекурсивный вызов этой же процедуры
    Перемещение(n - 1)
    |Вывод номера самого большого диска
    вывод n, " "
    |Рекурсивный вызов этой же процедуры
    Перемещение(n - 1)

```

**все**

**кон**

Основная часть программы при этом такая:

```

алг Головоломка_Ханойские_башни
нач цел n
    n := ...
вывод нс, "Последовательность
        переключиваний дисков: "
    Перемещение(n)

```

**кон**

Эффектно, не правда ли?

## Задание для самостоятельной работы

Разработайте один или несколько описанных вариантов программы на языке программирования, который вы изучаете, и пришлите их в редакцию. Все приславшие правильные программы будут награждены дипломами.

## ЗАДАЧНИК

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

#### Задача “Кто какой язык изучает?”

Напомним условие: “Вадим, Сергей и Михаил изучают различные иностранные языки: китайский, японский и арабский. На вопрос, какой язык изучает каждый из них, один ответил: “Вадим изучает китайский, Сергей не изучает китайский, а Михаил не изучает арабский”. Впоследствии выяснилось, что в этом ответе только одно утверждение верно, а два других ложны. Какой язык изучает каждый из молодых людей?”

##### Решение

В условии имеются три утверждения:

- 1) Вадим изучает китайский;
- 2) Сергей не изучает китайский;
- 3) Михаил не изучает арабский.

Если верно первое утверждение, то верно и второе, так как юноши изучают разные языки. Это противоречит условию задачи, поэтому первое утверждение ложно. Если верно второе утверждение, то первое и третье должны быть ложны. При этом получается, что никто не изучает китайский. Это противоречит условию, поэтому второе утверждение тоже ложно. Остается считать верным третье утверждение, а первое и второе — ложными. Следовательно, Вадим не изучает китайский, китайский изучает Сергей.

*Ответ:* Сергей изучает китайский язык, Михаил — японский, Вадим — арабский.

##### Правильные ответы представили:

— Абдувахидова Алина, Абдувахидова Софья, Милушкин Дмитрий, Попов Никита, Степанова Мария и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

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

— Аражапова Екатерина, Батаева Анастасия, Волчек Владимир, Вяткина Марина, Ермакова Анастасия, Калугин Даниил, Коковин Станислав, Кононенко Александра, Кошкарова Анастасия, Лютова Александра, Мартынова Кристина, Маслов Дмитрий, Моисеенкова Елена, Молева Александра, Нажипова Виктория, Одинцова Екатерина, Пуговкина Ксения, Рябинина Полина, Салимов Владислав, Сенцов Иван, Хажиев Флорит, Чурасов Михаил и Юрченкова Анна, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

## Литература

1. Логические и сдвиговые операции. / “В мир информатики” № 170 (“Информатика” № 16/2011).
2. Рекурсия — эффективно, но не всегда эффективно. / “В мир информатики” № 188–189 (“Информатика” № 7–8/2014).

— Батулин Илья, Лазуренко Глеб, Пак Александра и Сыроев Александр, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

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

— Булгаков Владислав, Воронков Андрей, Гнедина Ангелина, Дедов Дмитрий, Свиридов Максим, Суворов Александр, Федоров Максим и Яцкова Виктория, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Бык Владилена, Гарифулина Регина, Джавадов Эльшан, Жорова Мария, Ишмухаметова Яна, Лутфуллин Ильмир, Макаров Павел, Михайлов Никита, Мулюков Артур, Мустафина Диана, Мухаметшин Эдгар, Озеров Марсель, Рахимова Лейсан и Хамитов Дамир, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Волков Александр, г. Мытищи Московской обл., школа “Логос”, учитель **Елистратова А.А.**;

— Евграфова Ксения, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

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

— Курбанов Сухраб, Свердловская обл., Красноуфимский р-н, Тавринская средняя школа, учитель **Ярцев В.А.**;

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

— Мазанова Екатерина, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Мусатов Тимофей, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

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

### Задание “Три (и даже больше) вопроса” (рубрика “Поиск информации”)

##### Ответы

1. Битва, произошедшая 15 июля 1410 года, — это Грюнвальдская битва. В ней принимали участие представители одного кавказского народа — армяне (обратим внимание, что в вопросе шла речь именно о кавказском народе).

2. Единственное в мире соревнование автомобильных команд Формулы-1, проведение которого занимает не три, а четыре дня, проходит в Монако.

3. Судно, показанное на рисунке в задании, — это атомный ледокол “Россия”. Оно имеет три винта. Высота борта 17,2 м.

*Ответы прислали:*

— Бородюк Анна, Макачук Елена и Страхова Елизавета, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Бык Владилена, Гарифулина Регина, Жорова Мария, Ишмухаметова Яна, Михайлов Никита, Мустафина Диана и Рахимова Лейсан, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Васильев Андрей и Евграфова Ксения, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

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

— Дощик Константин, Лазуренко Глеб, Нетесов Николай, Пак Александра, Пантелюк Руслан, Приходько Геннадий, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Казанец Елена, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Коробов Сергей, Маркова Алевтина и Яснова Дарья, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Музалевская Екатерина, г. Орел, лицей № 4 им. Героя Советского Союза Г.Б. Злотина, учитель **Чапкевич И.М.**;

— Прошунин Юрий, Алтайский край, г. Рубцовск, школа № 1, учитель **Мусихин А.И.**;

— Салимов Владислав, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Удалова Елизавета, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

Заметим, что в приведенном списке указаны также читатели, представившие ошибочный ответ (ошибочные ответы).

## Задача “Цветные удочки”

Напомним условие: “Пятеро друзей — Дима, Тима, Фима, Сима и Клим — решили купить себе удочки. Удочки в магазине были пяти цветов: красного, синего, белого, зеленого и черного.

Известно, что:

- 1) Дима любит красный и синий цвета;
- 2) Симе понравились синяя и зеленая удочки;
- 3) Фима купил зеленую удочку;
- 4) Клим отдал предпочтение красной, синей и черной удочкам.

Кто какую удочку купил, если у всех ребят оказались удочки разного цвета?”

Задача может быть решена методом таблиц.

*Ответ:* Дима купил красную удочку, Тима — белую, Фима — зеленую, Сима — синюю, Клим — черную.

*Правильные ответы представили:*

— Аликеева Екатерина, Свердловская обл., Красноуфимский р-н, Тавринская средняя школа, учитель **Ярцев В.А.**;

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

— Аражапова Екатерина, Батаева Анастасия, Волчек Владимир, Вяткина Марина, Ермакова Анастасия, Калугин Даниил, Коковин Станислав, Кононенко Александра, Кошкарова Анастасия, Лютова Александра, Мартынова Кристина, Маслов Дмитрий, Моисеенкова Елена, Молева Александра, Нажипова Виктория, Одинцова Екатерина, Пуговкина Ксения, Рябинина Полина, Салимов Владислав, Сенцов Иван, Хажиев Флорит, Чурасов Михаил и Юрченкова Анна, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Батурин Илья, Лазуренко Глеб, Пак Александр и Сысоев Александр, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

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

— Булгаков Владислав, Воронков Андрей, Гнедина Ангелина, Дедов Дмитрий, Свиридов Максим, Суворов Александр, Федоров Максим и Яцкова Виктория, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Бык Владилена, Гарифулина Регина, Жорова Мария, Ишмухаметова Яна, Лутфуллин Ильмир, Михайлов Никита, Мулюков Артур, Мустафина Диана, Мухаметшин Эдгар, Рахимова Лейсан и Хамитов Дамир, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Волков Александр, г. Мытищи Московской обл., школа “Логос”, учитель **Елистратова А.А.**;

— Евграфова Ксения, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Завадский Вадим, Загребина Дарья, Мусатов Максим и Мусатов Тимофей, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

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

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

— Мазанова Екатерина, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Милушкин Дмитрий, Попов Никита, Степанова Мария и Хозин Марат, Чуб Алексей, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Никонов Михаил, г. Рубцовск Алтайского края, школа № 1, учитель **Воронова Т.Н.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;  
 — Хорькова Анна, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

## Числовые ребусы в троичной системе. Часть 7

Напомним, что требовалось решить три ребуса, в которых зашифрованы примеры вычислений в троичной системе счисления:

1.

$$\begin{array}{r} \phantom{+} \phantom{M} \phantom{1} \\ + \phantom{M} \phantom{1} \\ \hline * \phantom{*} * \phantom{*} \end{array}$$

2.

$$\begin{array}{r} \phantom{+} \phantom{A} \phantom{B} \\ + \phantom{A} \phantom{B} \\ \hline * \phantom{*} \end{array}$$

3.

$$\begin{array}{r} \phantom{+} \phantom{C} \phantom{D} \\ + \phantom{C} \phantom{D} \\ \hline * \phantom{0} * \end{array}$$

*Ответы прислали:*

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

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

— Милушкин Дмитрий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**

Учитывая небольшое число приславших ответы ☺ и важность рассматриваемой темы ☺, остановимся на решении подробно:

*Ребус 1*

Так как результат трехзначный, то его первая цифра, равная сумме двух цифр **М**, равна 1 (а **М** = 2):

$$\begin{array}{r} \phantom{+} \phantom{2} \phantom{1} \\ + \phantom{2} \phantom{1} \\ \hline 1 \phantom{1} \phantom{2} \end{array}$$

*Ребус 2*

Здесь крайняя левая цифра результата равна 2 (при **А** = 1 и отсутствии переноса в “уме” из крайнего правого разряда). Так как **В** ≠ **А**, то **А** может быть равно только 0:

$$\begin{array}{r} \phantom{+} \phantom{1} \phantom{0} \\ + \phantom{1} \phantom{0} \\ \hline 2 \phantom{0} \end{array}$$

*Ребус 3*

Поскольку результат — трехзначный, то его первая цифра равна 1. Далее, средняя цифра результа-

та может быть равна нулю только при **С** = 1 и переносе единицы в “уме” из крайнего правого разряда, то есть **Д** = 2:

$$\begin{array}{r} \phantom{+} \phantom{1} \phantom{2} \\ + \phantom{1} \phantom{2} \\ \hline 1 \phantom{0} \phantom{1} \end{array}$$

## Задача “Сколько лет детям?”

Напомним условие: “В семье четверо детей — Аня, Боря, Вера и Галя. Им 5, 8, 13 и 15 лет. Сколько лет каждому ребенку, если одна девочка ходит в детский сад, Аня старше Бори, и сумма лет Ани и Веры кратна трем?”

*Решение*

На основе информации из условия можно составить таблицу:

	5	8	13	15
Аня	–	–		
Боря	–			–
Вера				
Галя				

**Комментарии**

1. Так как Аня старше мальчика Бори, то она не может ходить в детский сад и ей не может быть 8 лет.

2. Мальчик Боря не может ходить в детский сад и не самый старший ребенок.

Из таблицы следует, что Ане 13 или 15 лет. В последнем случае сумма ее возраста и возможного возраста Веры (5, 8 или 13 лет) не кратна трем. Значит, Ане 13 лет. Тогда Боре — 8 лет.

Оставшиеся значения 5 и 15 лет. Из них 15 лет для Веры не подходят (сумма 13 + 15 = 28 — не кратна трем). Значит, Вере 5 лет, а Гале — 15.

*Правильные ответы представили:*

— Абдувахидова Алина, Абдувахидова Софья, Милушкин Дмитрий, Попов Никита, Степанова Мария и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Алпатов Никита, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Аражапова Екатерина, Батаева Анастасия, Волчек Владимир, Вяткина Марина, Ермакова Анастасия, Калугин Даниил, Коковин Станислав, Кононенко Александра, Кошкарова Анастасия, Лютова Александра, Мартынова Кристина, Маслов Дмитрий, Моисеенкова Елена, Молева Александра, Нажипова Виктория, Одинцова Екатерина, Пуговкина Ксения, Рябинина Полина, Салимов Владислав, Сенцов Иван, Хажиев Флорит, Чурасов Михаил и Юрченкова Анна, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Батурин Илья, Лазуренко Глеб, Пак Александр и Сысоев Александр, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

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

— Булгаков Владислав, Воронков Андрей, Гнедина Ангелина, Дедов Дмитрий, Свиридов Максим, Суво-

ров Александр, Федоров Максим и Яцкова Виктория, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Бык Владилена, Гарифулина Регина, Жорова Мария, Ишмухаметова Яна, Кудряшова Александра, Михайлов Никита, Мустафина Диана, Рахимова Лейсан и Хамитов Дамир, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Дежкина Мария, г. Рубцовск Алтайского края, школа № 1, учитель **Воронова Т.Н.**;

— Завадский Вадим, Мусатов Максим и Мусатов Тимофей, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Зыков Сергей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Иванов Денис, Свердловская обл., Красноуфимский р-н, Тавринская средняя школа, учитель **Ярцев В.А.**;

— Казанец Елена, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Мазанова Екатерина, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

— Удалова Елизавета, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

### Четыре задачи на перестановки цифр

Напомним условие задач.

1. Дано число  $15_q$ , записанное в некоторой системе счисления  $q$ . В какой системе при перестановке его цифр значение увеличивается в три раза?

2. Какое двузначное число, записанное в 9-ричной системе счисления, при перестановке его цифр увеличивается:

- а) в 4 раза;
- б) в 1,5 раза?

3. Есть ли такое восьмеричное число, которое при перестановке его цифр увеличивается в 1,25 раза?

*Решения*

Следует использовать так называемую “развернутую” запись исследуемых чисел.

1. Десятичный эквивалент заданного числа (в  $q$ -ичной системе счисления) равен  $q + 5$ , а числа, полученного при перестановке его цифр, —  $5q + 1$ . По условию:

$$\frac{5q+1}{q+5} = 3,$$

откуда  $q = 7$ .

*Ответ:* в семеричной.

2.

а) если искомое число имеет вид  $ab$ , то можем записать:

$$\frac{9b+a}{9a+b} = 4,$$

откуда  $5b = 35a$  или  $b = 7a$ . Так как  $b < 9$ , то имеем единственное решение:  $b = 7, a = 1$ .

*Ответ:* 17.

б)

*Ответ:* 35.

3. Есть — это число 34 ( $34_8 = 28_{10}; 43_8 = 35_{10}$ ).

*Ответы прислали:*

— Алимов Евгений, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Байрамалов Сергей, Свердловская обл., Красноуфимский р-н, Тавринская средняя школа, учитель **Ярцев В.А.**;

— Батулин Илья и Пак Александра, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Виктюк Степан и Прохоров Игорь, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

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

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Дибров Сергей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

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

— Торопов Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**

### Задача “Площадь комнаты”

Напомним условие: “Житель планеты  $\acute{a}$ -Кентавр пишет, что размеры его комнаты  $4 \times 12$  м, а площадь равна 53 квадратных метра. Как такое могло быть?”

*Решение*

Такое могло быть только в случае, когда применяется недесятичная система счисления. Но какая?

Здесь также задача решается с использованием развернутой записи заданных чисел. Обозначим основание неизвестной системы —  $q$ . Тогда можем записать:

$$4_q = 4_{10}; 12_q = (q + 2)_{10}; 53_x = (5x + 3)_{10}.$$

Сравним рассчитанную и заданную площади комнаты:

$$4 \times (q + 2) = 5q + 3,$$

откуда  $q = 5$ .

*Ответ:* для записи чисел житель планеты  $\acute{a}$ -Кентавр применял пятеричную систему счисления.

*Правильные ответы прислали:*

— Алимов Евгений, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Батулин Илья и Пак Александра, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

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

— Дибров Сергей, средняя школа поселка Осинька, Алтайский край, учитель **Евдокимова А.И.**;

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

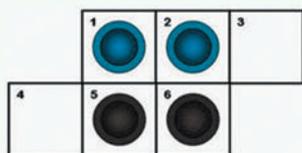
— Стороженко Степан, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Торопов Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

## Головоломка

### “Шесть клеток и четыре фишки”

Напомним, что следовало за минимальное число перемещений поменять местами синие и черные фишки, которые размещены в шести клетках так, как показано на рисунке. Разрешается двигать фишки только на смежное пустое место.



*Решение*

Задача решается за 12 перемещений: 1. Ч–5–4. 2. С–1–5. 3. С–2–1. 4. Ч–6–2. 5. Ч–2–3. 6. С–5–6. 7. Ч–4–5. 8. С–1–2. 9. Ч–5–1. 10. С–6–5. 11. С–2–6. 12. Ч–3–2.

Возможно также “симметричное” решение.

Если же принять, что фишки можно перемещать и по диагонали, то задача упрощается — решение возможно за пять перемещений. Вот один из вариантов: 1. С–1–4. 2. Ч–6–1. 3. С–2–6. 4. Ч–5–2. 5. С–4–5.

*Ответы представили:*

— Алпатов Никита, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

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

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

— Бык Владилена, Гарифулина Регина, Жорова Мария, Ишмухаметова Яна, Лутфуллин Ильмир, Михайлов Никита, Мустафина Диана и Рахимова Лейсан, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Сметанин Тимофей, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

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

— Шахов Игорь, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Щукарев Максим, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

В ряде ответов задача решалась за большее число перемещений — до 16.

## Задача “Компьютер на яхте”

Напомним, что в задаче шла речь о возможном выходе из строя одного, двух или трех блоков компьютера, что определялось по загоранию трех лампочек. Был приведен ряд условий, при которых загорается та или иная лампочка. По ним требовалось определить, какие блоки вышли из строя.

*Ответ:* нужно заменить блоки  $b$  и  $c$ ; блок  $a$  замены не требует.

*Правильный ответ представили:*

— Аражапова Екатерина, Батаева Анастасия, Волчек Владимир, Вяткина Марина, Ермакова Анастасия, Калугин Даниил, Коковин Станислав, Кононенко Александра, Кошкарова Анастасия, Лютова Александра, Мартынова Кристина, Маслов Дмитрий, Моисеенкова Елена, Молева Александра, Нажипова Виктория, Одинцова Екатерина, Пуговкина Ксения, Рябинина Полина, Салимов Владислав, Сенцов Иван, Хажиев Флорит, Чурасов Михаил и Юрченкова Анна, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

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

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

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

— Жорова Мария, Ишмухаметова Яна и Мустафина Диана, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Зеленский Константин, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Торопов Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

— Шаров Станислав, средняя школа поселка Осинька, Алтайский край, учитель **Евдокимова А.И.**

## Задача “Красный куб”

Напомним условие: “Представьте себе деревянный куб со сторонами 30 см, вся поверхность которого окрашена в один красный цвет, и ответьте, пожалуйста, на вопросы:

1) сколько потребуется разрезов, чтобы разделить куб на кубики со стороной 10 см;

2) сколько получится таких кубиков;

3) сколько кубиков будут иметь по четыре окрашенные грани;

4) сколько кубиков будут иметь по три окрашенные грани;

5) сколько кубиков будут иметь по две окрашенные грани;

6) сколько кубиков будут иметь по одной окрашенной грани;

7) сколько кубиков будут неокрашенными?”.

*Ответы*

1) 6 разрезов;

2) 27 кубиков;

3) ни одного;

4) 8 — столько, сколько вершин у куба;

5) 12 — столько, сколько ребер у куба;

6) 6 — столько, сколько граней у куба;

7) 1.

*Правильные ответы прислали:*

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

— Алпатов Никита, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Бык Владилена, Гарифулина Регина, Джавадов Эльшан, Жорова Мария, Ишмухаметова Яна, Лутфуллин Ильмир, Михайлов Никита, Мулюков Артур, Мустафина Диана, Озеров Марсель и Рахимова Лейсан, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Зеленский Константин, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Зыков Сергей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Иванов Алексей, Свердловская обл., Краснофимский р-н, Тавринская средняя школа, учитель **Ярцев В.А.**;

— Климов Андрей, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Лазуренко Глеб, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Милушкин Дмитрий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Михайлов Иван, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**

### Задача “Что сказал старик?”

Напомним, что в старинной задаче речь шла о соревновании, в котором победителем считался тот из двух наездников, чья лошадь придет в назначенное место второй, а не первой.

*Ответ.* Старик сказал наездникам: “Поменяйтесь лошадьми”.

*Ответы прислали:*

— Абдувахидова Алина и Абдувахидова Софья, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Бондаренко Петр и Дибров Сергей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

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

— Бык Владилена, Гарифулина Регина, Жорова Мария, Ишмухаметова Яна, Лутфуллин Ильмир, Михайлов Никита, Мулюков Артур, Мустафина Диана и Рахимова Лейсан, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Зеленский Константин, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

— Хомутов Андрей, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Чуркин Марат, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

В одном из писем был предложен такой “оригинальный” ответ, как “Старик предложил казакам соревноваться без лошадей”.

### Кроссворд № 1 “Елочка”

*Ответы*

По горизонтали: 2. Бит. 3. Класс. 4. Текст. 5. Сканер. 6. Анализ. 7. Логика. 8. Информация. 9. Компьютер. 10. Объект. 11. Дерево. 12. Алгоритм. 13. Обобщение. 14. Источник. 15. Пользователь.

По вертикали: абстрагирование.

*Ответы представили:*

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

— Абдувахидова Алина, Абдувахидова Софья и Милушкин Дмитрий, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Абушкин Дмитрий и Потапов Макар, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Абдурзаев Вадим и Филиппов Вадим, средняя школа села Ириновка, Новобурасский р-н Саратовской обл., учитель **Брунов А.С.**;

— Валуев Иван и Гаязов Рашид, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

— Григорьев Иван, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Дегтярь Анатолий и Новиченко Владимир, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Дощик Константин, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Зубов Владислав, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Миронова Екатерина, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**

## Кроссворд № 2

### Ответы

По горизонтали: 5. Высказывание. 7. Алгоритм. 8. Модель. 9. Ячейка. 10. Формализация.

По вертикали: 1. Управление. 2. Моделирование. 3. Конъюнкция. 4. Исполнитель. 6. Следование.

### Ответы прислали:

— Абаев Николай, средняя школа поселка Осинка, Алтайский край, учитель **Евдокимова А.И.**;

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

— Воронцов Кирилл и Приказчиков Андрей, средняя школа села Ириновка, Новобураский р-н Саратовской обл., учитель **Брунов А.С.**;

— Валуев Иван, Гаязов Рашид и Скокова Светлана, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

— Григорьев Иван, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Дегтярь Анатолий и Новиченко Владимир, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Дощик Константин, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Зубов Владислав, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

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

— Миронова Екатерина, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**

## Кроссворд № 3

### Ответы

По горизонтали. 1. Секунда. 4. Урок. 7. Высказывание. 9. Меню. 10. Таблица. 12. Риал. 17. Модель. 18. Хакер. 20. Два. 21. Ячейка. 27. Формализация. 28. Анонс. 29. Абзац. 30. Поток. 31. Математика.

По вертикали. 1. Сигнал. 2. Управление. 3. Моделирование. 5. Конъюнкция. 6. Исполнитель. 8. Набор. 11. Адресант. 13. Линейка. 14. Следование. 15. Алгоритм. 16. Мигание. 17. Матрица. 19. Единица. 20. Дефис. 22. Число. 23. Кнопка. 24. Цифра. 25. Шаг. 26. Фирма.

### Ответы представили:

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

— Байбуза Дарья, Кузнецов Семен и Михайлова Алена, средняя школа села Ириновка, Новобураский р-н Саратовской обл., учитель **Брунов А.С.**;

— Валуев Иван, Гаязов Рашид и Скокова Светлана, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

— Зеленский Константин, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

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

— Миронова Екатерина, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

— Чумак Никита, средняя школа поселка Осинка, Алтайский край, учитель **Евдокимова А.И.**;

— Шпаченко Владислав, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

## Головоломка “Найти закон формирования последовательности”

Напомним, что требовалось установить закон формирования следующей последовательности:

MN, MMN, MNN, MMMN, MNNN, ...

### Ответ

Начальный элемент последовательности — MN. В промежуток между M и N вставляется вначале M, потом N. В каждом следующем элементе количество M и N увеличивается на 1, то есть вставляется вначале M, потом N, далее MM, затем NN и так до бесконечности.

Последовательность выглядит следующим образом:

MN, MMN, MNN, MMMN, MNNN, MMMMN, MNNNN и т.д.

### Правильные ответы прислали:

— Алпатов Никита, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Гришин Василий, средняя школа поселка Осинка, Алтайский край, учитель **Евдокимова А.И.**;

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

— Жорова Мария, Ишмухаметова Яна и Мустафина Диана, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Зеленский Константин, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Коростелев Сергей, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Стороженко Степан, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Шпаченко Владислав, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

### Задача “Четырехзначные числа”

Напомним, что в задаче, предназначенной для учащихся 1–7-х классов, требовалось указать, сколько существует четырехзначных двоичных чисел.

*Ответ:* 8 чисел (1000, 1001, 1010, 1011, 1100, 1101, 1110 и 1111).

*Правильные ответы прислали:*

— Абдувахидова Алина, Абдувахидова Софья, Милушкин Дмитрий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Антонова Мария, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Ашурков Кирилл, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Волков Александр, г. Мытищи Московской обл., школа “Логос”, учитель **Елистратова А.А.**;

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

— Завадский Вадим, Мусатов Максим и Мусатов Тимофей, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Климов Андрей, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Макаров Павел и Мулюков Артур, Республика Башкортостан, г. Стерлитамак, школа № 24, учитель **Орлова Е.В.**;

— Михайлов Иван, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Остапенко Юрий, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

### Задача “Опять двуносы чайник”

Напомним, что требовалось определить, как, используя чайник с двумя носиками, полностью заполнить три чашки:

- 1) вместимостью 3, 4 и 7 условных единиц;
- 2) вместимостью 4, 5 и 18 условных единиц.

Во всех случаях принять, что в чайнике имеется достаточно большое количество воды.

*Решение*

1)

№	Действие	В 1-й чашке (3)	Во 2-й чашке (4)	В 3-й чашке (7)
Исходное состояние		0	0	0
1	Налить в 1-ю и 3-ю чашки	3	0	3
2	Налить во 2-ю и 3-ю чашки	3	4	7

2)

№	Действие	В 1-й чашке (4)	Во 2-й чашке (5)	В 3-й чашке (18)
Исходное состояние		0	0	0
1	Налить в 1-ю и 3-ю чашки	4	0	4
2	Налить во 2-ю и 3-ю чашки	4	5	9
3	Влить из 1-й чашки обратно в чайник	0	5	9
4	Влить из 2-й чашки обратно в чайник	0	0	9
5	Налить в 1-ю и 3-ю чашки	4	0	13
6	Налить во 2-ю и 3-ю чашки	4	5	18

*Ответы прислали:*

— Леоненко Степан, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Назаров Валерий, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Хвойновский Вадим, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

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

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

— Шумилов Иван, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Яковлева Надежда, Свердловская обл., Красноуфимский р-н, Тавринская средняя школа, учитель **Ярцев В.А.**;

— Яновский Игорь, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

### Новая задача о двуносом чайнике

Как, используя чайник с двумя носиками, полностью заполнить три чашки:

- 1) вместимостью 3, 4 и 17 условных единиц;
- 2) вместимостью 4, 5 и 41 условных единиц?

Во всех случаях принять, что в чайнике имеется достаточно большое количество воды.

Задания, предложенные для самостоятельной работы в статье “«Ведьма» Аньези”, выполнили:

1) с использованием электронной таблицы:

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

— Баскаков Валентин, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Батурин Илья, Бульбова Лидия, Волошин Марк, Гашимов Геннадий, Глушко Иван, Дощик Константин, Кармаев Константин, Лазуренко Глеб, Лебедева Екатерина, Мисюра Алексей, Назаркина Татьяна, Нетесов Николай, Пак Александра, Петрова Алёна, Сысоев Александр и Чунин Павел, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Зубов Владислав, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Леоненко Степан, средняя школа поселка Осинка, Алтайский край, учитель **Евдокимова А.И.**;

2) с помощью разработанной программы:

— Бульбова Лидия, Волошин Марк, Кармаев Константин, Лебедева Екатерина, Назаркина Татьяна, Петрова Алёна и Чунин Павел, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Назаров Валерий, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

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

Программы решения задач, предложенных для самостоятельной работы в статье “Лестница из чисел” (рубрика “Школа программирования”), разработали:

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

— Лобанева Анастасия, г. Смоленск, школа № 29, учитель **Родикова Р.Д.**

Редакция решила наградить Анастасию и Евгения дипломами.

Большое число заданий, предложенных для самостоятельной работы в статье “Гармонический ряд” (в том числе на разработку программ, доказательство фактов и использование электронных таблиц), выполнила Пак Александра, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.** Александра также будет награждена дипломом.

Поздравляем всех награжденных!

Отметим ответы Артура Мулюкова из школы № 24 г. Стерлитамака Республики Башкортостан, снабдившего их красочными иллюстрациями с логотипом журнала “Информатика”.

## Площадь квадрата

В учебнике, изданном на планете β-Сириус, написано: “Площадь квадрата равна квадрату его стороны. Например, если сторона равна 11, то площадь равна 1001”. Почему?

Из книги Д.М. Златопольского “Системы счисления: учебные и занимательные материалы” (М.: Ленанд, 2015)

## Постройка плотины

Три бобра построили плотину за 12 дней. К сожалению, весной ее смыло. Тогда бобры позвали

друзей и все вместе отстроили плотину за четыре дня. Сколько друзей позвали бобры? Производительность всех бобров одинаковая.

Задача предназначена для учащихся 1–7-х классов.



## Работники кафе

В одном небольшом кафе работали администратор, повар, кондитер, кассир и официант. Их фамилии: Галкина, Шалкина, Малкин, Палкин и Балкин. Известно, что:

- 1) повар — холостяк;
- 2) кассир и администратор во время учебы жили в общежитии в одной комнате;
- 3) Балкин и Шалкина встречаются только на работе;
- 4) жена Малкина очень расстроилась, когда муж сказал ей, что администратор отказал ему в отгуле на субботний вечер;
- 5) Палкин собирается стать свидетелем на свадьбе и кассира, и кондитера.

Кто на какой должности в этом кафе?

## Литература

1. Богомолова О.Б. Логические задачи. М.: БИНОМ. Лаборатория знаний, 2005.

## Есть ли такая цифра?

Есть ли такая цифра  $A$ , при которой  $432_5 < 2A1_6$ ?

## Банка с бактериями ☺

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

## Еще одна банка с бактериями

В банку поместили одну бактерию. Через минуту она поделится пополам, затем каждая из получившихся бактерий через минуту делится пополам и т.д. Через час банка стала полной. Че-

рез какое время банка будет заполнена, если в нее поместить:

- а) две бактерии;
- б) четыре бактерии?

### Сколько единиц и сколько нулей?

Найдите количество единиц  $e$  и количество нулей  $n$  в двоичной записи выражения:

$$(2^6 - 1) - (2^2 - 1).$$

### Кто на каком инструменте играет?

Два мальчика умеют играть на гитаре, а еще один — на балалайке. На чем играет Андрей, если Максим и Павел играют на разных инструментах, а Павел и Андрей — тоже?

Задача предназначена для учащихся 1–7-х классов.

### Литература

1. Богомолова О.Б. Логические задачи. М.: БИНОМ. Лаборатория знаний, 2005.

## КРЕПКИЙ ОРЕШЕК



Как всегда, в этой рубрике проводим разбор заданий, решение которых вызвало трудности.

### Головоломка “Расставить цифры”

Напомним, что требовалось расставить цифры от 1 до 6 в пустые клетки приведенной ниже таблицы в соответствии с указанными знаками так, чтобы в каждой строке и в каждом столбце все цифры были разными. Некоторые цифры уже стоят в нужных клетках (цифр 6, выделенных красным цветом, в заданной таблице не было).

Благодаря Владислава Салимова, г. Челябинск, школа № 124 (учитель **Юртаева Г.Ю.**), и Марата Хозина, Владимирская обл., г. Струнино, школа № 11 (учитель **Волков Ю.П.**), приславших ответ, приведем начало решения.

Прежде всего можно определить цифры, большие известной цифры 5 (они выделены красным цветом).

	>	2	<		>	1	<		<	
∨		∧		∨		∧		∧		∧
	<	3	>		<		>		<	
∧		∧		∧		∨		∧		∨
	<		<		>		<	6	>	1
∨		∨		∨		∧		∨		∧
	>	1	<		<		<	5	<	6
∧		∧		∧		∨		∨		∨
6	>	!	>	!	>	!	>	1	<	2
∨		∧		∨		∧		∧		∧
	<		>		<		>		<	

Затем следует проанализировать цифры на месте вопросительных знаков (их в условии головоломки, естественно, не было).

Дальнейший анализ проведите самостоятельно и пришлите ответ в редакцию.

Разбор решений трех числовых ребусов, в которых используется слово-число **АМУР**, будет проведен в следующем выпуске “В мир информатики”.

## ЯПОНСКИЙ УГОЛОК

### Два судоку

Решите, пожалуйста, две японские головоломки “судоку”:

1) простую:

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

2) сложную:

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



57 лет спустя

В 1958 году была выпущена одна из первых отечественных ЭВМ — модель “М-20” (официальное название — “машина электронная вычислительная общего назначения М-20”). Вот что написано о ней в [1]: “В середине 60-х годов прошлого столетия в нашей стране были широко распространены вычислительные машины типа М-20, построенные коллективом, возглавляемым академиком С.А. Лебедевым. По тем временам это были вполне современные компьютеры, обладающие производительностью около 20 тыс. операций в секунду. Однако их оперативная память была малой и позволяла решать системы линейных алгебраических уравнений с плотной матрицей порядка всего лишь 50–100. Требования практики заставляли искать способы решения систем значительно большего порядка. В качестве медленной памяти на этих машинах использовались магнитные барабаны. Они играли тогда такую же роль, какую сейчас в персональном компьютере играют жесткие диски. Естественно, с поправкой на объем хранимой информации. Под этот тип медленной памяти была разработана специальная блочная технология решения больших алгебраических задач. Она позволяла с использованием только 300 слов оперативной памяти решать системы практически любого порядка. Точнее, такого порядка, при котором матрица и правая часть могли целиком разместиться в медленной памяти. При этом системы решались почти столь же быстро, как будто вся информация о них на самом деле была размещена в оперативной па-

мяти. Созданные на основе данной технологии программы были весьма эффективны. В частности, на машинах типа М-20 они позволяли решать системы 200-го порядка всего за 9 минут”.

Приведем одну из программ, написанных (естественно, в машинных кодах!) для этой ЭВМ:

Адрес	Команда				
$k + 1$	0	52	0000	$n$	0000
$k + 2$	4	52	0000	0000	$k + 12$
$k + 3$	6	02	$a + 1$	$a + 2$	0000
$k + 4$	4	36	$a + 2$	$k + 7$	$c + 1$
$k + 5$	5	00	$a + 1$	0000	$a + 2$
$k + 6$	1	00	$a + 1$	0000	$a + 1$
$k + 7$	6	52	0003	0000	$c + 1$
$k + 10$	0	33	$k + 12$	$c + 1$	0000
$k + 11$	1	71	0000	$k + 3$	0001
$k + 12$	0	00	0000	0000	0000
$k + 13$	1	32	0003	$k + 2$	7777

Как вы думаете, какую задачу она решает? Ответ мы дадим, приведя фрагмент аналогичной программы на языке Паскаль:

```
n := ...;
for i := 1 to n - 1 do
  for j := 1 to n - i do
    if a[j] > a[j + 1] then
      begin
        t := a[j];
        a[j] := a[j + 1];
        a[j + 1] := t
      end;
```

Свое мнение присылайте в редакцию.

Литература

1. Воеводин В.В., Воеводин Вл. В. Параллельные вычисления. СПб.: БХВ-Петербург, 2002.

ВНИМАНИЕ! КОНКУРС!

Конкурс № 112

В качестве заданий этого конкурса предлагаем решить две задачи “Число кратчайших путей” и выполнить задания, предложенные для самостоятельной работы в статье о “полукороле”.

Ответы (можно не ко всем задачам) отправьте в редакцию до 15 октября по адресу: 121165, Москва, ул. Киевская, д. 24, “Первое сентября”, “Информатика” или по электронной почте: [vmi@1september.ru](mailto:vmi@1september.ru).

В апрельском номере “Информатики” в рубрике “Предлагаю коллегам” была опубликована статья учителя информатики гимназии № 1 г. Жуковский Московской области Ю.В. Пашковской “Играем в двоичную систему счисления”, в которой описывалась игра, которую можно сделать как элементом урока, так и основой командного соревнования, проводимого, к примеру, в рамках “Недели информатики” или во время другого внеклассного мероприятия. На фотографии справа запечатлен момент этой игры.



журнал

# Информатика – Первое сентября

1-е полугодие 2016 года

## ПОДПИСКА

на сайте [www.1september.ru](http://www.1september.ru) и в почтовых отделениях РФ

**МАП** МЕЖРЕГИОНАЛЬНОЕ АГЕНТСТВО ПОДПИСКИ

**КАТАЛОГ РОССИЙСКОЙ ПРЕССЫ**  
**ПОЧТА РОССИИ**

**2016**  
первое полугодие

Индекс	Название издания	Периодичн. в полугодие	1 месяц		6 месяцев	
			Ката- ложная цена (руб.)	Под- писная цена (руб.)	Ката- ложная цена (руб.)	Под- писная цена (руб.)
Название блока в разделе «Журналы»	<b>ПЕРВОЕ СЕНТЯБРЯ. ЖУРНАЛЫ ИЗДАТЕЛЬСКОГО ДОМА</b> (499)249-31-38					
79066	<b>Информатика – Первое сентября. Бумажная версия</b> <i>В июне не выходит. Подписка на июнь не принимается</i> (-) 160 г 64 стр.	5	440.00		2200.00	
12684	<b>Информатика – Первое сентября. Электронная версия на CD (полная копия бумажной версии)</b> <i>В июне не выходит. Подписка на июнь не принимается</i> (-) 75 г	5	160.00		800.00	
сайт <a href="http://1september.ru">1september.ru</a>	<b>Информатика – Первое сентября. Электронная версия</b>	5	–		–	500.00

Подписку принимают во всех отделениях связи Российской Федерации, а также на сайте [www.1september.ru](http://www.1september.ru)

При подключении школы к проекту «Школа цифрового века» (см. [digital.1september.ru](http://digital.1september.ru)) каждый учитель получает доступ ко всем журналам Издательского дома «Первое сентября». Стоимость подключения школы на год – 6 тыс. рублей независимо от количества учителей

При оформлении подписки на сайте оплата производится по квитанции в отделении банка или электронными платежами on-line

