

ИНФОРМАТИКА

О введении информатики и вычислительной техники в учебно-воспитательный процесс и обеспечении общей компьютерной грамотности учащихся средних учебных заведений.

Общегосударственная программа создания, развития производства и эффективного использования вычислительной техники и автоматизированных систем до 2000 года наряду с другими крупными задачами предусматривает массовое обучение в средних общеобразовательных школах, средних специальных учебных заведениях и профессионально-технических училищах работе с электронно-вычислительными машинами. В целях реализации этой программы необходимо определить конкретные меры по введению информатики и вычислительной техники в учебно-воспитательный процесс и обеспечению общей компьютерной грамотности учащихся средних учебных заведений.

Центральный Комитет КПСС и Совет Министров постановляет:

153-44
3.

Госкомитету СССР обеспечить издание необходимых тиражом экспериментального учебного пособия "Основы информатики и вычислительной техники" до I квартала 1986 г.

Министерству просвещения СССР, Министерству высшего образования СССР, Государственному комитету СССР по профессионально-техническому образованию в 1986 году провести конкурс на создание учебника "Информатика и вычислительная техника". В XI пятилетке создать полнейший комплект учебников, учебно-методических пособий для преподавателей и учащихся общеобразовательных школ, профтехучилищ и техникумов по информатике и вычислительной технике.

4. Министерству просвещения СССР, Государственному комитету среднего специального технического образования СССР, Академии наук СССР, Академии педагогических наук СССР обеспечить поставку и подготовку технических требований на разработку и производство персональных ЭВМ для учебных целей.

Госплану СССР, Министерству радиопромышленности СССР, Министерству электронной промышленности СССР обеспечить поставку в школы, средние специальные учебные заведения и средние специальные учебные заведения учебные персональные компьютеры в XII пятилетке, согласно приложению № 1, и для полного обеспечения учебных заведений и учащихся общеобразовательных школ, профтехучилищ, техникумов по информатике и вычислительной технике.

5. Министерству просвещения СССР, Государственному комитету среднего специального технического образования СССР, Министерству высшего образования СССР по профессионально-техническому образованию в эксперименте по изучению и использованию компьютеров в учебно-воспитательном процессе общеобразовательных школ, средних профессионально-технических училищ и средних специальных учебных заведений.

1. Министерству просвещения СССР, Государственному комитету среднего и среднего специального образования СССР, Академии педагогических наук СССР ввести во всех средних учебных заведениях страны, начиная с 1985/86 учебного года, общеобразовательный курс информатики и вычислительной техники в объеме 130-150 учебных часов. Определить классы в школах и курсы в профессионально-технических училищах и техникумах, в которых будут проводиться занятия по этому предмету.

Предусмотреть в учебном плане общеобразовательной школы дополнительно по одному часу в неделю X-XI классов для изучения курса информатики и вычислительной техники, установив в этих классах учебную нагрузку в пределах 32 часов.

В двухмесячный срок разработать и утвердить учебную программу курса основ информатики и вычислительной техники в двух вариантах: для средних учебных заведений, имеющих электронно-вычислительные машины или возможность организовать занятия учащихся в вычислительных центрах предприятий, организаций; а также для средних учебных заведений, не имеющих вычислительной техники.

Учебная программа для средних учебных заведений в вычислительных центрах предприятий, организаций должна предусматривать теоретические и практические занятия с учащимися. Учебная программа для средних учебных заведений, не имеющих вычислительной техники, должна предусматривать теоретические занятия, практикум с микрокалькуляторами, а также экскурсии учащихся в вычислительные центры.

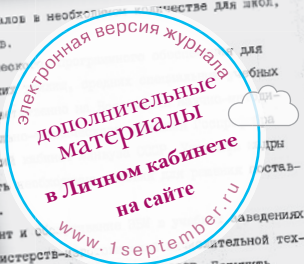
В программах расширить роль средств автоматизации и вычислительной техники в создании материально-технической базы коммунального хозяйства, а также в развитии материально-технической базы коммунального хозяйства. Особое внимание в содержании программ должно быть обращено

6. Разрешить на период с 1986 года по 1990 год привлекать к преподаванию курса "Основы информатики и вычислительной техники" в общеобразовательных школах, профтехучилищах и средних специальных учебных заведениях, а также на курсах в институтах усовершенствования учителей квалифицированных преподавателей высших учебных заведений, работников научно-исследовательских институтов и других учреждений на условиях совместительства до 0,25 ставки и почасовой оплаты.

Государственному комитету СССР по труду и социальным вопросам внести соответствующие дополнения в действующие положения о штатном совместительстве.

9. Академии наук СССР, Академии педагогических наук СССР совместно с Министерством высшего и среднего специального образования СССР, Государственным комитетом СССР по профессионально-техническому образованию, республиканскими институтами педагогики организовать комплексные исследования проблем, связанных с внедрением в учебно-воспитательный процесс электронно-вычислительной техники: содержание, методы и средств обучения, разработка программного обеспечения, интенсификация учебного процесса по различным предметам, влияние использования компьютеров на психо-физиологические и дидактические аспекты учебно-воспитательного процесса.

Преобразовать научно-исследовательский институт школьного оборудования и средств обучения АПН СССР в "Научно-исследовательский институт информатики, вычислительной техники и школьного оборудования" первой категории с дополнительной численностью сотрудников 200 чел. Поручить этому институту разработку материально-технического и программного обеспечения использования персональных компьютеров на занятиях по всем общеобразовательным предметам, а также



30 лет школьной информатике в России



НА ОБЛОЖКЕ

► 30 лет назад, в январе 1985 года, началась история школьной информатики в России как общеобразовательного предмета. Началась она, в соответствии с традициями того времени, с постановления ЦК КПСС. На обложке — черновик этого постановления из архива академика А.П. Ершова (<http://ershov.iis.nsk.su/>). Скорость, с которой все было сделано, ошеломляет даже сегодня. В кратчайшие сроки были написаны и изданы учебники, подготовлены учителя, выстроена массовая система методической и научной работы. И уже с 1985/86 учебного года началось массовое преподавание нового предмета.

В НОМЕРЕ

- 3** ПАРА СЛОВ
 - Nophone — “смартфон на любителя”
- 4** ЕГЭ
 - Функции по осени считают, или Практикум “Календарь”
 - Новые “вилы” от Министерства образования, или Как решать принципиально новые задачи
- 22** ОГЭ
 - ОГЭ на “ого!”: новые подарки-задачи для 9-го класса
- 32** УЧЕБНИКИ
 - Обработка графической информации
- 41** ВНЕКЛАССНАЯ РАБОТА
 - Шесть вопросов по информатике для конкурсов “Что? Где? Когда?” и “Брейн-ринг”
- 43** ЗАНИМАТЕЛЬНЫЕ МАТЕРИАЛЫ ДЛЯ ПЫТЛИВЫХ УЧЕНИКОВ И ИХ ТАЛАНТЛИВЫХ УЧИТЕЛЕЙ
 - “В мир информатики” № 204

В ЛИЧНОМ КАБИНЕТЕ

Облачные технологии от Издательского дома “Первое сентября”

Уважаемые подписчики бумажной версии журнала!

Дополнительные материалы к номеру и электронная версия журнала находятся в вашем Личном кабинете на сайте www.1september.ru.

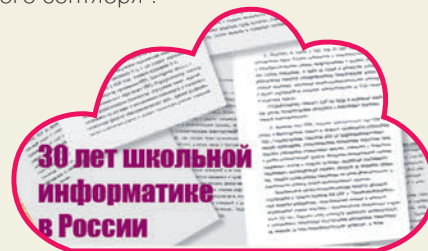
Для доступа к материалам воспользуйтесь, пожалуйста, кодом доступа, вложенным в № 1/2015.

Срок действия кода: с 1 января по 30 июня 2015 года.

Для активации кода:

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

Справки: podpiska@1september.ru или через службу поддержки на портале “Первое сентября”.



ЭЛЕКТРОННЫЕ МАТЕРИАЛЫ

- ▮ Презентации и исходные файлы к статьям номера

ИНФОРМАТИКА

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

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

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

РЕДАКЦИЯ:
гл. редактор С.Л. Островский
редакторы

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

Дизайн макета И.Е. Лукьянов
верстка Н.И. Пронская
корректор Е.Л. Володина
секретарь Н.П. Медведева
Фото: фотобанк Shutterstock
Журнал распространяется по подписке
Цена свободная
Тираж 16 000 экз.
Тел. редакции: (499) 249-48-96
E-mail: inf@1september.ru
<http://inf.1september.ru>

ИЗДАТЕЛЬСКИЙ ДОМ “ПЕРВОЕ СЕНТЯБРЯ”

Главный редактор:
Артем Соловейчик
(генеральный директор)

Коммерческая деятельность:
Константин Шмарковский
(финансовый директор)

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

Реклама, конференции и техническое обеспечение Издательского дома:
Павел Кузнецов

Производство:
Станислав Савельев

Административно-хозяйственное обеспечение:
Андрей Ушков

Педагогический университет:
Валерия Арсланьян (ректор)

ЖУРНАЛЫ ИЗДАТЕЛЬСКОГО ДОМА “ПЕРВОЕ СЕНТЯБРЯ”
Английский язык – А.Громушкина
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 12.11.2014, фактически 12.11.2014
Заказ №
Отпечатано в ОАО “Первая Образцовая типография” Филиал “Чеховский Печатный Двор”
ул. Полиграфистов, д. 1, Московская область, г. Чехов, 142300
Сайт: www.chpd.ru
E-mail: 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



Nophone — “смартфон на любителя”

► Сегодняшняя сфера бытовой электроники — это сплошная “гонка за лидером”: производители стараются выпускать все более мощные устройства, а покупатели — успеть за очередными новинками. Смартфоны здесь не исключение (в качестве наглядного примера можно взять хотя бы ажиотаж вокруг нового, шестого “айфона”). Но вот новинка под названием “Nophone”, пожалуй, заставит многих взглянуть на эту проблему с другой стороны.

В аппарате Nophone нет ни процессора, ни графического ускорителя, ни оперативной памяти, ни видеокамеры. У него нет дисплея, нет поддержки никаких беспроводных технологий и даже не предусмотрена возможность звонить и отправлять СМС. Вообще это просто кусок пластика, который своими размерами, формой и весом подобен обычному, “среднестатистическому” смартфону. Зато у данной модели есть селфи-версия, которая отличается наличием зеркала на лицевой стороне аппарата.

— Зачем же он нужен? — наверняка готов спросить читатель.

Однако сами посудите: у этого аппарата по сравнению с любыми другими — сплошные преимущества. Его не нужно заряжать,

не нужно оплачивать счет за мобильную связь, обеспечивается полная безопасность от вирусов, и даже если его вдруг украдут — особо не жалко ☺. Зато он всегда будет с вами — даже там, где по каким-либо причинам нельзя пользоваться сотовой связью, и даже там, где обычный “сотик” не будет работать (да хоть под водой!).

Конечно, данный проект — это, скорее, шутка. Но, как известно, “в каждой шутке есть... доля шутки”. Другая, уже вполне серьезная цель создателей аппарата Nophone — помочь уменьшить уровень тревожности у тех, кто подвержен мобильной зависимости и панически боится даже на секунду расстаться со своим смартфоном. (У этого психического заболевания есть даже свое название — “номофобия”.) А кстати — это и намек для тех, кто любит возносить хвалу своим “гаджетам” и постоянно хвастаться ими перед пользователями других устройств.

Правда, желающим обзавестись подобным “суперсмартфоном” придется выложить за него от 12 до 18 долларов (в зависимости от наличия зеркала). Те же, кому не хочется отдавать такие деньги за обычное карманное зеркальце, могут подождать, когда выпуск дешевых аналогов Nophone, как обычно, наладят в Китае (чего, думается, ждать придется недолго), или же воспользоваться обычной (даже уже использованной по прямому назначению) пудрицей ☺.

Д.Ю. Усенков, ст. н. с. Института информатизации РАО, Москва



Функции по осени считают, или Практикум “Календарь”

В.В. Мерзляков,
И.Н. Фалина,
Москва

► Предлагаем нашим коллегам — учителям дидактический материал для учащихся, которые уже изучали работу с процедурами и функциями. Называется задание — практикум “Календарь”.

Условие задания:

По введенному номеру года вывести в красивом и понятном виде календарь на указанный год. Для этого необходимо последовательно решить перечисленные ниже подзадачи. Решение каждой задачи (до седьмой включительно) нужно оформлять в виде функции. Все параметры — целые числа. Основная

программа отвечает только за ввод/вывод.

Подсказка: Программа “Календарь” должна опираться на дату выдачи задания (сегодня такое-то число, такой-то день недели).

Перечень функций и процедур, которые вы должны реализовать в своей программе.

1) По номеру года определить, високосный он или нет.

2) По номеру года определить количество дней в году (365 или 366).

3) По номеру года и номеру месяца определить количество дней в месяце.

4) Дана дата (день, месяц и год). Определить, сколько дней прошло до нее, начиная с 1 января данного года.

5) Дана дата (день, месяц и год). Определить, сколько дней прошло до нее, начиная с 1 января 1-го года.

6) Даны две даты (день, месяц и год для каждой). Определить, сколько дней прошло между ними.

7) Даны две даты (день, месяц и год для каждой) и день недели первой даты (в виде целого числа от 1 до 7

или от 0 до 6). Требуется определить день недели второй даты. Вторая дата может быть как позже, так и раньше первой.

8) Вводится месяц и год. Выдать в красивом виде календарь на этот месяц.

9) Вводится год. Выдать в красивом виде календарь на год.

Цели, с которыми мы даем задание-практикум “Календарь”

Задание можно давать с двумя основными учебными целями. Первая — практикум при первичном изучении процедур и функций. В этом случае задание дается тогда, когда впервые вводятся процедуры и функции (например, в 10-м классе).

Вторая — повторение и закрепление материала. В этом случае задание можно давать как первый практикум после летних каникул, что регулярно проводится в нашей школе. Естественно, что за лето школьники могут многое (или почти все) забыть. А здесь им предлагается хорошо структурированное и формализованное задание, которое позволит достаточно быстро вспомнить, как разрабатываются функции/процедуры в процедурных языках типа Паскаля.

Также при выполнении задания достигаются следующие методические цели:

1. *Декомпозиция.* На примере этого задания школьники видят, как на первый взгляд сложное задание в результате правильно выполненной декомпозиции (разбиения большой задачи на маленькие подзадачи, каждую из которых легко решать) становится “прозрачным” и реально выполнимым.

2. *Очевидная отладка.* Так как задание позволяет постепенно “наращивать” количество работающих функций, то школьникам легко проверять правильность написанных функций. Каждая следующая функция использует ранее описанные и отлаженные функции.

3. *Групповая работа.* Описанная выше особенность задания позволяет использовать групповую форму работы при выполнении проекта. Мы делим класс на группы по три человека. Каждой группе выдается одно и то же задание. Внутри группы школьники должны согласовать между собой, кто какую функцию пишет. Именно в процессе этого обсуждения школьники начинают строить каркас будущей программы.

4. *Глобальные и локальные переменные.* При написании процедур/функций школьники любят все данные хранить в глобальных переменных. В этом задании в явном виде указаны параметры функций. Правда, некоторых учеников это не смущает, и они по-прежнему стараются все “впихнуть” в глобальные переменные. При групповой форме работы школьники очень быстро наталкиваются на неудобства (и, вообще говоря, недопустимость) использования глобальных переменных, возникающие при слиянии частей программы, написанных разными людьми, в одно целое.

5. *Целочисленная арифметика.* При разработке алгоритмов функций для этого задания школьники должны осознанно использовать целочисленные операции `div` и `mod`. Правильно написанные функции формируют понимание, когда и как использовать операции `div` и `mod`. Эти навыки и умения необходимы в том числе для успешного решения задач ЕГЭ по информатике.

6. *Тонкости программирования.* Так как у всех функций (если они написаны правильно и красиво) достаточно маленький и простой для понимания код, то это позволяет учителю при проверке обратить внимание на “тонкости” используемых алгоритмов. Например:

6.1. для определения високосности года достаточно написать одно сложное логическое выражение, а не использовать три условных оператора с простыми проверками;

6.2. это задание заставляет вспомнить и применить оператор выбора `CASE`;

6.3. при выводе календаря естественно напрашивается использование модификатора ширины поля в операторах `write` и `writeln`;

6.4. задание показывает важность использования мнемоничных имен; использование “говорящих” имен повышает читаемость, наглядность, а следовательно, и надежность программы;

6.5. при работе над заданием школьники осознают важность очередности описания функций. Например, функция “количество дней в году” вызывает функцию “определения високосности года”, следовательно, в блоке описания функций сначала надо описать функцию `vis`, а затем — `days_in_year`. Имена функций могут быть другими, но обязательно должны нести смысловую нагрузку.

7. *Интерфейс программы.* Для красивого вывода необходимо использовать разные возможности языка. Под красивым видом понимается следующее (зависит от уровня школьников):

7.1. Календарь можно выводить в один столбик по месяцам

```
<название месяца>
<строка дней недели>
<строка чисел месяца первой недели>
...
```

Обязательное требование: первое число месяца стоит на месте, соответствующем дню недели первого числа.

7.2. Календарь можно выводить в более привычном и удобном для чтения виде

Январь <строка дней недели> Дни месяца по неделям	Февраль <строка дней недели> Дни месяца по неделям	Март <строка дней недели> Дни месяца по неделям
---	--	---

Для реализации второй формы вывода календаря можно использовать массив или процедуру `GOTOXY(x, y)`.

7.3. Если к моменту выдачи задания учащиеся умеют работать с двумерными массивами, то можно предложить им “развернуть” календарь, чтобы дни недели записывались в столбик. В этом случае календарь будет иметь, например, следующий вид:

Январь				Февраль				Март							
Пн	7	14	21	28	Пн	4	11	18	25	Пн	4	11	18	25	
Вт	1	8	15	22	29	Вт	5	12	19	26	Вт	5	12	19	26
Ср	2	9	16	23	30	Ср	6	13	20	27	Ср	6	13	20	27
Чт	3	10	17	24	31	Чт	7	14	21	28	Чт	7	14	21	28
Пт	4	11	18	25	Пт	1	8	15	22	Пт	1	8	15	22	29
Сб	5	12	19	26	Сб	2	9	16	23	Сб	2	9	16	23	30
Вс	6	13	20	27	Вс	3	10	17	24	Вс	3	10	17	24	31

7.4. Календарь можно выводить в четыре столбика по кварталам.

Ниже приведен текст программы (до восьмого этапа включительно), который более-менее можно считать “эталонным”. В этом тексте учтены многие тонкости, на которые, на наш взгляд, надо обращать внимание школьников и о которых говорилось выше.

```
function vis (y:integer):boolean;
//Проверка года на високосность
begin
  if (y mod 400 = 0) or (y mod 4 = 0)
    and (y mod 100 <> 0)
  then vis := true
  else vis := false
end;

function days_in_year(y:integer):integer;
//Число дней в году
begin
  if vis(y) then days_in_year := 366
  else days_in_year := 365
end;

function days_in_month(m,y:integer):integer;
//Число дней в месяце
begin
  case m of
    1,3,5,7,8,10,12: days_in_month := 31;
    4,6,9,11: days_in_month := 30;
    2: if vis(y) then days_in_month := 29
       else days_in_month := 28;
  end;
end;

function
days_from_1_january(d,m,y:integer):
integer;
//Количество дней с первого января
// текущего года
var i,s:integer;
begin
  s := d;
  for i := m - 1 downto 1 do
    s := s + days_in_month(i, y);
    days_from_1_january := s
  end;

function
days_from_begin_of_time(d,m,y:integer):
integer;
```

```
//Количество дней с первого января
//первого года
var i,s:integer;
begin
  s := days_from_1_january(d,m,y);
  for i := y - 1 downto 1 do
    s := s + days_in_year(i);
    days_from_begin_of_time := s
  end;

function days_between(d1,m1,y1,d2,m2,y2:
integer):integer;
//Количество дней между двумя датами
begin
  days_between :=
days_from_begin_of_time(d2,m2,y2)-
days_from_begin_of_time(d1,m1,y1)
end;

function
day_of_week(d1,m1,y1,k,d2,m2,y2:integer):
integer;
//Определяем день недели
var s,c:integer;
begin
  c := days_between(d1,m1,y1,d2,m2,y2);
  if c > 0 then s := (k + c mod 7) mod 7
  else s := (k + c mod 7 + 7) mod 7;
  day_of_week := s
end;

procedure calendar(m,y:integer);
var i,k,first,last:integer;
begin
  writeln;
  writeln('пн':4,'вт':4,'ср':4,'чт':4,
         'пт':4,'сб':4,'вс':4);
  first := day_of_week(19,10,2014,6,1,m,y);
  last := days_in_month(m,y);
  k := 1;
  for i := 0 to first - 1 do
    //Вывод нужного количества пробелов
    begin
      write(' ':4);
      inc(k);
    end;
  for i := 1 to last do //основной вывод
    begin
      if k = 8 then begin k := 1; writeln end;
      write(i:4);
      inc(k)
    end
  end;
end;

var m,y:integer;
begin
  write('Введите месяц и год: ');
  readln(m,y);
  calendar(m,y)
end.
```

Обращаем ваше внимание, что в основной программе переменные *m* и *y* нарочно описаны после всех функций, чтобы ими могла пользоваться только основная программа. Данная программа написана на PascalABC (существенно, что тип *integer* занимает 4 байта).



ДИСТАНЦИОННЫЕ КУРСЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ

(с учетом требований ФГОС)

До 15 января производится прием заявок на второй поток 2014/15 учебного года

образовательные программы:

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

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

По окончании выдается удостоверение о повышении квалификации установленного образца

Перечень курсов и подробности – на сайте edu.1september.ru

Пожалуйста, обратите внимание:

заявки на обучение подаются только из Личного кабинета, который можно открыть на любом сайте портала www.1september.ru



Новые “вилы” от Министерства образования, или Как решать принципиально новые задачи

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

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

А.Ремизова,
ученица 11-го “А” класса
школы № 1360
г. Москвы

► Наша школа недавно приняла на себя роль “экспериментальной площадки” для апробации новых заданий ЕГЭ по математике и информатике. И в конце октября 2014 года нашим школьникам было предложено написать пробную диагностическую работу в формате ЕГЭ по информатике, в которой многие задания были не просто усложненными, но и принципиально новыми по формулировкам.

Насколько оправданно еще более усложнять задания ЕГЭ при том, что результативность сдачи экзамена и на прежних заданиях в масштабе страны оказалась, мягко говоря, невысокой, — вопрос спорный. Может быть, для того, чтобы

закрыть дорогу в профильные вузы еще большему количеству одиннадцатиклассников? Однако наша задача теперь — предполагая, что такие задания могут встретиться на реальном ЕГЭ, научить ребят решать и их.

Ниже приводятся решения задач одного из вариантов пробной диагностической работы, благо на листах с этими заданиями нигде не указан запрет на их публикацию.

Итак...

1. Укажите наибольшее число, двоичная запись которого содержит ровно три значащих нуля и две единицы. Ответ запишите в десятичной системе счисления.

Решение

1) Значащие нули не могут располагаться в самом начале числа, — значит, число должно начинаться хотя бы с одной единицы.

2) Вспомним формулу перевода двоичного числа в десятичное: чем левее в двоичном числе единица, тем большая степень двойки входит в качестве слагаемого в эту запись. Следовательно, максимальное значение двоичного числа получится, если все единицы будут “собраны” в старших разрядах.

3) Требуется, чтобы в записи числа было ровно три значащих нуля и две единицы. Следовательно, число — 5-разрядное. Если все единицы размещаются в старших разрядах, то это число 11000_2 .

4) Ответ требуется записать в десятичной форме. Переведем полученное двоичное число в десятичное:

$$11000_2 = 2^4 + 2^3 = 16 + 8 = 24.$$

Ответ: 24.

2. Между населенными пунктами А, В, С, D, E, F, Z построены дороги с односторонним движением. В таблице указана протяженность каждой дороги. Отсутствие числа в таблице означает, что прямой дороги между пунктами нет. Например, из А в В есть дорога длиной 4 км, а из В в А дороги нет.

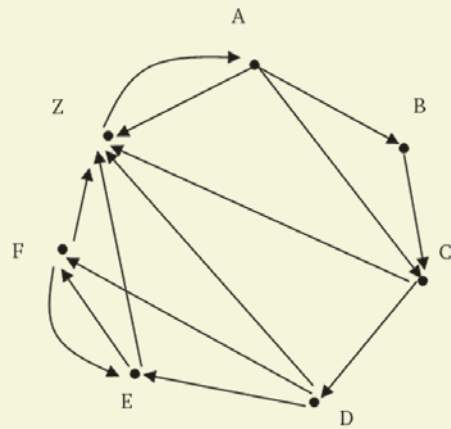
	А	В	С	D	E	F	Z
А		4	6				30
В			3				
С				11			27
D					4	7	10
E						4	8
F					5		2
Z	29						

Сколько существует таких маршрутов из А в Z, которые проходят через 6 и более населенных пунктов? Пункты А и Z при подсчете учитывать. Два раза проходить через один пункт нельзя.

Решение

Речь идет о построении графа по заданной табличной форме (как ранее в аналогичных задачах А2), но графа *ориентированного*, в котором ребра обладают направлениями. Их нужно обозначать не отрезками, а стрелками. В случае же, если в таблице заданы и прямой, и обратный пути (как между пунктами А и Z), надо их изобразить двумя отдельными соответствующими стрелками. (Если бы в задаче требовалось определить длину пути, каждая стрелка должна была бы изображаться со своим *весом* — обозначением длины пути.) Точки (вершины графа) располагаем произвольно, не учитывая реальное их расположение, т.е. граф отражает лишь условную “карту дорог”.

1) Строим требуемый граф:



2) Начинаем подсчитывать пути. При этом помним, что через один и тот же пункт в пределах одного какого-то пути нельзя проходить дважды. (Можно при решении задачи на бумаге прорисовывать каждый путь своим цветом при помощи фломастера.) Например:

— AZ: путь напрямую из пункта А в пункт Z существует, но проходит только через два пункта, а не через шесть и более, — значит, он нам не подходит. (Обратный путь из Z в А вообще не учитываем, так как по условию задачи ищутся только пути из А в Z.)

— ABCDEFZ: путь через все вершины графа — он подходит, так как проходит через семь вершин и ни одна из них не посещается дважды.

Аналогичным способом просматриваем остальные пути и выбираем среди них нужные.

Сложно? Есть риск что-то пропустить? Вспомним тогда прежнюю задачу типа В9, где по ориентированному графу требовалось определить количество всех путей из одного города в другой. Ведь, по сути, это та же самая задача!

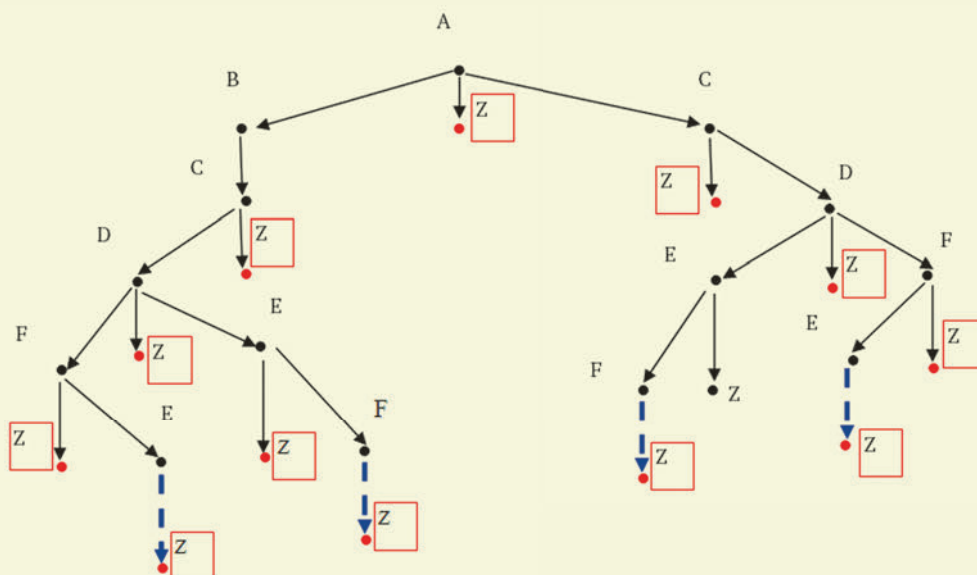
3) Рисуем дерево всех возможных путей из А в Z, при этом конечные точки маршрута (Z) выделяем цветом и рамкой. При этом важно не забыть о запрете на повторное посещение вершин. Это означает, что обратный путь FE нельзя рисовать в тех ветвях дерева, где уже ранее фигурировали буквы E и F (такие ветви в графе, где из E допускается только один путь в Z, отметим синими пунктирными стрелками) — см. рис. на с. 10.

4) Остается подсчитать количество ветвей в этом графе, включающих в себя шесть и более вершин (т.е. когда запись последовательности имен проходимых вершин состоит из шести и более букв). Это ветви (пути): ABCDFZ, ABCDFEZ, ABCDEFZ, ABCDEFZ, ACDFEZ, ACDFEZ. Всего — шесть путей.

Ответ: 6.

3. Каждое из логических выражений F и G содержит шесть переменных. В таблицах истинности выражений F и G есть ровно десять одинаковых строк, причем ровно в пяти из них в столбце значений стоит 1.

Сколько строк таблицы истинности для выражения $F \vee G$ содержат единицу в столбце значений?



Решение¹

1) В каждом из логических выражений имеется шесть переменных, следовательно, для каждого выражения возможно $2^6 = 64$ различных набора значений переменных.

2) По условию, в таблицах истинности выражений есть десять одинаковых строк (когда обе функции имеют значение 1 или значение 0). Значит, строк, в которых значения функций не совпадают, будет $64 - 10 = 54$.

3) Выражение $F \vee G$ равно 1, если хотя бы одно из значений (F или G) равно единице, и равно нулю тогда и только тогда, когда оба значения (и F, и G) равны нулю. По условию, среди одинаковых строк в таблицах истинности F и G есть ровно пять строк, в которых значения равны 1. Значит, таких строк, где и для F, и для G значения равны нулю, будет $10 - 5 = 5$ штук.

4) Итак, имеем:

- 54 строки, в которых значения F и G различны (одно равно 0, другое равно 1), т.е. значение выражения $F \vee G$ равно 1;
- 5 строк, в которых значения F и G одинаковы и равны 1, т.е. значение выражения $F \vee G$ также равно 1;
- 5 строк, в которых значения F и G одинаковы и равны 0, т.е. значение выражения $F \vee G$ равно 0.

Условиям задачи соответствует, таким образом, $54 + 5 = 59$ строк.

Комментарий

Вопрос мог быть поставлен и так: сколько строк содержат в столбце значений 1 для выражения $F \wedge G$? При решении такой задачи ход рассуждений будет аналогичным, но на четвертом шаге надо учесть, что для логической операции И результат равен единице тогда и только тогда, когда *оба* значения (и F, и G) равны 1. Таких строк, очевидно, только пять.

¹ Предложено Анастасией Ремизовой (11-й «А» класс, школа № 1360).

Решение задач на определение количества строк таблицы истинности выражения $F \vee G$, содержащих нули, или количества строк таблицы истинности выражения $F \wedge G$, содержащих нули, оставим читателям для самостоятельного решения.

4. Для групповых операций с файлами используются маски имен файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться следующие символы:

- символ “?” (вопросительный знак) означает ровно один произвольный символ;
- символ “*” (звездочка) означает любую последовательность символов произвольной длины, в том числе “*” может задавать и пустую последовательность.

В каталоге находится шесть файлов:

- maverick.map
- maverick.mp3
- taverna.mp4
- revolver.mp4
- vera.mp3
- zveri.mp3

Ниже представлено восемь масок. Сколько из них таких, которым соответствуют ровно три файла из данного каталога?

ver.mp*	*?ver?.mp?	?*ver*.mp?*	*v*r?.m?p*
?????.mp*	???*???.m*	*a*.a*	*a*.p*

Решение

Это — несколько усложненный вариант прежней задачи типа А4. Принцип решения — тот же: “примерка” каждой маски к именам файлов и определение количества файлов, соответствующих маске. Маска, для которой будет отобрано ровно три файла (или маски), — и есть ответ.

1) Маска ***ver*.mp*** предполагает, что где-то в имени файла содержится набор символов **ver**, а расширение начинается с символов **mp**. Значит, такая маска отберет файлы **maverick.mp3**, **taverna.mp4**,

revolver.mp4, vera.mp3, zveri.mp3 — всего пять файлов. Значит, эта маска нам не годится.

2) Маска ***?ver?.mp?** отличается от предыдущей тем, что до и после символов **ver** обязательно должны стоять минимум по одному символу, а в расширении после символов **mp** может быть только один символ. Значит, такая маска отберет файлы **maverick.mp3, taverna.mp4** и **zveri.mp3** (а имена **revolver.mp4** и **vera.mp3** не удовлетворяют этой маске, так как в них группа символов **ver** располагается с краю имен). Всего — три файла. Значит, эта маска *подходит*.

3) Маска ***?ver*.mp?*** — минимум один символ должен быть перед **ver**, а в расширении после **mp** должно быть не менее одного символа. Такая маска отберет файлы **maverick.mp3, taverna.mp4, revolver.mp4** и **zveri.mp3** — четыре файла. Данная маска не годится.

4) Маска ***v*r?.m?p*** — не обращая внимания на маску имени, заметим: в расширении между **m** и **p** должен стоять хотя бы один какой-то символ. А у нас этому условию соответствует единственный файл **maverick.map**. Значит, эта маска тоже не годится.

5) Маска **???*???.mp*** — ей удовлетворяют файлы с любыми именами не менее чем из шести букв и с расширениями, начинающимися с **mp**. Значит, будут отобраны файлы **maverick.mp3, taverna.mp4** и **revolver.mp4** (у файлов **vera.mp3** и **zveri.mp3** имена слишком короткие, а расширение файла **maverick.map** не соответствует расширению в маске). Значит, данная маска тоже *подходит*.

6) Маска **???*???.m*** — отличается от предыдущей тем, что здесь расширение может быть любым, лишь бы оно начиналось с **m**. Тогда в дополнение к файлам **maverick.mp3, taverna.mp4** и **revolver.mp4** в подборку попадет и файл **maverick.map**. Всего — четыре файла. Значит, эта маска не годится.

7) Маска ***a*.a*** — соответствует любым файлам, у которых и в имени, и в расширении есть хотя бы одна буква **a**. Этому условию соответствует только один файл — **maverick.map**. Значит, эта маска тоже не годится.

8) Маска ***a*.p*** — здесь в имени требуется хотя бы одна буква **a**, а в расширении — хотя бы одна буква **p**. Этому условию соответствуют файлы: **maverick.map, maverick.mp3, taverna.mp4** и **vera.mp3**. В этой подборке четыре файла, значит, данная маска тоже не годится.

Итого нам подошло две маски — ***?ver?.mp?** и **???*???.mp***.

Ответ: 2.

5. Автомат получает на вход четырехзначное число. По этому числу строится новое число по следующим правилам.

1. Складываются первая и вторая, а также третья и четвертая цифры исходного числа.

2. Полученные два числа записываются друг за другом в порядке возрастания (без разделителей).

Пример. Исходное число: 6531. Суммы: $6+5 = 11$, $3+1 = 4$. Результат: 411.

Укажите наибольшее число, в результате обработки которого автомат выдаст число 1113.

Решение

Подобные задачи в предыдущих вариантах ЕГЭ уже были. Но раньше нужно было выбрать подходящие числа из приведенных вариантов ответа, а теперь, наоборот, искать такое число.

1) Числа (по умолчанию) десятичные. Значит, суммы цифр могут быть в диапазоне от 1 ($1+0$) до 18 ($9+9$).

2) Автомат выдал число 1113. Значит, оно состоит из двух значений сумм, записанных по возрастанию: 11 и 13.

3) Число 11 может быть суммой цифр: $2+9$, $3+8$, $4+7$ или $5+6$. Число 13 может быть суммой цифр: $4+9$, $5+8$, $6+7$. Значит, пары цифр в исходном числе могут быть такими: одна пара — 29, 92, 38, 83, 47, 74, 56, 65 и другая пара — 49, 94, 58, 85, 67, 76 (ведь соответствующие цифры могут меняться местами в соответствующих суммах).

4) Наибольшим является число, в которое входят наибольшие цифры, причем они по возможности располагаются слева направо по убыванию (поскольку чем старше разряд, тем больше “вес” цифры). Тогда из доступных нам “для конструирования” наборов пар цифр выберем две наибольшие (по одной из каждого набора): 92 и 94 и запишем их по убыванию: 9492.

Ответ: 9492.

6. В фрагменте базы данных представлены сведения о родственных отношениях. На основании приведенных данных определите, сколько прямых потомков (то есть детей и внуков) Кривич Л.П. упомянуто в таблице.

Таблица 1			Таблица 2	
ID	Фамилия И.О.	Пол	ID родителя	ID ребенка
2146	Кривич Л.П.	Ж	2146	2302
2155	Павленко А.К.	М	2146	3002
2431	Хитрук П.А.	М	2155	2302
2480	Кривич А.А.	М	2155	3002
2302	Павленко Е.А.	Ж	2302	2431
2500	Сокол Н.А.	Ж	2302	2511
3002	Павленко И.А.	М	2302	3193
2523	Павленко Т.Х.	Ж	3002	2586
2529	Хитрук А.П.	М	3002	2570
2570	Павленко П.И.	М	2523	2586
2586	Павленко Т.И.	Ж	2523	2570
2933	Симонян А.А.	Ж	2529	2431
2511	Сокол В.А.	Ж	2529	2511
3193	Биба С.А.	Ж	2529	3193
...

Решение

Данная задача решается так же, как привычные уже задачи типа А6, но здесь нужно, начиная от родителей, определять по базе данных сначала детей, по-

том внуков и т.д., подсчитывая их количество. (Аналогичным способом можно подсчитывать и количества других родственников.) Реально, как видно далее из решения, эта задача, наоборот, несколько упрощена по сравнению с прежними, — особенно теми, где нужно было искать племянников и племянниц.

1) В табл. 1 ищем Кривич Л.П. и определяем ее ID: 2146.

2) В правой таблице ищем все пары ID, в которых первое число (в левой колонке) равно 2146 (строки выделены зеленым цветом). Для них в правой колонке записаны ID 2302 и 3002. Это — дети Кривич Л.П.

3) В правой таблице ищем все пары ID, в которых первое число (в левой колонке) равно 2302 или 3002 (строки выделены, соответственно, голубым и желтым цветами). Это — внуки Кривич Л.П.

4) Подсчитываем количество закрашенных в табл. 2 строк: 7 (двое детей и пятеро внуков).

Ответ: 7.

7. Производилась двухканальная (стерео) звукозапись с частотой дискретизации 64 кГц и 24-битным разрешением. В результате был получен файл размером 480 Мбайт, сжатие данных не производилось. Определите приблизительно, сколько времени (в минутах) проводилась запись. В качестве ответа укажите ближайшее к времени записи целое число, кратное 10.

Решение

Это — вполне привычная задача типа А8. Только раньше мы, зная длительность записи, оценивали получаемый размер файла. А теперь требуется, наоборот, по заданному размеру файла оценить длительность. А значит, нужно будет составить уравнение.

1) Разрешение равно 24 битам.

2) Частота дискретизации равна 64 кГц — 64 000 измерений в секунду.

3) Длительность записи (в секундах) примем равной x .

4) Запись двухканальная — объем информации умножается на 2.

5) Результирующий объем файла равен 480 Мб, или $480 \cdot 2^{23}$ бит.

(Не забываем, что все расчеты ведутся в битах и в секундах.)

6) Составляем уравнение:

$$24 \times 64\,000 \times x \times 2 = 480 \cdot 2^{23}.$$

Для удобства вычислений по возможности выделяем степени двойки:

$$3 \cdot 2^3 \times 125 \cdot 2^9 \times x \times 2 = 15 \cdot 2^5 \cdot 2^{23};$$

$$3 \cdot 5^3 \times x \times 2^{13} = 3 \cdot 5 \cdot 2^{28};$$

$$25 \times x = 2^{15};$$

$$x = 2^{15}/25 = 1310,72 \text{ (секунды)}.$$

7) Переводим x в минуты:

$$x \approx 1311 / 60 \approx 21,85 \text{ (минуты)}.$$

8) Округляем до ближайшего значения, кратного 10: $x \approx 21,85 \approx 20$ (минут).

Ответ: 20.

8. По каналу связи передаются сообщения, содержащие только четыре буквы — П, О, С, Т. Для

передачи используется двоичный код, допускающий однозначное декодирование. Для букв Т, О, П используются такие кодовые слова: Т — 101, О — 0, П — 100.

Укажите кратчайшее кодовое слово для буквы С, при котором код будет допускать однозначное декодирование. Если таких кодов несколько, укажите код с наименьшим числовым значением.

Решение

Задача на кодирование и применение условия Фано. Аналогичные задания были и в прежних вариантах ЕГЭ, но там требовалось лишь выбрать один из готовых вариантов ответа, а теперь нужно самим найти такой код.

Вспомним условие Фано:

Неравномерный код может быть однозначно декодирован, если никакой из кодов не совпадает с началом (префиксом) какого-либо другого, более длинного кода.

Таким образом, требуется найти кратчайший код, не совпадающий ни с одним имеющимся кодом, ни с началом одного из имеющихся кодов (Т — 101, О — 0, П — 100). Поиск решения ведем простым перебором от однобитовых кодов к кодам большей разрядности:

- код 0 — не годится (совпадает с кодом буквы О);

- код 1 — не годится (совпадает с началом кодов 100 и 101);

- код 10 — не годится (совпадает с началом кодов 100 и 101);

- код 11 — не совпадает с началом никакого из имеющихся кодов и ни с одним из имеющихся кодов. Следовательно, это и есть ответ.

Ответ: 11.

9. На числовой прямой даны два отрезка:

$$P = [43, 49] \text{ и } Q = [44, 53].$$

Укажите наибольшую возможную длину отрезка А, для которого формула

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

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

Решение

Уже привычная задача “на отрезки”, два способа решения которой описаны в статье авторов — “ЕГЭ: возможные особенности заданий 2014 года” (“Информатика”, 2014, № 5. С. 4–21). Впрочем, есть и еще один, более простой способ, который мы рассмотрим ниже.

1) Как обычно, заменяем высказывания логическими переменными: $a = (x \in A)$, $p = (x \in P)$, $q = (x \in Q)$. Тогда исходное логическое выражение принимает вид:

$$(a \rightarrow p) \vee q.$$

2) Избавимся от операции следования, используя закон де Моргана ($x \rightarrow y = \neg x \vee y$):

$$(a \rightarrow p) \vee q = \neg a \vee p \vee q.$$

3) Требуется тождественное равенство этого выражения единице на всей числовой оси. При этом помним, что значения логических переменных рав-

ны 1 в пределах соответствующих отрезков и 0 — вне их (причем границы входят в отрезки).

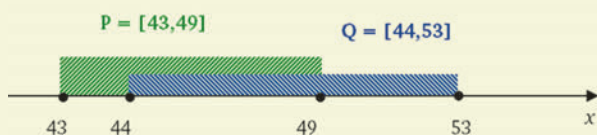
Анализируем полученное уравнение

$$\neg a \vee p \vee q = 1:$$

- переменные соединены операцией ИЛИ, следовательно, результат 1 будет автоматически обеспечен, если хотя бы одна из переменных — p или q — равна 1 (то есть везде, где числовая прямая “накрыта” хотя бы одним отрезком — P или Q);

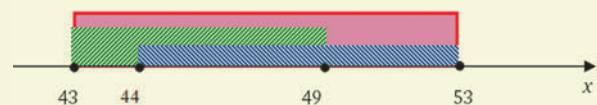
- там, где нет ни отрезка P , ни отрезка Q (т.е. переменные p и q обе равны 0), для обеспечения истинности результата должна быть равна 1 переменная $\neg a$, или, соответственно, переменная a должна быть равна 0 (отрезка A не должно быть нигде, где нет ни отрезка P , ни отрезка Q).

4) Рисуем числовую прямую и изображаем на ней отрезки P и Q :



5) Теперь совершенно очевидно, как расположить на этой числовой прямой отрезок A , чтобы он нигде “не высовывался” за пределы области, закрытой хотя бы одним отрезком P или Q :

$$A = [43, 53]$$



6) А теперь — внимание! — нужно правильно вычислить длину полученного отрезка A , помня, что речь идет о прямой натуральных чисел. В данном случае это не имеет особого значения, но в других аналогичных задачах при выполнении операции отрицания границы отрезка A могут быть “выколоты” (могут не входить в отрезок), и тогда нужно при вычислении длины отрезка брать соседние натуральные граничные значения внутри отрезка. В нашем же случае конечные точки 43 и 53 входят в отрезок A .

Теперь вспомним, как мы вычисляем длину отрезка на обычной сантиметровой линейке, — например, между отметками 1 см и 3 см. Длина этого отрезка равна двум сантиметровым *интервалам* между указанными отметками на шкале линейки, т.е. вычисляется как разность значений. Точно так же нужно поступить и здесь, поэтому длина отрезка A равна $53 - 43 = 10$.

Ответ: 10.

10. При регистрации в компьютерной системе каждому пользователю выдается пароль, состоящий из восьми символов и содержащий только заглавные символы нижнего ряда латинской раскладки клавиатуры: Z, X, C, V, B, N, M. Каждый такой пароль в компьютерной программе записывается минимально возможным и одинаковым целым количеством байт (при этом используют посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством бит).

Определите объем памяти (в байтах), отводимый этой программой для записи 25 паролей. В ответе запишите только число.

Решение

Это — все та же задача типа A11.

1) В алфавит входит семь символов (Z, X, C, V, B, N, M). При такой мощности алфавита для кодирования одного символа требуется три бита ($2^3 = 8$, а $2^2 = 4$ — недостаточно).

2) Пароль состоит из восьми символов, значит, для их хранения потребуется $8 \times 3 = 24$ бита.

3) Тогда для хранения одного пароля требуется $24/8 = 3$ байта. (Если бы результат деления был не целым, потребовалось бы взять ближайшее большее целое количество байт.)

4) Для хранения 25 паролей тогда потребуется $25 \times 3 = 75$ байт.

Ответ: 75.

11. В программе описан одномерный целочисленный массив с индексами от 0 до 10. Ниже представлен фрагмент программы, обрабатывающей данный массив²:

```
s := 0;
n := 10;
for i := 0 to n - 1 do begin
    s := s + A[i] - A[i + 1];
end;
```

В начале выполнения этого фрагмента в массиве находились трехзначные натуральные числа, ни одно из которых не делится на 20. Какое наибольшее значение может иметь переменная s после выполнения данной программы?

Решение

Задача похожа на привычное задание типа A12, за одним исключением: исходное содержимое массива неизвестно.

1) Анализируем алгоритм:

- длина массива равна 10; переменная s обнулена для подсчета суммы/разности;
- в цикле выполняется сложение прежнего значения s с разностью текущего элемента массива и последующего.

Такое вычисление значения s можно распisać в одну строку в виде:

$$s = (A[0] - A[1]) + (A[1] - A[2]) + (A[2] - A[3]) + (A[3] - A[4]) + (A[4] - A[5]) + (A[5] - A[6]) + (A[6] - A[7]) + (A[7] - A[8]) + (A[8] - A[9]) + (A[9] - A[10]).$$

$$s = A[0] + (A[1] - A[1]) + (A[2] - A[2]) + (A[3] - A[3]) + (A[4] - A[4]) + (A[5] - A[5]) + (A[6] - A[6]) + (A[7] - A[7]) + (A[8] - A[8]) - A[9] + (A[9] - A[9] + A[10]).$$

² В варианте диагностической работы, как и в вариантах ЕГЭ, в подобных случаях даются несколько листингов одной и той же программы, записанных на различных языках программирования: Бейсик, Паскаль, Си и “алгоритмический язык”. Мы же ради экономии места приводим только листинг на Паскале. — **Прим. авт.**

Как видим, большинство слагаемых сокращается, и получаем:

$$s = A[0] - A[10].$$

2) При этом требуется, чтобы итоговое значение s было максимальным. Когда это возможно, учитывая, что все числа — трехзначные и не кратные 20? Очевидно, уменьшаемое должно быть максимально возможным таким числом, а вычитаемое — минимально возможным. Всем этим условиям отвечают значения: $A[0] = 999$, $A[9] = 101$ (минимально возможное трехзначное число 100 непригодно, так как оно кратно 20). Тогда значение s (максимально возможное, требуемое в задаче) равно $999 - 101 = 898$.

Ответ: 898.

12. Система команд исполнителя РОБОТ, “живущего” в прямоугольном лабиринте на клетчатой плоскости, состоит из восьми команд. Четыре команды — это команды-приказы: **вверх**, **вниз**, **влево**, **вправо**. При выполнении любой из этих команд РОБОТ перемещается на одну клетку, соответственно, вверх, вниз, влево или вправо. Четыре других команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ: **сверху свободно**, **снизу свободно**, **слева свободно**, **справа свободно**.

Цикл

ПОКА условие

последовательность команд

КОНЕЦ ПОКА

выполняется, пока условие истинно.

В конструкции

ЕСЛИ условие

ТО команда1

ИНАЧЕ команда2

КОНЕЦ ЕСЛИ

выполняется команда1 (если условие истинно) или команда2 (если условие ложно).

В конструкциях ПОКА и ЕСЛИ условие может содержать команды проверки, а также слова И, ИЛИ, НЕ.

Если РОБОТ начнет движение в сторону находящейся рядом с ним стены, то он разрушится и программа прервется.

Сколько клеток лабиринта соответствуют требованию, что, начав движение в этой клетке и выполнив предложенную программу, РОБОТ уцелеет и остановится в закрашенной клетке (клетка F6)?

НАЧАЛО

ПОКА **снизу свободно** ИЛИ **справа свободно**

ЕСЛИ **снизу свободно**

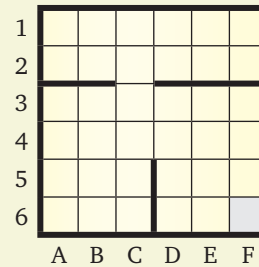
ТО **вниз**

ИНАЧЕ **вправо**

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

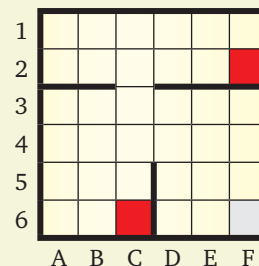


Решение

Привычная задача про РОБОТа (типа A13) теперь “обзавелась” новой особенностью алгоритма работы — полным ветвлением. Но принципы решения — все те же.

Анализируем предложенную программу РОБОТа.

• В начале очередного прохода цикла проверяется: есть ли стенки снизу и справа. Если хотя бы одно из этих направлений свободно, то выполняется очередной проход цикла. Значит, точка останова (клетка, в которой выполнение цикла корректно прекратится) имеет вид: \perp . Причем “правильной” является только одна такая клетка, а остальные две (позначим их красным цветом) — это “ловушки” для РОБОТа.

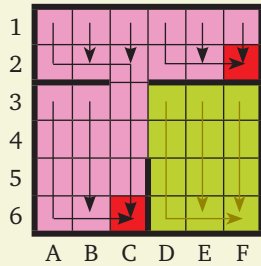


• Когда начинается очередной проход цикла, проверяется наличие стенки снизу. Если ее нет, то выполняется перемещение на одну клетку вниз. Если же стенка снизу есть, то выполняется перемещение на одну клетку вправо, причем *без проверки наличия стенки справа*. Следовательно, все клетки, в которых есть стенки снизу и есть стенки справа, можно было бы сразу пометить как непригодные (начав движение из них, РОБОТ разобьется). Но это не требуется: ведь такое условие записано в цикле ПОКА, который контролирует каждый отдельный шаг РОБОТа. (Вот если бы в цикле ПОКА было записано два или больше таких оператора ЕСЛИ, то для второго и т.д. такого оператора пришлось бы это исключение клеток сделать.)

• После выполнения одного такого хода проход цикла завершается, и вновь выполняется проверка его условия. Нетрудно понять, что в этом случае движение РОБОТа будет следующим: “идти вниз, пока это возможно (до стенки снизу), а наткнувшись на стенку снизу — пытаться уходить вправо, отыскивая проход вниз в этой стенке”.

Именно так мы и будем пытаться “трассировать” движение РОБОТа, “честно” выполняя предложенную программу. Увидев же, куда в итоге придет

РОБОТ, помечаем все пройденные клетки или розовым цветом (если попадаем в “ловушку”), или зеленым (если доходим до клетки F6). Сама же клетка F6 нам тоже подходит: если РОБОТ изначально находится в ней, то он просто никуда не пойдет и сразу же выполнит условие задачи.



Остается только подсчитать количество клеток, закрашенных зеленым: их 12.

Ответ: 12.

13. У исполнителя Аккорд две команды, которым присвоены номера:

1. отними 1

2. умножь на x

(x — неизвестное положительное число).

Выполняя первую из них, Аккорд отнимает от числа на экране 1, а выполняя вторую, умножает это число на x .

Программа для исполнителя Аккорд — это последовательность номеров команд.

Известно, что программа 12121 переводит число 4 в число 23.

Определите значение x .

Решение

Задача подобна прежним заданиям типа В1, но теперь нам нужно не составлять программу для исполнителя при известных командах, а, по сути, определить одну из команд при известной программе.

1) Попробуем расписать предложенную программу для исполнителя Аккорд, подставляя пока во второй команде значения x и заполнив левую колонку заданными исходным и конечным числами:

4	1	отними 1
	2	умножь на x
	1	отними 1
	2	умножь на x
	1	отними 1
23		

2) Очевидно, что для первой команды можно сразу записать результат ее выполнения. Точно так же можно определить число, которое было перед выполнением последней команды:

4	1	отними 1
3	2	умножь на x
	1	отними 1
	2	умножь на x
24	1	отними 1
23		

3) Вместо оставшихся числовых значений запишем соответствующие математические выражения и определим, какое математическое выражение должно давать известное нам предпоследнее число:

	4	1	отними 1
	3	2	умножь на x
	$3x$	1	отними 1
	$3x - 1$	2	умножь на x
	$x(3x - 1) = 24$	1	отними 1
	23		

4) Тем самым мы получили “ключевое” уравнение, позволяющее вычислить x :

$$x(3x - 1) = 24.$$

Решаем его как обычное квадратное уравнение:

$$3x^2 - x - 24 = 0;$$

$$D = 1 + 4 \cdot 3 \cdot 24 = 289 = 17^2;$$

$$x_{1,2} = (1 \pm 17)/(2 \cdot 3), \text{ тогда } x_1 = 18/6 = 3, \\ x_2 = -16/6 = -2^2/3.$$

Поскольку указано, что x — это положительное число, нам подходит только значение 3.

Ответ: 3.

14. Ниже приведен фрагмент программы. При каком наименьшем введенном числе a после выполнения программы значение переменной c будет равно 60?

```
readln(a);
b := 50;
a := a * 2 - b;
if a < b then
  c := b - 10
else
  c := b + 10;
```

Решение

В отличие от прежних заданий типа В2, здесь требуется не просто формально “протрассировать” заданную программу, а проанализировать алгоритм ее работы и понять, какими могут быть значения введенного числа a .

1) Цикла здесь нет, и изменение значения переменной b выполняется однократно (одним из операторов в ветвях then или else).

2) Изначально $b = 50$, а нам нужно, чтобы это значение стало равным 60. Следовательно, нам нужно для этого выполнить оператор $c := b + 10$ в ветви else.

3) Чтобы выполнялась ветвь else, требуется, чтобы к моменту выполнения оператора if уже пересчитанное значение a ($a := a * 2 - b$) было не меньше (т.е. больше или равно) b , т.е. числа 50.

4) Решаем полученное неравенство:

$$a \cdot 2 - 50 \geq 50; \Rightarrow 2a \geq 100; \Rightarrow a \geq 50.$$

5) Это — почти уже ответ. Нам же нужно определить, какое возможное значение a является наименьшим. Так как неравенство нестрогое, это значение $a = 50$.

Ответ: 50.

15. Сколько существует различных символьных последовательностей длины 5 в трехбуквенном алфавите {K, O, T}, которые содержат ровно две буквы O?

Решение

Задача подобна бывшим когда-то заданиям, в которых требовалось определить, какое слово стоит на указанном месте в списке слов, “синтезированных” из букв и отсортированных по порядку, например:

AAAAA
AAAAO
AAAOA
AAAOO
AAOAA
...

Нетрудно вспомнить, что такие задачи мы решали, “переводя” буквы в цифры соответствующей системы счисления и работая с полученными числами. Здесь задача несколько иная, но принцип решения будет аналогичен.

1) Поскольку никаких слов изначально не задано, мы можем сами “назначить” цифры, соответствующие каждой букве. А поскольку букв в алфавите три, мы имеем дело с троичной системой счисления и выберем следующие соответствия букв и цифр:

O – 0, K – 1, T – 2.

Почему мы выбрали нуль именно для буквы O? А потому, что нам понадобится искать слова, содержащие ровно две буквы O, а с нулями это, возможно, окажется проще.

2) Итак, речь идет об определении количества пятиразрядных троичных чисел (причем незначащие нули слева важны!), содержащих ровно два нуля.

3) Первый нуль может быть в одной из пяти позиций — получаем пять вариантов.

Как в каждом из этих вариантов может располагаться второй нуль? Посмотрим:

- когда первый нуль находится в позиции “1”, второй нуль может располагаться в одной из четырех оставшихся позиций — “2”, “3”, “4”, “5”;

- когда первый нуль находится в позиции “2”, второй нуль может располагаться в одной из трех оставшихся позиций правее — “3”, “4” и “5” (ведь ситуацию, когда нули располагаются в позициях “1” и “2”, мы уже рассмотрели перед этим);

- когда первый нуль находится в позиции “3”, второй нуль может располагаться в одной из двух оставшихся позиций правее — “4” и “5” (почему, — рассмотрено выше);

- когда первый нуль находится в позиции “4”, второй нуль может располагаться только в одной позиции правее — “5”.

Итого получаем $4 + 3 + 2 + 1 = 10$ вариантов размещения в пятиразрядном числе двух нулей.

Возможны и другие рассуждения, приводящие к тому же результату.

Условно обозначим наши нули разными цветами: например, первый — синим, а второй — красным.

Тогда синий нуль можно разместить в пяти рядах — получаем пять вариантов. И в каждом из этих пяти вариантов красный нуль можно разместить в любой из оставшихся четырех позиций — т.е. по четырем вариантам. Значит, всего получаем $5 \cdot 4 = 20$ вариантов размещения наших разноцветных нулей.

Но теперь вспомним, что на самом деле оба нуля (синий и красный) — совершенно равноправны. А значит, пары “синий — красный” и “красный — синий” (если читать число слева направо) — это одни и те же пары. Следовательно, каждая такая пара посчитана дважды, и всего вариантов размещения в пятиразрядном числе двух одинаковых нулей будет 10.

Итак, существует десять вариантов размещения в числе двух нулей. В каждом из этих десяти вариантов остается три цифры, которые могут быть равны или 1, или 2.

Сколько может быть таких неповторяющихся комбинаций? Очевидно, столько, сколько может быть различных трехразрядных чисел в системе счисления, состоящей из двух цифр (т.е. двоичной). Значит, для каждого из десяти ранее найденных вариантов получается $2^3 = 8$ “подвариантов”.

Тогда общее число троичных чисел, в которых из пяти цифр ровно две — нулевые, будет равно $8 \cdot 10 = 80$.

Ответ: 80.

16. Ниже приведена программа. При каком наибольшем введенном числе d после выполнения программы будет напечатано 55?

```
var n, s, d: integer;
begin
  readln(d);
  n := 0;
  s := 0;
  while s <= 365 do
  begin
    s := s + d;
    n := n + 5;
  end;
  write(n);
end.
```

Решение

Опять же, задача по виду подобна прежним заданиям В5, но ранее там требовалось определить конечное значение вычисляемой переменной, а теперь — наоборот, искать по заданному конечному значению исходное.

1) Анализируем алгоритм:

- в цикле ПОКА задано условие выполнения цикла — до тех пор, пока значение s не превышает 365;
- внутри цикла значение переменной s увеличивается на заданное значение d , а значение переменной n увеличивается на 5;
- по завершении цикла на экран выводится значение переменной n .

2) Тогда, согласно условию, значение n равно 55. Учитывая, что изначально переменная n

обнуляется, это означает, что цикл выполнен 55/5 = 11 раз.

3) Учитывая, что переменная s изначально тоже обнуляется, получаем, что за эти 11 проходов цикла переменная s увеличивается на $11d$ и становится больше 365 (условие прекращения работы цикла). Тогда получаем неравенство:

$$11d > 365,$$

откуда $d > 365/11$, т.е. $d > 33^{2/11}$, или, учитывая, что числа — целые, $d > 33$.

4) Чтобы теперь оценить верхнюю границу интервала возможных значений d , надо учесть, что при слишком большом значении d цикл мог бы прерваться еще после десяти проходов (а то и раньше). Значит, нам нужно определить значение d , на единицу меньше, чем минимальное значение d , обеспечивающее выполнение десяти проходов цикла. В этом случае неравенство будет таким:

$$10d > 365,$$

тогда $d > 365/10$, или $d > 36,5$.

5) Рассуждая аналогично п. 3, можно заключить, что минимальное значение d , при котором цикл завершится через десять проходов, равно 37 (ближайшее целое число, большее 36,5). Нам же требуется значение d , на единицу меньше. Это — число 36.

Таким образом, диапазон возможных значений d , обеспечивающих выполнение 11 проходов цикла, составляет [34 .. 36], а искомое максимальное значение d , при котором будет выведено число 55, равно 36.

Ответ: 36.

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

$$F(n) = 1 \text{ при } n \leq 2;$$

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

Чему равно значение функции $F(8)$?

Решение

Это — задача на рекурсию, практически неизменно повторяющая задание типа В6 из демоварианта 2014 года.

Решение заключается в аккуратном расписывании всех значений $F(n)$ для возрастающих значений n от 1 до требуемого значения 8, при этом для вычисления очередного значения используются ранее вычисленные предыдущие значения $F(n)$:

$$F(0) = 1;$$

$$F(1) = 1;$$

$$F(2) = F(0) \cdot (2+2) = 4;$$

$$F(3) = F(1) \cdot (3+2) = 5;$$

$$F(4) = F(2) \cdot (4+2) = 24;$$

$$F(5) = F(3) \cdot (5+2) = 35;$$

$$F(6) = F(4) \cdot (6+2) = 192;$$

$$F(7) = F(5) \cdot (7+2) = 315;$$

$$F(8) = F(6) \cdot (8+2) = 1920.$$

Последнее вычисленное значение — и есть искомый ответ. Заметим, что $F(7)$ для получения ответа на вопрос задачи можно и не вычислять.

Ответ: 1920.

18. Известно, что для целого числа x выполнено:

$$33_{4+x} - 33_4 = 33_{10}.$$

Определите значение x . Ответ запишите в десятичной системе счисления.

Решение

По сути, это — несколько усложненное (по формулировке условия) и вместе с тем — немного упрощенное (по степени сложности решения) © задание типа В7 на поиск основания системы счисления.

1) Запишем “каноническое” преобразование обоих недесятичных чисел в десятичную систему счисления (согласно формуле вида $a_1 \cdot n^1 + a_0 \cdot n^0$):

$$\bullet 33_{4+x} = 3 \cdot (4+x)^1 + 3 \cdot (4+x)^0 = \\ = 3 \cdot (4+x) + 3 = 12 + 3x + 3 = 15 + 3x;$$

$$\bullet 33_4 = 3 \cdot 4^1 + 3 \cdot 4^0 = 3 \cdot 4 + 3 = 15.$$

2) Теперь, когда все числа переведены в десятичную систему счисления, можно записать уравнение в виде:

$$15 + 3x - 15 = 33,$$

$$\text{откуда } 3x = 33, \text{ тогда } x = 11.$$

Ответ: 11.

19. Записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: a и b . Сколько существует таких чисел x , при вводе которых алгоритм печатает сначала 2, а потом 15?

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

Решение

Нетрудно увидеть, что эта задача — почти аналог прежнего задания типа В8. Но если ранее надо было определить одно входное число (например, наименьшее из возможных), то теперь у нас спрашивают их количество.

1) Как всегда, начинаем с анализа алгоритма:

• цикл ПОКА выполняется до тех пор, пока x не станет нулем;

• в цикле значение переменной a каждый раз увеличивается на 1, следовательно, a — это счетчик количества проходов цикла;

• переменная b служит для накопления суммы значений $x \bmod 10$, а затем переменная x (исходное число) переприсваивается на значение $x \div 10$. Такой алгоритм должен знать каждый школьник: это — разбор в цикле исходного числа x на отдельные цифры (справа налево), а в переменной b , таким образом, вычисляется сумма цифр исходного числа;

• после цикла выводится сначала значение переменной a , а затем — переменной b .

2) По условию, в ходе выполнения алгоритма были выведены числа 2 и 15, — значит, $a = 2$, $a b = 15$. Следовательно, цикл выполнился два раза (a значит, в исходном числе x было две цифры), а сумма цифр этого числа равна 15.

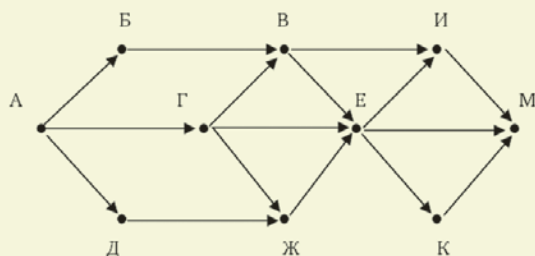
3) Для ответа на вопрос задачи нужно определить, сколько существует двузначных натуральных³ чисел, сумма цифр которых равна 15. Их нетрудно найти простым перебором (учитывая и то, что суммы цифр могут быть и “симметричными”). Для этого по очереди предполагаем первую цифру равной 1, 2, 3 и т.д. и определяем, чему может быть равна вторая цифра. Получаем:

69, 96, 78, 87.

Всего — четыре числа.

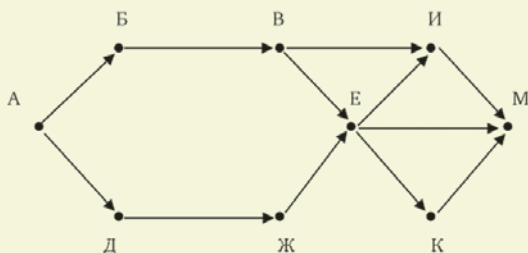
Ответ: 4.

20. На рисунке — схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К, М. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей, ведущих из города А в город М и **не проходящих** через город Г?

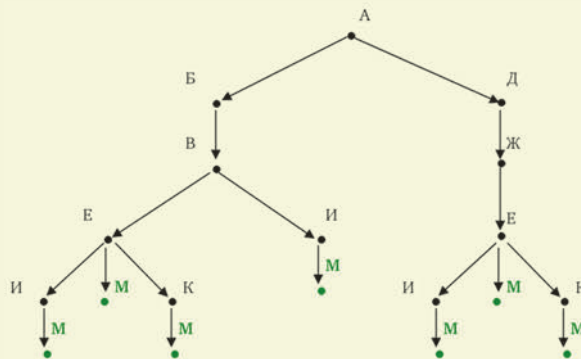


Решение

Такие задачи (тип В9) мы решать уже умеем (см. статью: Богомолова О.Б., Усенков Д.Ю. ЕГЭ: возможные особенности заданий 2014 года // “Информатика”, 2014, № 5. С. 4–21). Ведь применяемый для этого обычно способ с построением дерева путей универсален: он дает все возможные пути из одного города в другой, а уж найти среди них проходящие через “запретный” для нас город Г и исключить их из подсчета легко. Но мы сделаем еще проще (и тем самым существенно облегчим себе решение): раз через город Г идти нельзя, его можно сразу исключить из схемы!



Теперь и строить дерево путей гораздо проще:



Остается подсчитать количество конечных точек, соответствующих городу М (выделены в дереве зеленым цветом). Их — 7.

Ответ: 7.

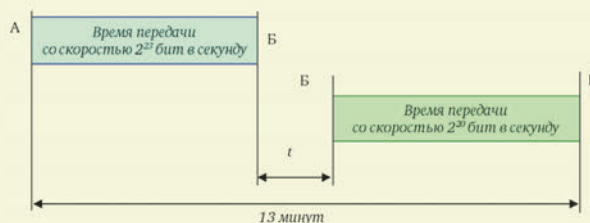
21. Данные объемом 80 Мбайт передаются из пункта А в пункт В по каналу связи, обеспечивающему скорость передачи данных 2^{23} бит в секунду, а затем из пункта В в пункт В по каналу связи, обеспечивающему скорость передачи данных 2^{20} бит в секунду. От начала передачи данных из пункта А до их полного получения в пункте В прошло 13 минут.

Через какое время в секундах началась передача данных в пункте В, т.е. каково время между началом передачи данных из пункта А и началом передачи данных в пункт В? В ответе укажите только число, слово “секунд” или букву “с” добавлять не нужно.

Решение

Ну а это — хорошо нам знакомая задача на передачу данных (см. статью: Богомолова О.Б., Усенков Д.Ю. “Задача о передаче”: решение задачи ЕГЭ при помощи сетевых диаграмм Ганта // “Информатика”, 2011, № 7. С. 7–8), которая предлагалась на ЕГЭ три-четыре года назад. Только там нам нужно было искать общее время передачи данных из А в В, а теперь придется составлять уравнение с одним неизвестным.

1) Построим диаграмму Ганта для этой задачи (как это делается, читатели могут узнать в вышеуказанной статье, а здесь мы вдаваться в подробности не будем):



При этом следует учитывать, что процесс передачи данных из В в В по условию задачи начинается уже после того, как завершится передача данных из А в В.

2) Составляем по этой схеме уравнение (t — это время задержки между окончанием приема данных из А и началом их передачи в В). Для этого вычисляем время передачи данных из А в В и из В в В:

³ То, что вводимое число x — натуральное, в условии не оговаривается, но предполагается, так как при $x \leq 0$ заданный в программе цикл работать не будет вовсе. — Прим. авт.

- из А в В: $80 \text{ (Мбайт)} / 2^{23} \text{ (бит в секунду)} = 5 \cdot 2^{27} \text{ (бит)} / 2^{23} \text{ (бит в секунду)} = 5 \cdot 2^4 = 80 \text{ (с)}$;
- из В в В: $80 \text{ (Мбайт)} / 2^{20} \text{ (бит в секунду)} = 5 \cdot 2^{27} \text{ (бит)} / 2^{20} \text{ (бит в секунду)} = 5 \cdot 2^7 = 640 \text{ (с)}$.

Записываем уравнение:

$$80 + t + 640 = 13 \text{ (минут)} = 13 \cdot 60 \text{ (с)}, \text{ т.е. } 720 + t = 780, \text{ откуда } t = 60.$$

3) Чтобы найти время от начала передачи данных из А в В до начала их передачи из В в В, нужно к найденному значению t прибавить время передачи данных из А в В:

$$80 + 60 = 140 \text{ (с)}.$$

Ответ: 140.

22. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес — в виде четырех байтов, причем каждый байт записывается в виде десятичного числа. При этом в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места — нули. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.

Например, если IP-адрес узла равен 231.32.255.131, а маска равна 255.255.240.0, то адрес подсети равен 231.32.240.0.

Для узла с IP-адресом 224.128.112.142 адрес сети равен 224.128.64.0. Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.

Решение

Такие задачи на определение адреса сети по заданному IP-адресу и маске уже всем знакомы (тип В11). Но здесь нам нужно искать не адрес сети, а, наоборот, маску по заданным IP-адресу и адресу сети. Причем поскольку для байтов, которые в указанных адресах совпадают, байт маски очевидно равен 255, а там, где в адресе сети нулевой байт, байт маски тоже, очевидно, равен нулю, экзаменаторов интересует именно “ключевой” байт маски — третий слева. Так что задача для нас даже немного упростилась.

1) Преобразуем заданные третьи слева байты исходного IP-адреса и адреса сети в двоичные числа:

- $112_{10} = 1110000_2$;
- $64_{10} = 1000000_2$.

2) Для получения из исходного IP-адреса заданного адреса сети требовалась поразрядная конъюнкция (для каждого бита) IP-адреса и маски, т.е.:

$$\begin{array}{r} 1110000 \\ \& \text{ *****} \\ 1000000 \end{array}$$

Биты маски нам не известны. Но их очень легко определить:

- там, где значения битов IP-адреса и адреса сети оба равны 1, бит маски должен быть тоже равен 1;
- там, где бит IP-адреса равен 1, а бит адреса сети равен 0, бит маски должен быть равен 0.

Единственная неоднозначность — там, где биты IP-адреса и адреса сети оба равны нулю: тут бит маски может быть равен как 0, так и 1. Но устранить эту неоднозначность нетрудно, зная, что в маске при ее просмотре слева направо может быть только один переход от единичных битов к нулевым, т.е., например, маска 11000110 недопустима. А мы видим, что уже со второго слева бита в маске должны начаться нули. Значит, и во всех “спорных” битах маски правее этого места тоже должны быть только нули.

3) Теперь нетрудно восстановить вышеприведенный “ребус” с конъюнкцией двоичных чисел:

$$\begin{array}{r} 1110000 \\ \& \underline{1000000} \\ 1000000 \end{array}$$

4) Остается перевести найденную двоичную маску в десятичное число. Для этого даже не понадобится никаких вычислений — полученная маска полностью совпала с двоичной записью байта адреса сети (64_{10}), значит, и искомым байт маски тоже равен 64.

Ответ: 64.

23. В языке запросов поискового сервера для обозначения логической операции “ИЛИ” используется символ “|”, а для логической операции “И” — символ “&”.

В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Найдено страниц (в тысячах)
США Япония Китай	450
Япония Китай	260
(США & Япония) (США & Китай)	100

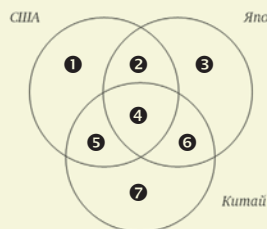
Какое количество страниц (в тысячах) будет найдено по запросу США?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Решение

Еще одна хорошо знакомая нам задача (тип В12). Но один из запросов, приведенных в таблице, стал сложнее. Впрочем, принцип решения остается прежним (см. статью: Богомолова О.Б., Усенков Д.Ю. «В поисках поиска»: задачи ЕГЭ, посвященные поиску информации на сайтах // “Информатика”, 2011, № 11. С. 17–23).

1) Рисуем примерную диаграмму Эйлера – Венна и нумеруем области на ней:



2) Записываем уравнения с “номера-переменными” по заданным в условии запросам. При этом особое внимание обратим на сложный запрос в третьей строке таблицы.

США Япония Китай	$\textcircled{1} + \textcircled{2} + \textcircled{3} + \textcircled{4} + \textcircled{5} + \textcircled{6} + \textcircled{7} = 450$
Япония Китай	$\textcircled{2} + \textcircled{3} + \textcircled{4} + \textcircled{5} + \textcircled{6} + \textcircled{7} = 260$
(США & Япония) (США & Китай) = США & (Япония Китай)	$\textcircled{2} + \textcircled{4} + \textcircled{5} = 100$
США	$\textcircled{1} + \textcircled{2} + \textcircled{4} + \textcircled{5} = ?$

3) Теперь главное — найти, как удобнее всего решить эту систему уравнений. Но это неожиданно легко.

Сразу очевидно, что из уравнений:

$$\textcircled{1} + \textcircled{2} + \textcircled{3} + \textcircled{4} + \textcircled{5} + \textcircled{6} + \textcircled{7} = 450$$

и

$$\textcircled{2} + \textcircled{3} + \textcircled{4} + \textcircled{5} + \textcircled{6} + \textcircled{7} = 260$$

можно найти значение $\textcircled{1}$, равное $450 - 260 = 190$.

Тогда, зная, что $\textcircled{2} + \textcircled{4} + \textcircled{5} = 100$, сразу можно вычислить, что $\textcircled{1} + \textcircled{2} + \textcircled{4} + \textcircled{5} = 190 + 100 = 290$.

Ответ: 290.

24. Определите, какое значение H нужно ввести, чтобы число, напечатанное в результате выполнения следующего алгоритма, было наименьшим.

```

var a,b,t,M,R,H: integer;
Function F(H, x: integer): integer;
begin
    F := 11*(x - H)*(x - H) + 13;
end;
begin
    readln(H);
    a := -10; b := 50;
    M := a; R := F(H, a);
    for t := a to b do begin
        if (F(H, t) > R) then begin
            M := t;
            R := F(H, t);
        end;
    end;
    write(R);
end.
    
```

Решение

А это — несколько усложненное задание типа В14 (с параболой). Такие задачи уже встречались ранее, но тогда требовалось для квадратичной функции одной переменной $F(x)$ найти, какое число выводится в результате. А теперь у нас — функция с двумя переменными $F(H,x)$, которая представляет собой выражение с параметром H , и искать надо значение этого параметра, такое, чтобы выводимое значение было наименьшим.

Прежде всего заметим, что работа в языке программирования (в данном случае в Паскале, но это верно и для других языков — по крайней мере для большинства из них) с функциями с одной и с двумя переменными полностью идентична: в обоих случаях в функцию передаются значения переменных,

а результат возвращается через саму функцию $F()$. Поэтому сосредоточимся на собственно решении.

1) Анализируем алгоритм.

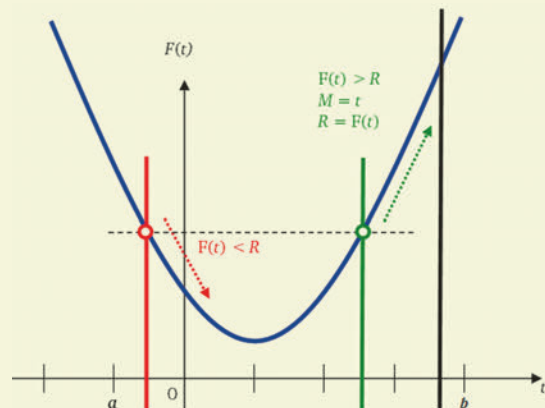
- В цикле идет перебор значений t от заданных ранее $a = -10$ до $b = 50$. При этом в переменной M запоминается сначала начальное значение a , а потом (при срабатывании условного оператора) — текущее значение t , а в переменной R хранится предыдущее значение функции $F(H,x)$.

- Если с увеличением значения t получаемое значение $F()$ увеличивается (и в результате оказывается больше предыдущего значения $F(t)$, запомненного в переменной R), текущее значение t заносится в переменную M , а значение функции в этой точке запоминается в R . То есть ищется максимум функции $F()$ на интервале $[a; b]$.

- До каких пор это будет продолжаться? Посмотрим на выражение, записанное в составе функции $F()$. Выполним в нем перемножение скобок. Получим:

$$11*(x-H)*(x-H) + 13 = 11x^2 - 22xH + (11H^2 + 13).$$

Графиком этой функции от x является парабола. Поскольку перед x^2 стоит знак “плюс”, ветви этой параболы направлены вверх. И мы в своих вычислениях “движемся” по левой ветви этой параболы сверху вниз. Значит, условный оператор в цикле не будет работать до тех пор, пока мы не пройдем вершину параболы (минимум на ее графике) и не поднимемся по правой ветви настолько, что значение выражения $11t^2 - 22tH + (11H^2 + 13)$ (замена переменной x на t происходит при переходе из основной программы в функцию) в некоторой точке t превысит его значение в исходной точке a . А после этого, пока мы “движемся” по правой ветви параболы вверх, у нас значение R будет расти вплоть до окончания цикла по достижении точки b (см. примерный рисунок). Следовательно, конечное значение R будет равно значению выражения $F(H,b) = 11b^2 - 22bH + (11H^2 + 13)$ либо, если расположить параболу так, что указанное значение меньше чем исходное ($F(H,a) = 11a^2 - 22aH + (11H^2 + 13)$), значение R будет равно этому исходному.



2) Как нужно расположить параболу, чтобы получить минимально возможное значение R ? И каким оно будет? Очевидно, что если ветвь then нач-



ОГЭ на “ого!”: новые ~~подарки~~ задачи для 9-го класса

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

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

► После апробации новых заданий ЕГЭ пришло время и для новых заданий ОГЭ (бывшего ГИА). В “экспериментальном” варианте ОГЭ по информатике тоже почти все задания (кроме одного) оказались принципиально новыми по формулировкам. Впрочем, в отличие от ЕГЭ задания для ОГЭ остались достаточно простыми: как правило, требуется лишь разобраться в новой формулировке, немного подумать и... догадаться, как начать решение.

Познакомим с принципами решения задач из одного из вариантов

апробационной работы наших читателей (поскольку запрет на их публикацию не указан).

1. В одной из кодировок Unicode каждый символ кодируется 16 битами. Вова написал текст:

“Ёж, лев, слон, олень, тюлень, носорог, крокодил, аллигатор — дикие животные”.

Дима вычеркнул из списка двух стоящих рядом зверей. Заодно он вычеркнул ставшие лишними запятые и пробелы — два пробела не должны идти подряд. При этом размер нового предложения в данной кодировке оказался на 26 байт меньше, чем размер исходного предложения. Напишите в ответе более короткое из удаленных слов. Запятые и пробелы писать не нужно.

Решение

1) После вычеркивания объем текста уменьшился на 26 байт. При этом в используемой кодировке Unicode каждый символ кодируется 16 битами, т.е. двумя байтами. Следовательно, вычеркнуто было $26/2 = 13$ символов.

2) Вычеркнуто два названия животных вместе с запятыми и пробелами после них. После каждого названия животного записано по одной запятой и по одному пробелу, значит, вместе с животными было вычеркнуто четыре символа (две запяты + два пробела). Следовательно, на сами вычеркнутые названия животных приходится $13 - 4 = 9$ символов.

3) Заметим: названия животных в тексте подобраны так, что количества букв в них возрастают от 2 до 9 с шагом в одну букву. Вычеркнуты два соседних названия животных. Значит, нужно найти такие два названия, в которых количества букв отличаются на единицу, а сумма количеств букв равна 9. Для этого достаточно вычислять суммарные количества букв в парах названий начиная с начала текста:

- “ёж + лев”: $2 + 3 = 5$ — не подходит (слишком мало),
- “лев + слон”: $3 + 4 = 7$ — не подходит (слишком мало),
- “слон + олень”: $4 + 5 = 9$ — годится,
- “олень + тюлень”: $5 + 6 = 11$ — не подходит (слишком много); для всех последующих пар названий сумма количеств букв будет еще большей.

Следовательно, были вычеркнуты слова “слон” и “олень”.

4) В задаче требуется указать более короткое из удаленных слов. Очевидно, это “слон”.

Ответ: слон.

2. Напишите наименьшее число x , для которого ложно высказывание:

НЕ ($x > 80$) ИЛИ (x кратно девяти).

Решение

1) Временно заменим элементарные высказывания логическими переменными:

- ($x > 80$) = A ,
- (x кратно девяти) = B .

Тогда исходное выражение можно записать так:

$$\bar{A} \vee B = 0.$$

2) Логическая операция, выполняемая последней, — операция ИЛИ. Ее результат равен “ЛОЖЬ” в одном-единственном случае — когда оба операнда ложны. Тогда этому выражению будет эквивалентна следующая система уравнений (с учетом отрицания переменной A):

$$\begin{cases} A = 1, \\ B = 0. \end{cases}$$

3) Вернемся к исходным значениям логических переменных.

Рассмотрим первое уравнение: $(x > 80) = 1$. Оно означает, что нам подходят любые натуральные значения x , большие 80 (не включая значение 80, так как неравенство строгое).

Рассмотрим второе уравнение: $(x$ кратно девяти) = 0. Оно означает, что из найденного выше диапазона допустимых натуральных чисел нужно вычеркнуть числа, кратные 9 (например, 81, 90 и т.д.).

4) Нам требуется указать минимальное значение x , удовлетворяющее этим правилам. Таковым является значение 82.

Ответ: 82.

3. Между населенными пунктами A, B, C, D, E построены дороги, протяженность которых (в километрах) приведена в таблице:

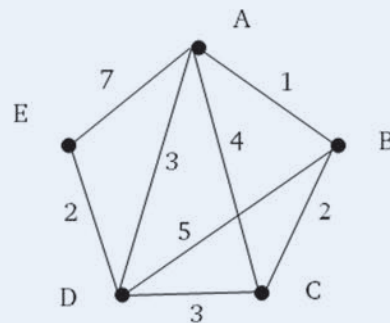
	A	B	C	D	E
A		1	4	3	7
B	1		2	5	
C	4	2		3	
D	3	5	3		2
E	7			2	

Определите длину кратчайшего пути между пунктами A и E , проходящего через пункт C . Передвигаться можно только по дорогам, протяженность которых указана в таблице.

Решение

Задача отличается от аналогичных, приводившихся в предыдущих версиях ГИА, тем, что требуется определить кратчайший путь не среди всех возможных путей, а только среди тех, которые проходят через пункт C . В остальном решение выполняется аналогично предыдущим задачам.

1) Строим по таблице граф дорог:



2) Определяем все пути из A в E , отбрасывая среди них не проходящие через C , и вычисляем длины этих путей:

- AE — не годится (не проходит через C);
- ADE — не годится (не проходит через C);
- $ACDE$ — годится, длина равна $4 + 3 + 2 = 9$ км;
- $ABDE$ — не годится (не проходит через C);
- $ABCDE$ — годится, длина равна $1 + 2 + 3 + 2 = 8$ км.

Кратчайший путь из A в E среди пригодных — это путь $ABCDE$, он имеет длину 8 км.

Ответ: 8.

4. Файл **Робокон.doc** имеет полное имя:

C:\2014\Супергерои\Полицейские\Придурманные\Робокон.doc,

а файл **Робокон1.doc** имеет полное имя

C:\2014\Фильмы\Полицейские\США\Робокон1.doc.

Файловый менеджер позволяет за один шаг перенести файл из каталога в любой из его подкаталогов или в каталог, содержащий данный каталог.

Какое наименьшее количество шагов понадобится, чтобы перенести файл **Робокон.doc** в тот каталог, где находится файл **Робокон1.doc**?

Решение

Прежде всего заметим: указание “Файловый менеджер позволяет за один шаг перенести файл из каталога в любой из его подкаталогов или в каталог, содержащий данный каталог” означает, что за один шаг можно переместить файл на один уровень выше или ниже в дереве каталогов.

1) Анализируем приведенные строки полных записей имен файлов, а точнее — их части, соответствующие путям к этим файлам, и ищем совпадающие части в начале этих путей: “C:\2014\”. Следовательно, это та точка в файловом дереве, где происходит требуемое нам ветвление каталогов.

2) Теперь можно изобразить данный фрагмент файлового дерева следующим образом (особо обратим внимание: каталоги **Полицейские** — это **разные** каталоги, так как они расположены в различных ветвях дерева):



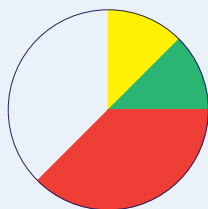
3) Теперь уже нетрудно подсчитать минимально необходимое количество шагов по перемещению файла из каталога “Придуманные” (левая ветвь в дереве) в каталог “США” (правая ветвь). Для этого достаточно подсчитать количество переходов между каталогами (“стрелочек”) на кратчайшем пути из одного каталога в другой. Их — 6. Это и есть ответ к данной задаче.

Ответ: 6.

5. Дан фрагмент электронной таблицы:

	A	B	C	D
1	1	???	4	???
2	???	=C1-A1	???	=C1+D1

В ячейки диапазона A1:D1 записаны натуральные числа, а в ячейки диапазона A2:D2 — формулы. На рисунке изображена диаграмма, построенная по значениям диапазона ячеек A2:D2. Известно содержимое только ячеек A1, C1, B2, D2. Определите, какое число записано в ячейке D1. Значения во всех ячейках диапазона A1:D2 положительны.



Решение

Задача выглядит существенно более сложной, чем даже аналогичные задания ЕГЭ. Но посмотрим, можно ли упростить ее решение.

1) Попробуем произвести в таблице все возможные вычисления, обозначив искомое значение в ячейке D1 за x :

	A	B	C	D
1	1	???	4	x
2	???	3	???	$4+x$

2) Анализируем диаграмму. На ней — две пары одинаковых секторов, расположенных рядом друг с другом. Следовательно, в диапазоне A2:D2, по которому строится эта диаграмма, будут соседствовать одинаковые значения. Учитывая, что (согласно условию) x — натуральное число (т.е. неотрицательное), значения B2 и D2 не равны друг другу, а значит, соответствуют разным по размеру секторам. А поскольку в задаче нас интересует только значение в ячейке D1, можно без каких-либо трудностей считать, что значения в ячейках A2 и C2 равны, соответственно, значениям в ячейках B2 и D2. А значение в ячейке B1 тогда не участвует в вычислениях и может быть любым (в том числе можно считать эту ячейку пустой):

	A	B	C	D
1	1		4	x
2	3	3	$4+x$	$4+x$

3) На глаз можно оценить, что большие секторы вдвое больше, чем меньшие (поскольку значения в диапазоне A2:D2 получаются натуральные — операция деления не используется, — такой грубой оценки достаточно). Также очевидно, что значение $4 + x$ соответствует большему сектору, а значение 3 — меньшему (так как x — неотрицательное). Тогда можно составить уравнение:

$$4 + x = 2 \cdot 3,$$

откуда легко найти значение x : $x = 6 - 4 = 2$.

4) Проверяем, выполнив в таблицу подстановку найденного значения x :

	A	B	C	D
1	1		4	2
2	3	3	6	6

Очевидно, диаграмма, построенная по диапазону A2:D2, будет иметь требуемый вид.

Ответ: 2.

6. Исполнитель Чертежник перемещается по координатной плоскости, оставляя след в виде линии. Чертежник может выполнять команду **Сместиться на (a,b)**, где a, b — целые числа. Эта команда перемещает Чертежника из точки с координатами (x,y) в точку с координатами $(x+a,y+b)$. Если числа a, b положительные, значение соответствующей координаты увеличивается, если отрицательные — уменьшается.

Например, если Чертежник находится в точке с координатами (9,5), то команда **Сместиться на (1,-2)** переместит Чертежника в точку (10,3).

Команда	a	b
a := 8	8	
b := 3	8	3
b := a / 4 + b	8	8 / 4 + 3 = 5
a := 3 * a + 8 * b	3 * 8 + 8 * 5 = 64	5

Итак, после выполнения первой программы значение переменной a станет равно 64.

2) Аналогичным образом проводим трассировку второй программы, считая, что в первой ее строке переменной a присваивается значение, равное x .

Команда	a	b
a := x	x	
b := 3	x	3
b := a / 4 + b	x	x / 4 + 3
a := 3 * a + 8 * b	3 * x + 8 * (x / 4 + 3)	5

Тогда получаем, что после выполнения второй программы значение переменной a равно $3 * x + 8 * (x / 4 + 3)$.

3) Зная, что после выполнения программы 2 значение переменной a на 20 больше, чем после выполнения программы 1, можно составить уравнение:

$$3 * x + 8 * (x / 4 + 3) = 64 + 20.$$

Решаем это уравнение:

$3x + 2x + 8 * 3 = 84, \Rightarrow 5x = 84 - 24, \Rightarrow 5x = 60,$
откуда $x = 12$.

Ответ: 12.

9. Ниже приведена программа¹. В результате выполнения программы было напечатано число 45. Какое значение переменной d было введено?

```
var s, k, d: integer;
begin
  readln(d);
  s := 0;
  for k := 4 to d to
    s := s + 5;
  writeln(s);
end.
```

Решение

1) Анализируем текст программы. Переменная s , значение которой выводится на экран, служит для накопления суммы. При этом изначально ее значение равно нулю, а на каждом шаге цикла увеличивается на 5.

2) Если в конце работы программы на экран выведено значение s , равное 45, то сколько для этого потребовалось шагов цикла? Очевидно, $45 / 5 = 9$ шагов.

3) Цикл начинается со значения k , равного 4, и продолжается, пока k не станет равно введенному значению d . Чтобы такой цикл выполнялся 9 раз, надо, чтобы значение d было равно 12 (цикл вы-

¹ В заданиях такого типа обычно приводится несколько вариантов одной и той же программы на различных языках программирования. Мы же ограничимся только текстом программы на языке Паскаль. — Прим. авт.

полняется для значений k , равных 4, 5, 6, 7, 8, 9, 10, 11, 12, т.е. ровно 9 раз).

Ответ: 12.

10. В результате выполнения программы напечатано число 10. Какое наибольшее значение может иметь переменная S после выполнения программы?

```
var k, m, S, N: integer;
Dat: array[1..100] of integer;
begin
  N := 5;
  m := 0; S := 0;
  for k := 1 to N do begin
    readln(D[k]);
  end;
  for k := 1 to N do begin
    S := S + Dat[k];
    if Dat[k] > m then begin
      m := Dat[k];
    end;
  end;
  writeln(m);
end.
```

Решение

1) Анализируем программу и определяем назначение каждой из переменных в ней:

- N — определяет, какая часть массива Dat реально используется в вычислениях (из него используются элементы с индексами от 1 до N , то есть пять первых элементов);

- S — используется для накопления суммы этих элементов массива;

- m — изначально равно нулю, а затем переопределяется — в нее заносится значение элемента массива, если этот элемент больше, чем текущее значение m . Это — типичный алгоритм поиска максимума. Значит, в m определяется (и в конце работы программы выводится на экран) максимальное значение среди обрабатываемых N элементов массива.

2) Итак, из пяти значений массива максимальное равно 10. Каким может быть наибольшее возможное значение суммы этих элементов?

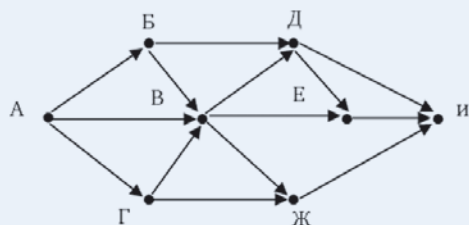
Очевидно, сумма будет наибольшей, если каждое из слагаемых (элементов массива) будет по возможности наибольшим. А наибольшее возможное значение элемента массива уже определено: оно равно 10. Если каждый элемент этой части массива будет равен 10, то вычисленный максимум будет равен 10. (Если какие-то элементы массива меньше 10, то это не даст максимально возможную сумму. Если же какие-то элементы массива больше 10, то максимум был бы равен уже не 10, а этому большему значению.)

3) Итак, заданному условию соответствует случай, когда каждый из обрабатываемых пяти элементов массива равен 10. Тогда их сумма (максимально возможная) будет равна $5 * 10 = 50$.

Ответ: 50.

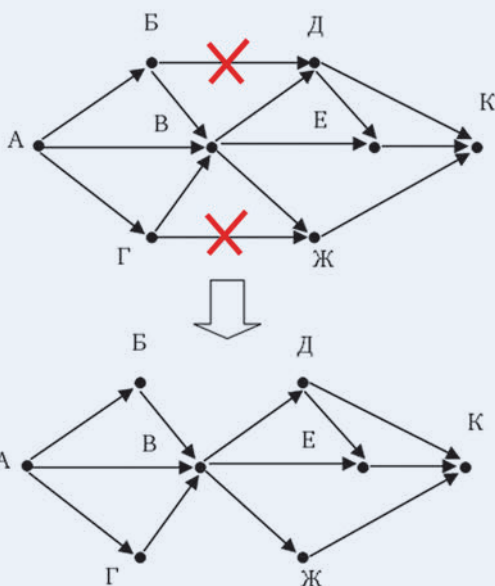
11. На рисунке — схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж и К. По каждой дороге

можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К, проходящих через город В?



Решение

1) Убираем из графа путей все стрелки, которые ведут в обход пункта В:

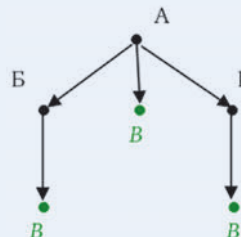


2) Как видим, задача существенно упростилась. Теперь достаточно будет подсчитать сначала возможное количество различных путей из А в В, а затем — количество возможных различных путей

из В в К, и перемножить эти два числа (так как для каждого пути из А в В возможно соответствующее число путей из В в К).

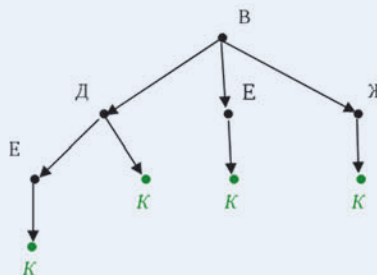
3) Для подсчета количества путей лучше всего строить *дерево путей*, это позволит учесть все возможные варианты и ни одного не пропустить.

а) пути из А в В (из А можно пройти в Б, В и Г; затем из Б и Г — соответственно, в В):



Подсчитываем количество ветвей в этом дереве (оно равно количеству конечных вершин, помеченных буквой В; такие вершины выделены зеленым цветом). Их — три, значит, из А в В возможно три пути.

б) пути из В в К — дерево путей строится аналогично:



Подсчитываем количество ветвей: оно равно 4.

4) Общее количество различных путей из А в К, проходящих через В, тогда равно $3 \cdot 4 = 12$.

Ответ: 12.

12. Ниже в табличной форме представлен фрагмент базы данных “Отправление поездов дальнего следования”:

Пункт назначения	Категория поезда	Время в пути	Вокзал
Пенза	скорый	14:28	Казанский
Пенза	фирменный	10:25	Казанский
Петрозаводск	скорый	13:27 ²	Ленинградский
Петрозаводск	фирменный	12:55	Ленинградский
Полоцк	пассажирский	11:35	Белорусский
Псков	фирменный	13:00	Ленинградский
Пятигорск	скорый	34:38	Курский
Ростов-на-Дону	скорый	27:32 ³	Курский
Ростов-на-Дону	скорый	24:28	Курский
Саратов	фирменный	15:05	Павелецкий
Саратов	фирменный	15:35 ⁴	Павелецкий
Северодвинск	фирменный	23:27	Ярославский

² В исходной задаче здесь стояло значение “13:97”, т.е. “13 часов 97 минут” ©. Поскольку нам важна именно правильность решения, а не совпадение ответов с предложенными в данном варианте, авторы сочли возможным выбрать взамен любое корректное значение. — Прим. авт.

³ В исходном варианте задачи — “27:82”. — Прим. авт.

⁴ В исходном варианте задачи — “15:95”. — Прим. авт.

Какое наименьшее целое число можно поставить в запросе
(Категория поезда = “скорый”) ИЛИ (Время в пути < X часов),
чтобы этому запросу удовлетворяло ровно шесть записей? В ответе укажите одно число — искомое значение X.

Решение

1) В запросе использована логическая связка ИЛИ. Это означает, что в результирующую выборку войдут все записи, соответствующие запросу (Категория поезда = “скорый”). Помечаем их в таблице голубым цветом:

Пункт назначения	Категория поезда	Время в пути	Вокзал
Пенза	скорый	14:28	Казанский
Пенза	фирменный	10:25	Казанский
Петрозаводск	скорый	13:27	Ленинградский
Петрозаводск	фирменный	12:55	Ленинградский
Полоцк	пассажирский	11:35	Белорусский
Псков	фирменный	13:00	Ленинградский
Пятигорск	скорый	34:38	Курский
Ростов-на-Дону	скорый	27:32	Курский
Ростов-на-Дону	скорый	24:28	Курский
Саратов	фирменный	15:05	Павелецкий
Саратов	фирменный	15:35	Павелецкий
Северодвинск	фирменный	23:27	Ярославский

2) Как видим, в выборку по запросу уже вошло пять записей. Чтобы в выборке было шесть записей, нужно так подобрать значение X, чтобы запросу (Время в пути < X часов) соответствовала еще одна запись (и, возможно, любое число записей среди уже отобранных). При этом замечаем, что нас интересует только количество часов (число до двоеточия, записанное в графе “Время в пути”); минуты можно не учитывать, так как нас интересует условие “меньше указанного количества часов”.

3) Для нахождения требуемого значения X выберем сначала наименьшее количество часов времени в пути, отмеченное в таблице (оно равно 10), и посмотрим, какие строки войдут в выборку (будем в них помечать зеленым цветом графы от “времени в пути” и правее):

Пункт назначения	Категория поезда	Время в пути	Вокзал
Пенза	скорый	14:28	Казанский
Пенза	фирменный	10:25	Казанский
Петрозаводск	скорый	13:27	Ленинградский
Петрозаводск	фирменный	12:55	Ленинградский
Полоцк	пассажирский	11:35	Белорусский
Псков	фирменный	13:00	Ленинградский
Пятигорск	скорый	34:38	Курский
Ростов-на-Дону	скорый	27:32	Курский
Ростов-на-Дону	скорый	24:28	Курский
Саратов	фирменный	15:05	Павелецкий
Саратов	фирменный	15:35	Павелецкий
Северодвинск	фирменный	23:27	Ярославский

Поскольку минимальное время в пути (10:27) больше значения 10, в выборку дополнительно не попадет ни одной строки. Значит, значение $X = 10$ нам не подойдет.

Попробуем следующее значение — $X = 11$:

Пункт назначения	Категория поезда	Время в пути	Вокзал
Пенза	скорый	14:28	Казанский
Пенза	фирменный	10:25	Казанский
Петрозаводск	скорый	13:27	Ленинградский
Петрозаводск	фирменный	12:55	Ленинградский
Полоцк	пассажирский	11:35	Белорусский
Псков	фирменный	13:00	Ленинградский
Пятигорск	скорый	34:38	Курский
Ростов-на-Дону	скорый	27:32	Курский
Ростов-на-Дону	скорый	24:28	Курский
Саратов	фирменный	15:05	Павелецкий
Саратов	фирменный	15:35	Павелецкий
Северодвинск	фирменный	23:27	Ярославский

Условие выполнено: для $X = 11$ в выборку войдет ровно шесть записей.

Проверим, что получится, если выбрать большее значение, например, $X = 12$:

Пункт назначения	Категория поезда	Время в пути	Вокзал
Пенза	скорый	14:28	Казанский
Пенза	фирменный	10:25	Казанский
Петрозаводск	скорый	13:27	Ленинградский
Петрозаводск	фирменный	12:55	Ленинградский
Полоцк	пассажирский	11:35	Белорусский
Псков	фирменный	13:00	Ленинградский
Пятигорск	скорый	34:38	Курский
Ростов-на-Дону	скорый	27:32	Курский
Ростов-на-Дону	скорый	24:28	Курский
Саратов	фирменный	15:05	Павелецкий
Саратов	фирменный	15:35	Павелецкий
Северодвинск	фирменный	23:27	Ярославский

Теперь в выборку дополнительно добавились две строки, и всего в выборке стало семь записей. Это слишком много, и при увеличении значения X такая тенденция к увеличению числа строк, входящих в выборку, сохранится. Значит, увеличивать далее значение X нет смысла.

Вывод: в результирующую выборку попадет ровно шесть записей при $X = 11$.

Ответ: 11.

13. Сколько значащих нулей содержит двоичная запись десятичного числа 127? В ответе укажите одно число — количество нулей.

Решение

1) Выполняем перевод числа 127 в двоичную систему счисления: $127_{10} = 1111111_2$.

2) Как можно видеть, никаких нулей в двоичной записи числа нет. Значит, количество незначащих нулей в записи этого числа равно нулю.

Ответ: 0.

Примечание. Незначащими нулями называют нули, которые в записи числа расположены правее, чем хотя бы одна единица. Например, в двоичном числе **1000110** — четыре незначащих нуля (они выделены красным цветом). Нули же, записанные слева от числа, не являются незначащими, поэтому, например, в двоичной записи числа 0010110 два первых слева нуля в подсчет незначащих нулей *не входят*.

14. У исполнителя Альфа две команды, которым присвоены номера:

1. прибавь 1

2. умножь на b

(b — неизвестное натуральное число, $b \geq 2$).

Выполняя первую из них, Альфа увеличивает число на экране на 1, а выполняя вторую, умножает это число на b .

Программа для исполнителя Альфа — это последовательность номеров команд.

Известно, что программа 11211 переводит число 6 в двузначное число.

Определите наибольшее возможное значение b .

Решение

1) Распишем программу для исполнителя в виде соответствующей последовательности команд и подставим известное нам исходное число:

6	1	прибавь 1
	1	прибавь 1
	2	умножь на b
	1	прибавь 1
	1	прибавь 1

2) Проведем вычисления по командам исполнителя, насколько это возможно, а далее записываем вычисления в виде выражений:

6	1	прибавь 1
7	1	прибавь 1
8	2	умножь на b
$8 \times b$	1	прибавь 1
$8 \times b + 1$	1	прибавь 1
$8 \times b + 2$		

3) Теперь нужно определить максимально возможное значение b для получения в результате при вычислении выражения $8 \cdot b + 2$ двузначного числа. Для этого можно составить неравенство:

$$8 \cdot b + 2 < 100.$$

Решаем его:

$$8 \cdot b + 2 < 100 \Rightarrow 8 \cdot b < 98 \Rightarrow b < 12,25.$$

Поскольку нас интересует наибольшее из возможных натуральных чисел, удовлетворяющих этому неравенству, выбираем $b = 12$.

4) Проверяем решение, подставляя в ранее составленную таблицу найденное значение b :

6	1	прибавь 1
7	1	прибавь 1
8	2	умножь на 12
96	1	прибавь 1
97	1	прибавь 1
98		

Полученное значение (98) удовлетворяет условию задачи. Если же выбрать значение b , равное 13, то получим:

6	1	прибавь 1
7	1	прибавь 1
8	2	умножь на 13
104	1	прибавь 1
105	1	прибавь 1
106		

т.е. получается уже трехзначное, а не двухзначное число.

Ответ: 12.

15. Файл размером 2000 Кбайт из пункта А в пункт Б передается за 30 секунд, а из пункта Б в пункт В передается за 20 секунд. Файл размером 3000 Кбайт, используя те же каналы связи, передали из пункта А в пункт В, при этом между окончанием приема данных в пункте Б и началом их передачи в пункт В произошла задержка. Общее время передачи данных из пункта А в пункт В составило 90 секунд. Определите время задержки в секундах. В ответе укажите только число, единицы измерения писать не нужно.

Решение

1) Зная объем первого файла и время его передачи, определим скорости передачи информации по используемым каналам связи:

$$\bullet v_{AB} = 2000 \text{ (кб)} / 30 \text{ (с)} = 2000 \cdot 2^{13} \text{ (бит)} / 30 \text{ (с)} = 125 \cdot 2^{17} / (5 \cdot 6) = 25 \cdot 2^{16} / 3 \text{ (бит/с)};$$

$$\bullet v_{BB} = 2000 \text{ (кб)} / 20 \text{ (с)} = 2000 \cdot 2^{13} \text{ (бит)} / 20 \text{ (с)} = 125 \cdot 2^{17} / (5 \cdot 4) = 25 \cdot 2^{15} \text{ (бит/с)}.$$

2) По вычисленным скоростям передачи данных определяем “чистое” время передачи второго файла:

$$\bullet t_{AB} = 3000 \text{ (кб)} / (25 \cdot 2^{16} / 3) \text{ (бит/с)} = 375 \cdot 2^{16} \text{ (бит)} / (25 \cdot 2^{16} / 3) \text{ (бит/с)} = 375 \cdot 2^{16} / (25 \cdot 2^{16}) \cdot 3 = (375 / 25) \cdot 3 = 45 \text{ (с)};$$

$$\bullet t_{BB} = 3000 \text{ (кб)} / 25 \cdot 2^{15} \text{ (бит/с)} = 375 \cdot 2^{16} \text{ (бит)} / (25 \cdot 2^{15}) \text{ (бит/с)} = 375 \cdot 2 / 25 = 30 \text{ (с)}.$$

3) Составляем уравнение, где t — искомое время задержки:

$$45 + t + 30 = 90.$$

$$\text{Отсюда } t = 90 - 75 = 15 \text{ (с)}.$$

Ответ: 15.

16. Алгоритм получает на вход цепочку из нулей и единиц, а затем строит из нее новую цепочку следующим образом. Если количество единиц в цепочке нечетно, то к цепочке слева приписывается единица, в противном случае справа приписывается ноль.

К двоичной записи числа семь указанный алгоритм применили три раза.

Укажите в ответе десятичное число, двоичной записью которого является полученная последовательность единиц и нулей.

Решение

1) Преобразуем заданное десятичное число семь в двоичную систему счисления:

$$7_{10} = 111_2.$$

2) Трижды применяем к полученной цепочке из трех единиц заданный алгоритм:

Шаг применения алгоритма	Было	Кол-во единиц	Стало
1	111	нечетно	1111
2	1111	четно	11110
3	11110	четно	111100

3) Полученное двоичное число 111100 преобразуем в десятичную систему счисления:

$$111100_2 = 60_{10}.$$

Ответ: 60.

17. В субботу Вася сказал Пете, что описание игр доступно в Интернете по адресу <https://obr.org/game.doc>. В понедельник Петя узнал, что Вася переместил описание в корневой каталог на сервере vasily2014.ru, доступ к которому осуществляется по протоколу ftp. Имя файла не изменилось. Укажите новый адрес нужного Пете файла.

Решение

1) Разберем исходный адрес файла в сети Интернет:

https://	obr.org	/game.doc
протокол	сервер	имя файла

2) Изменились:

- сервер — vasily2014.ru,
- протокол — ftp.

3) Составляем новый адрес:

fttp://	vasily2014.ru	/game.doc
протокол	сервер	имя файла

Тогда получаем новый адрес файла в виде:

<ftp://vasily2014.ru/game.doc>

Ответ: <ftp://vasily2014.ru/game.doc>

18. В языке запросов поискового сервера для обозначения логической операции “ИЛИ” используется символ “|”, а для логической операции “И” — символ “&”.

В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет:

Запрос	Найдено страниц (в тысячах)
США Китай	450
Китай	260
США & Китай	50

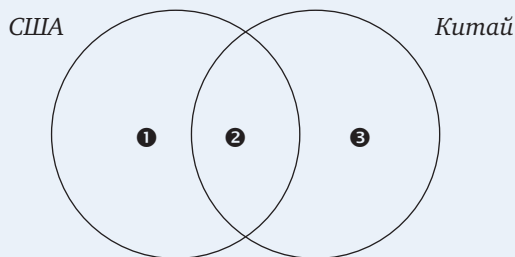
Какое количество страниц (в тысячах) будет найдено по запросу:

США?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Решение

Задача, традиционно решаемая при помощи кругов Эйлера. Это, пожалуй, единственная задача, содержание которой не изменилось.



1) Расписываем запросы:

- *США* | *Китай* — $1 + 2 + 3 = 450$,
- *Китай* — $2 + 3 = 260$,
- *США* & *Китай* — $2 = 50$,
- *США* (искомый запрос) — $1 + 2$.

2) Решаем полученную систему уравнений:

$$2 = 50,$$

↓

$$2 + 3 = 260, \Rightarrow 3 = 260 - 50 = 210,$$

$$2 + 2 + 3 = 450, \Rightarrow 1 = 450 - 210 - 50 = 190.$$

3) Тогда искомого запросу *США* соответствует количество страниц, равное:

$$1 + 2 = 190 + 50 = 240.$$

Ответ: 240.

19. В программе описан одномерный целочисленный массив с индексами от 1 до 10. Ниже представлен фрагмент программы, обрабатывающей данный массив:

```
s := 0;
n := 10;
for i := 1 to n - 1 do begin
  k := 10 - i;
  s := s + A[k + 1] - A[k];
end;
```

В начале выполнения этого фрагмента в массиве находились двузначные натуральные числа, ни одно из которых не делится на 5. Какое наибольшее

значение может иметь переменная s после выполнения данной программы?

Решение

1) Анализируем алгоритм:

- переменная s предназначена для подсчета суммы/разности, изначально она обнуляется;
- в цикле выполняется сложение прежнего значения s с разностью последующего элемента массива и текущего.

Такое вычисление значения s можно расписать в одну строку в виде:

$$s = (A[2] - A[1]) + (A[3] - A[2]) + (A[4] - A[3]) + \\ + (A[5] - A[4]) + (A[6] - A[5]) + (A[7] - A[6]) + \\ + (A[8] - A[7]) + (A[9] - A[8]) + (A[10] - A[9]).$$

Раскроем скобки и перегруппируем слагаемые:

$$s = -A[1] + (A[2] - A[2]) + (A[3] - A[3]) + \\ + (A[4] - A[4]) + (A[5] - A[5]) + (A[6] - \\ - A[6]) + (A[7] - A[7]) + (A[8] - A[8]) + \\ + (A[9] - A[9]) + A[10].$$

Как видим, большинство слагаемых сокращается, и получаем:

$$s = A[10] - A[1].$$

2) При этом требуется, чтобы итоговое значение s было максимальным. Когда это возможно, учитывая, что все числа — двузначные, натуральные и не делящиеся на пять? Очевидно, уменьшаемое ($A[10]$) должно быть максимально возможным таким числом, а вычитаемое ($A[1]$) — минимально возможным.

3) Максимально возможное двузначное, натуральное, не делящееся на пять число: 99.

Минимально возможное двузначное, натуральное, не делящееся на пять число: 11 (число 10 не подходит, так как оно делится на пять).

Тогда значение s (максимально возможное, требуемое в задаче) равно $99 - 11 = 88$.

Ответ: 88.



СПРАВОЧНИК ЕГЭ

Таблицы степеней чисел 2 и 3

n	0	1	2	3	4	5	6	7	8	9	10
2^n	2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8	2^9	2^{10}
Значение	1	2	4	8	16	32	64	128	256	516	1024

n	0	1	2	3	4	5	6	7	8	9	10
3^n	3^0	3^1	3^2	3^3	3^4	3^5	3^6	3^7	3^8	3^9	3^{10}
Значение	1	3	9	27	81	243	729	2187	6561	19 683	59 049



Обработка графической информации

Растровая графика

► Ключевые слова:

- растровое изображение
- пиксель
- форматы файлов
- оцифровка изображений
- графический редактор
- панель инструментов
- палитра
- основной цвет
- фоновый цвет
- масштабирование рисунка
- текстовые надписи

Как хранятся изображения?

Вы уже знаете, что растровые изображения состоят из отдельных точек (пикселей), каждая из которых закрашена своим цветом. Информация о цвете пикселей хранится в памяти

компьютера в виде двоичных кодов — каждому цвету присвоен свой код. Пусть у нас есть черно-белый рисунок размером 4×4 пикселя:

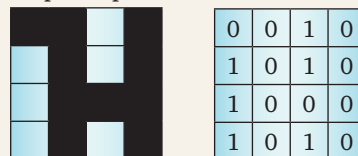


Рис. 1

Черному цвету присвоим код 0, а белому — код 1. Таблица на рис. 1 показывает коды всех пикселей. Когда рисунок сохраняется в файле, эти коды записываются построчно: сначала первая строка таблицы (коды пикселей верхнего ряда), потом — вторая и т.д. В нашем случае получим 0010101010001010

Достаточно ли этих данных, чтобы вывести рисунок на экран? Скорее всего нет. Даже если мы знаем, что это черно-белый рисунок, у нас есть только 16 кодов, определяющих цвет 16 пикселей. При этом невозможно определить размеры рисунка, тут есть несколько вариантов: 1×16 , 2×8 , 4×4 , 8×2

К.Ю. Поляков,

д. т. н., Санкт-Петербург,
<http://kpolyakov.spb.ru>,

Е.А. Еремин,

к. ф.-м. н., г. Пермь

или 16×1 . Поэтому в самом начале файла находится *заголовок*, в котором записана вся необходимая информация для вывода рисунка на экран: размеры рисунка, метод кодирования цвета¹, алгоритм сжатия.

Существует очень много различных форматов файлов для хранения изображений. Чаще всего мы встречаемся с такими форматами:

- формат BMP (файлы с расширением **.bmp**) — стандартный формат растровых рисунков в операционной системе *Windows*; чаще всего не использует сжатие, поэтому рисунки занимают много места;
- формат JPEG (файлы с расширениями **.jpg** и **.jpeg**) был разработан специально для хранения фотографий; в нем используется сжатие с потерями; степень сжатия (и качество рисунка) можно задать при сохранении файла;
- формат GIF (файлы с расширением **.gif**) предназначен для рисунков с количеством цветов не более 256; использует сжатие без потерь; пиксели могут быть прозрачными; в таком формате можно хранить анимацию (короткие видеоролики);
- формат PNG (файлы с расширением **.png**) — универсальный формат, использующий сжатие без потерь; пиксели могут быть частично прозрачными (это проявится, например, при наложении такого рисунка на фон страницы сайта в Интернете).

Файлы, в которых хранятся растровые рисунки, обычно имеют значительный размер. Он зависит от количества пикселей и количества используемых цветов.

Как получить растровое изображение?

Итак, растровое изображение — это точечный рисунок, сохраненный в цифровом формате. Существует два основных способа получения растровых изображений:

- ввод с помощью какого-либо устройства, например, сканера, цифрового фотоаппарата или веб-камеры; напомним, что при сканировании

¹ Подробнее о кодировании цветов вы узнаете в 8-м классе.

происходит преобразование “некомпьютерной” информации в компьютерные данные (*оцифровка*);

- создание рисунка с помощью какой-либо программы.

Графический редактор

Для создания и редактирования рисунков разработаны специальные программы — *графические редакторы*. Они умеют загружать рисунок в оперативную память, выводить его на экран, изменять коды цветов по командам пользователя и сохранять полученное изображение в файле.

Все растровые графические редакторы (то есть редакторы, предназначенные для обработки растровых рисунков) можно условно разделить на две группы — простейшие (*Paint* в системе *Windows*, *KolourPaint* в *Linux*) и профессиональные (*Adobe Photoshop*, *GIMP*). В этой главе мы познакомимся с простейшими редакторами на примере редактора *Paint* для *Windows 8*. Тем не менее все изученные вами принципы будут работать и в других редакторах этого класса, например, в *KolourPaint*.

Посмотрим, какие элементы интерфейса есть в окне редактора *Paint* (рис. 2). В верхней части окна расположена “лента” с тремя вкладками. Вкладка *Файл* отвечает за работу с файлами на диске. На вкладке *Главная* расположены кнопки для работы с буфером обмена, инструменты для редактирования рисунков и цветовая палитра. Справа мы видим линейку прокрутки, это означает, что не весь рисунок по высоте помещается в окно. Вкладка *Вид* позволяет изменить масштаб просмотра (увеличить или уменьшить изображение на экране без изменения количества пикселей), вывести линейки (над рисунком и слева от него).

Снизу находится строка состояния: в ней программа показывает координаты указателя мыши (считая от верхнего левого угла рисунка), размеры рисунка в пикселях и объем файла. В правой части строки состояния с помощью движка можно изменить масштаб просмотра.

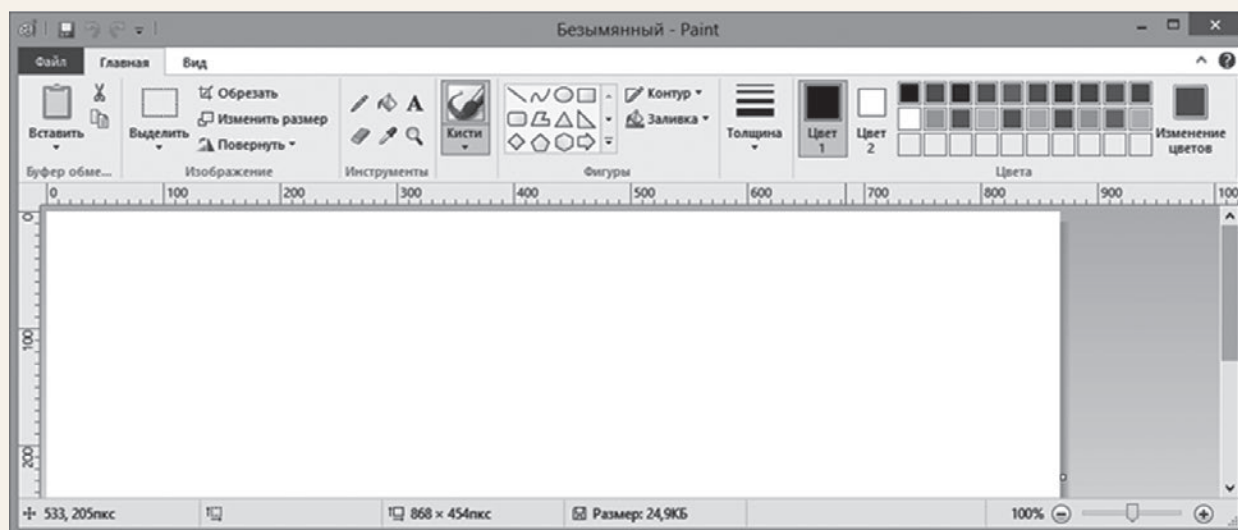



Рис. 2

Рисование от руки


Простейший способ рисования — использование карандаша или кисти. Если включить инструмент *Карандаш* , то при нажатой левой кнопке мыши можно рисовать линии на поле рисунка.


Меню *Толщина* позволяет (для всех инструментов) выбрать толщину линии.


Цвет линии будет совпадать с цветом *Цвет 1* на панели *Главная*. Этот цвет еще называют основным, или *цветом переднего плана*. Если для рисования использовать не левую кнопку мыши, а правую, линия будет иметь *Цвет 2 (фоновый цвет)*.

Для того чтобы изменить основной или фоновый цвет, нужно щелчком мыши выбрать *Цвет 1* или *Цвет 2*, а затем щелкнуть по нужному цвету в палитре. Кнопка *Изменение цветов* позволяет добавить новые цвета в нижний ряд палитры.


Рисовать можно не только карандашом, но и различными кистями (меню *Кисть*), которые имитируют кисти художника и распылитель краски.

Чтобы стереть какие-то части рисунка, используют инструмент *Ластик* . При нажатой левой кнопке мыши ластик стирает (закрашивает цветом фона) все пиксели, через которые он проходит. При нажатой правой кнопке мыши стираются только те пиксели, которые закрашены основным цветом.

С помощью инструмента *Заливка*  можно залить области одного цвета основным цветом (щелчком левой кнопки мыши) или фоновым цветом (щелчком правой кнопки мыши). Если контур, ограничивающий нужную область, имеет хотя бы небольшой разрыв, краска “вытекает” и может залить ту часть, которую закрашивать не нужно.

Инструмент *Масштаб*  позволяет увеличить нужную часть рисунка. Выполнив увеличение несколько раз, вы увидите, что изображение состоит из пикселей.

Текст

На рисунок можно добавить надписи с помощью инструмента *Текст* . Выбрав этот инструмент, нужно щелкнуть на поле рисунка, и в этой точке появляется рамка для ввода текста. Размеры рамки можно изменять, перетаскивая маркеры в углах и на серединах сторон рамки. Саму рамку можно перетаскивать мышью за границу.

На ленте открывается дополнительная вкладка *Текст*, на которой можно выбрать нужный шрифт и установить прозрачный или непрозрачный фон:

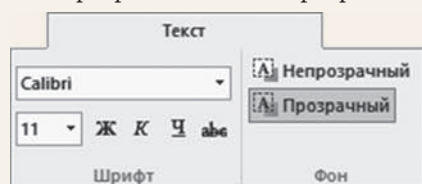


Рис. 3

Внутри рамки можно вводить и редактировать текст, как в обычном текстовом редакторе. Но как только редактирование закончено (вы щелкнули

мышью вне рамки), текст превращается в набор пикселей, и теперь его можно редактировать только как растровый рисунок. Например, становится очень непросто заменить один символ на другой или удалить ненужный символ.

Контрольные вопросы

1. В каком коде хранятся изображения в компьютере?
2. Что такое заголовок файла? Зачем он нужен?
3. Какие форматы графических файлов вы знаете? Чем они отличаются?
4. От чего зависит размер файла, в котором хранится растровый рисунок?
5. Что такое графический редактор?
6. Что такое основной цвет и фоновый цвет? Зачем нужны два рабочих цвета?
7. Как изменить рабочие цвета?
8. Как выполняется заливка?
9. В каких случаях может быть залито все поле рисунка? Как это исправить?
10. Что такое масштабирование? Изменяются ли при масштабировании размеры рисунка в пикселях?
11. Как можно редактировать текст, который уже зафиксирован в растровом рисунке?

Темы сообщений:

- а) Растровые графические редакторы.
- б) Графический редактор Paint.NET.
- в) Графический редактор GIMP.
- г) Графические онлайн-редакторы.
- д) Форматы растровых рисунков.
- е) Что такое разрешение?
- ж) Как кодируется цвет?

Интересные сайты:

- а) www.getpaint.net — графический редактор Paint.NET;
- б) www.gimp.org — графический редактор GIMP;
- в) www.progimp.ru — уроки по редактору GIMP;
- г) www.gimpart.org — уроки по редактору GIMP;
- д) psd.ru — уроки по редактору Photoshop;
- е) apps.pixlr.com/editor/ — онлайн-редактор растровой графики.


Геометрические фигуры

Ключевые слова:


- линия (отрезок)
- прямоугольник
- овал (эллипс)
- многоугольник
- контур
- заливка

Рисовать все от руки — весьма непростое занятие, ровную линию получить практически невозможно. Поэтому в любом графическом редакторе есть инструменты для рисования геометрических фигур — линий, прямоугольников, окружностей, ломаных.

Линия

Чтобы нарисовать отрезок, нужно включить инструмент *Линия* , нажать кнопку мыши в начальной точке отрезка, переместить мышь (при нажатой кнопке) в конечную точку и там отпустить кнопку. Если использовать левую кнопку мыши, отрезок будет иметь основной цвет, а при работе правой кнопкой — фоновый цвет.

Прямоугольник

Инструмент *Прямоугольник*  служит для рисования прямоугольников и квадратов. Чтобы нарисовать прямоугольник, нужно нажать кнопку мыши в одном из углов будущего прямоугольника, провести мышь в противоположный угол и там отпустить нажатую кнопку. Если удерживать нажатой клавишу **Shift**, то получится квадрат, то есть программа следит за тем, чтобы высота и ширина прямоугольника были одинаковы. Размеры прямоугольника выводятся в строке состояния (до того, как вы отпустите кнопку мыши и зафиксируете прямоугольник на рисунке).

Прямоугольник — это замкнутая фигура, поэтому, кроме контура, для нее можно использовать заливку внутренней области. При работе левой кнопкой мыши контур прямоугольника будет основного цвета, а заливка — фонового цвета; при работе правой кнопкой мыши цвета меняются местами.

С помощью выпадающих меню *Контур* и *Заливка* можно выбрать нужные свойства контура и заливки, а также отключить контур или заливку вообще:

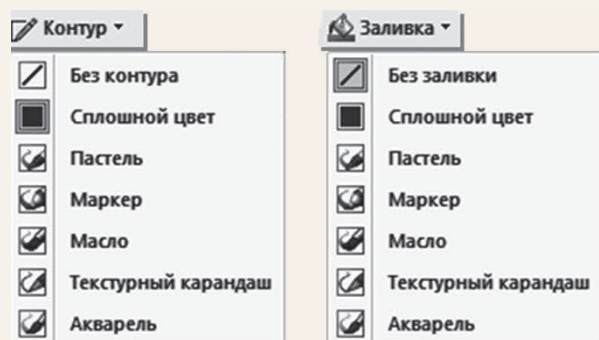



Рис. 4

Овал (эллипс)

Инструмент *Овал*  (во многих версиях программ он называется *Эллипс*) служит для рисования овалов и окружностей. Рисование выполняется не от центра фигуры, а от угла прямоугольника, в который она вписана, то есть от одной из точек (1, 2, 3 или 4):

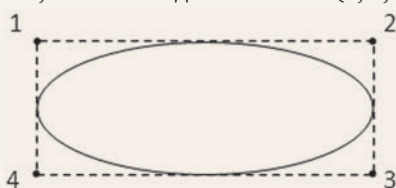



Рис. 5

Чтобы определить размеры овала, нужно протащить указатель мыши из любого угла описывающего прямоугольника в противоположный угол при нажатой левой кнопке. Так же, как и при ри-

совании прямоугольников, внутренняя часть фигуры будет залита фоновым цветом, если в меню *Заливка* включен нужный режим. При работе правой кнопкой мыши цвета меняются местами.

При нажатой клавише **Shift** программа будет рисовать круг (или окружность), автоматически выдерживая одинаковые размеры фигуры по вертикали и горизонтали.

Многоугольник

Инструмент *Многоугольник*  позволяет строить многоугольники (замкнутые ломаные линии). Первая сторона рисуется так же, как отрезок, а все следующие вершины устанавливаются щелчками мыши. Двойной щелчок заканчивает рисование и замыкает ломаную линию в многоугольник.

Многоугольник — это замкнутая фигура. Чтобы его внутренняя часть была залита фоновым цветом, нужно выбрать соответствующий режим в меню *Заливка*.

Контрольные вопросы

1. Как установить толщину отрезка?
2. Чем отличается рисование фигур с помощью левой и правой кнопок мыши?
3. Как нарисовать только контур фигуры (без заливки)?
4. Как нарисовать фигуру без контура?
5. Как нарисовать круг, если вы знаете его центр и радиус?
6. Почему нарисованную фигуру сложно переместить после освобождения кнопки мыши?

Темы сообщений:

- а) Геометрические фигуры в редакторе Paint.NET;
- б) Геометрические фигуры в редакторе GIMP.

Работа с фрагментами

Ключевые слова:

- фрагмент
- выделение области
- прямоугольная область
- область произвольной формы

Выделение области

С помощью инструментов группы “Выделение” можно выделить некоторую область рисунка (*фрагмент*) для того, чтобы затем выполнить с ней какие-то операции: переместить, скопировать, повернуть, изменить размеры.

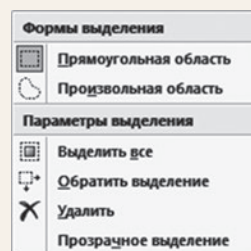


Рис. 6

Существует два основных способа — выделение прямоугольной области (она строится так же, как и прямоугольник) и выделение произвольной области (нужно обвести мышью нужную область при нажатой левой кнопке).

Если установить флажок “Прозрачное выделение”, то пиксели, закрасненные цветом фона, не будут выделены.

Можно выделить весь рисунок, нажав на клавиши **Ctrl** + **A**.

Выделенная область обозначается штриховой рамкой. Эта линия временно показывается только на экране, она не меняет пиксели рисунка в памяти компьютера и бесследно исчезает, если отменить выделение (щелкнув мышью вне рамки).

Перемещение и копирование

Выделенную область (фрагмент) можно перетаскивать мышью, при этом освободившееся место заливается цветом фона. Если при перетаскивании удерживать клавишу **Ctrl**, фрагмент будет скопирован.

При нажатии клавиши **Ctrl** фрагмент удаляется и его место заливается цветом фона.

Чтобы изменить размер фрагмента, нужно перетащить мышью маркеры на рамке выделенной области.

Для того чтобы зафиксировать фрагмент, нужно щелкнуть мышью вне его рамки. При этом часть изображения, накрытая этим фрагментом, пропадает. Если снова выделить тот же фрагмент и передвинуть его, мы увидим только фоновый цвет. Это связано с тем, что наш графический редактор работает только с *однослойными* (плоскими) рисунками, в которых все пиксели находятся на “одной высоте”. Профессиональные растровые редакторы (*Photoshop*, *GIMP*) умеют работать с многослойными изображениями: слои располагаются друг над другом, и при изменении пикселей одного слоя соответствующие пиксели остальных слоев не меняются. Такой подход позволяет легко создавать сложные рисунки из частей, не стирая пиксели, которые могут потом понадобиться.

Преобразования

Кнопка **Повернуть** открывает меню, позволяющее поворачивать фрагмент на углы 90, 180 и 270 градусов, а также строить его отражение по горизонтали и вертикали. Эти команды также доступны из *контекстного меню*, которое появляется при щелчке правой кнопкой мыши на выделенной области.

Сохранение выделенной области

Многие графические редакторы позволяют сохранять выделенную область в отдельном файле и вставлять в текущий рисунок фрагмент с диска. Обычно это делается через контекстное меню,

которое содержит команды *Копировать в файл* и *Вставить из файла*.

Контрольные вопросы

1. Какие способы выделения областей вы знаете?
2. Как выделить объект, не затрагивая фон?
3. Какие сложности могут возникнуть при перемещении фрагментов сложных рисунков? Чем они вызваны?
4. Объясните различие между однослойными и многослойными документами.
5. Как повернуть фрагмент?

Темы сообщений:

- a) Выделение областей в редакторе Paint.NET;
- б) Выделение областей в редакторе GIMP;
- в) Работа со слоями в редакторе Paint.NET;
- г) Работа со слоями в редакторе GIMP.

Вставка рисунков в документ

Ключевые слова:

- вставка рисунка
- изменение размеров рисунка
- обтекание текстом
- буфер обмена

Вставка рисунка с диска

Текстовые процессоры позволяют объединять в одном документе различные виды информации: текст, таблицы, рисунки, диаграммы. Для того чтобы добавить рисунок в документ *Word*, нужно щелкнуть по кнопке *Рисунок* на вкладке *Вставка* и выбрать файл на диске.

Рисунок сначала вставляется в текст как “большая буква”, раздвигая строки:

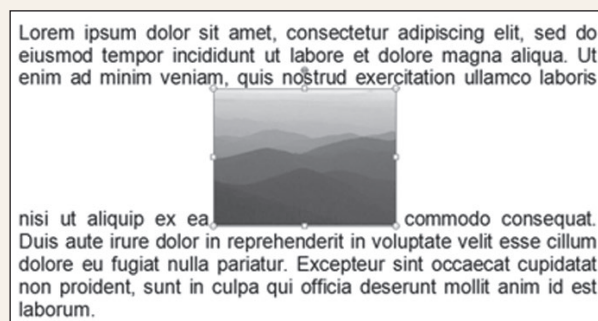


Рис. 7

Его можно перетаскивать мышкой в любое место. Размеры рисунка можно изменять, перетаскивая маркеры на его границе. Такой режим используется чаще всего для вставки небольших рисунков (например, изображения кнопки) прямо в текст.

Для того чтобы большой рисунок обтекался текстом (рис. 8), нужно установить для него режим обтекания *Вокруг рамки*. Для этого служит раздел *Обтекание текстом* в контекстном меню рисунка. Обычно рисунок выравнивается так, чтобы сохранить вертикаль. Например, на рис. 8 левая граница рисунка находится на одной вертикали с границей текста.

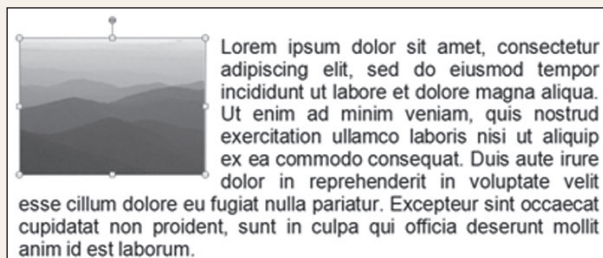


Рис. 8

Вставка рисунка через буфер обмена

Напомним, что буфер обмена — это область оперативной памяти, через которую программы могут обмениваться данными. Предположим, что вы построили рисунок в графическом редакторе и хотите вставить его в документ без сохранения на диске. Для этого нужно скопировать рисунок в буфер обмена (клавиши **Ctrl** + **C**), перейти в окно текстового документа и вставить содержимое буфера обмена (**Ctrl** + **V**). Рисунок будет помещен в то место, где был курсор.

Контрольные вопросы

1. Какие режимы обтекания рисунков вы знаете? Когда они используются?

2. Как вставить рисунок через буфер обмена?

Темы сообщений:

а) Что такое *Lorem ipsum*?

Векторная графика

Ключевые слова:

- векторный рисунок
- векторный редактор
- примитивы
- выделение объектов
- изменение порядка объектов
- выравнивание
- распределение
- кривая Безье
- гладкий узел
- угловой узел

Что хранится в памяти?

Главный недостаток растровой графики вы уже увидели — при изменении размеров рисунка будут искажения: изменяется и форма объектов, и их цвет. Все дело в том, что программа воспринимает рисунок не как набор объектов, а как множество точек разного цвета. Человек же, глядя на рисунок,

представляет нарисованные объекты у себя в сознании.

Растровая графика неприменима там, где нужно масштабировать рисунки, то есть изменять их размеры без потери качества. Например, плакат должен хорошо выглядеть как на листе формата А4, так и на баннере, размеры которого измеряются в метрах.

Сейчас инженеры разрабатывают новые автомобили, самолеты, приборы в системах автоматизированного проектирования. В них строится трехмерная (объемная) модель объекта, которую затем нужно просматривать с разных сторон и рассчитывать на прочность. В таких задачах растровая графика вовсе неприменима.

Поэтому появилась другая идея: хранить в памяти компьютера не отдельные пиксели, а информацию о *геометрических фигурах*, из которых составлен рисунок:

- толщину, цвет и стиль контура;
- стиль заливки (один цвет, переход между несколькими цветами, узор);
- координаты фигуры, угол поворота, угол наклона.

Такая графика называется *векторной*.

Векторный рисунок хранится в памяти как множество геометрических фигур с заданными свойствами контура и заливки внутренней области.

Векторный рисунок можно “разобрать” на части, раставив мышкой его элементы, а потом снова собрать полное изображение:

Заметим, что все данные о фигурах, составляющих векторный рисунок, записаны в память компьютера в виде двоичных кодов.

Для создания и редактирования векторных рисунков используют программы, которые называют *векторными редакторами*. Самые известные профессиональные векторные редакторы — *Adobe Illustrator* и *Corel Draw* — коммерческие программы, которые стоят достаточно дорого. Вместо них можно использовать бесплатную программу *Inkscape* (inkscape.org), которая относится к свободному программному обеспечению. Большими возможностями в этой области обладают и офисные пакеты *Microsoft Office* и *OpenOffice.org*, которые мы и будем использовать для выполнения практических работ.

Для работы с векторными рисунками нужно выбрать какой-то набор стандартных фигур (их



Рис. 9

называют *примитивы* — простейшие элементы), из которых будут строиться эти рисунки. Это всем знакомые геометрические фигуры, например, отрезки, прямоугольники, многоугольники, эллипсы и т.п. Многие *векторные редакторы* содержат свой набор дополнительных фигур (треугольники, стрелки, звездочки, выноски и др.).

Векторные рисунки могут быть сохранены в различных форматах, в том числе:

- WMF, EMF (файлы с расширениями **.wmf**, **.emf**) — стандартные форматы векторных рисунков в *Windows*; в таких форматах хранятся рисунки в коллекции *Microsoft Office*;
- ODG (файлы с расширениями **.odg**) — формат векторных рисунков пакета *OpenOffice*;
- AI (файлы с расширениями **.ai**) — формат программы *Adobe Illustrator*;
- CDR (файлы с расширениями **.cdr**) — формат программы *Corel Draw*;
- EPS (файлы с расширениями **.eps**) — формат для хранения как растровых, так и векторных изображений и их комбинаций, используется в полиграфии;
- SVG (файлы с расширениями **.svg**) — формат векторных рисунков, которые можно встраивать в веб-страницы.

Объем файлов, в которых хранятся векторные рисунки, обычно очень небольшой. Он зависит от сложности рисунка: количества фигур и их свойств.

Примитивы

Векторный рисунок состоит из *примитивов* — элементарных фигур. К ним относятся отрезок, прямоугольник, эллипс, кривая, ломаная и др. На *рис. 10* показаны некоторые часто используемые фигуры в редакторе *Word*.



Рис. 10

Основные фигуры — отрезок, прямоугольник, эллипс — рисуются так же, как и в растровых редакторах, однако после завершения рисования они не превращаются в набор пикселей, а остаются отдельными объектами, каждый из которых можно редактировать (перемещать, изменять свойства и даже удалить) независимо от других.

Чтобы что-то сделать с объектом, его нужно выделить щелчком мыши. Для того чтобы выделить сразу несколько объектов, по второму и всем следующим нужно щелкать при нажатой клавише **Shift**.

На рамке выделенного объекта появляются маркеры, перетаскив которые можно изменить размеры фигуры. Оба конца отрезка тоже можно перетаскивать мышью.

Чтобы переместить сам объект, нужно схватить его за контур. Если при перетаскивании удержи-

вать клавишу **Ctrl**, фигура будет скопирована в новое место.

Нажатие на клавишу **Delete** удаляет выделенные объекты.

Для контура любой фигуры можно задать цвет, толщину, стиль линии (сплошная, штриховая, точечная и др.). На концах отрезка могут быть стрелки различной формы и размеров. В редакторе *Word* для настройки контура используется меню *Контур фигуры* на вкладке *Формат*, а в *OpenOffice.org Writer* — панель *Свойства*.

С помощью меню *Заливка фигуры* (для *Word* — на вкладке *Формат*) можно выбрать цвет заливки (или отключить заливку вообще). Кроме того, возможны и другие типы заливки:

- *градиент* — плавный переход между двумя или несколькими цветами;
- *рисунок* (для заполнения области используется изображение из файла);
- *текстура* — рисунок, имитирующий материал (бумага, холст, дерево и др.);
- *узор* — рисунок, составленный из двух цветов.

Изменение порядка элементов

Новые фигуры добавляются на рисунок поверх существующих. Иногда нужно изменить порядок их расположения, например, переместить какую-то фигуру впереди (выше) всех остальных объектов. Для этого в *Word* служат команды вкладки *Формат*:

- *На передний план* (поверх всех остальных объектов);
- *Переместить вперед* — поменять местами выделенный объект с тем, который находится непосредственно над ним;
- *На задний план* (позади всех остальных объектов);
- *Переместить назад* — поменять местами выделенный объект с тем, который находится непосредственно под ним.

Так можно перемещать (по уровням) сразу несколько выделенных объектов. Все эти команды доступны из контекстного меню.

На *рис. 11* показан результат применения команды *«На передний план»* к объекту с белой заливкой.

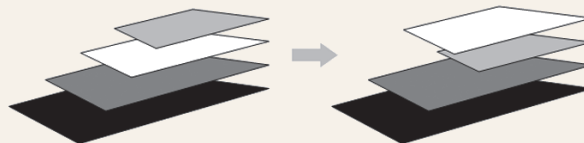


Рис. 11

Выравнивание, распределение

При работе с векторными рисунками часто возникают такие задачи:

- выровнять несколько объектов по какому-то краю (например, по верхней границе):

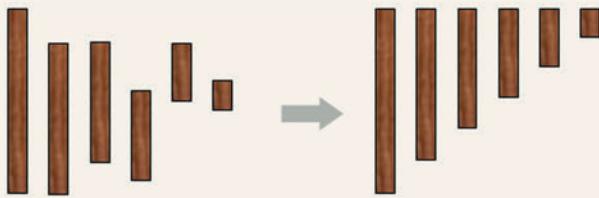


Рис. 12

- распределить несколько объектов так, чтобы они оказались на равных расстояниях по горизонтали или по вертикали:

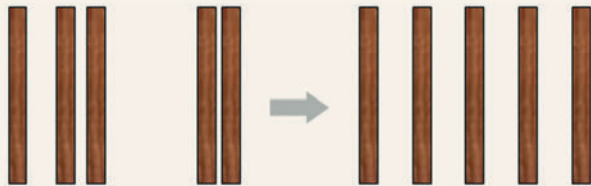


Рис. 13

Конечно, все можно сделать вручную, передвигая мышью отдельно каждую фигуру, но это очень трудоемко и неточно. В текстовых процессорах эти операции автоматизированы. В программе *Microsoft Word* они выполняются с помощью команд группы *Выровнять* на вкладке *Формат*, а в *OpenOffice.org Writer* — с помощью раздела *Выравнивание* в контекстном меню.

Группировка

Предположим, что ваш векторный рисунок строится из сложных объектов (например, дом, автомобиль, собака, человек), каждый из которых, в свою очередь, тоже состоит из частей.

Для того чтобы передвинуть такой сложный объект, нужно переместить все его части, не сдвинув их друг относительно друга. Лучше всего заранее *сгруппировать* все эти части (например, все элементы дома) в единый объект с помощью команды *Сгруппировать* из контекстного меню. После этого весь объект выделяется одним щелчком мыши и перетаскивается за один раз.

Сгруппированный объект имеет общую рамку, поэтому очень легко изменять его размеры с помощью маркеров этой рамки (рис. 14). Сделать это без группировки крайне сложно.

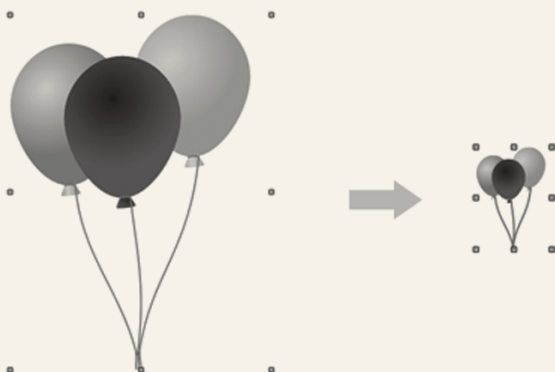


Рис. 14

Для того чтобы отменить группировку и разделить сложный объект на составные части, нужно использовать команду *Разгруппировать* из контекстного меню.

Векторные кривые

К сожалению, не все рисунки можно построить из готовых примитивов (отрезков, прямоугольников, овалов, ломаных). Большинство нужных элементов, из которых состоят рисунки профессиональных дизайнеров, — это *кривые*. В памяти компьютера хранятся координаты опорных точек (узлов), а сама форма кривой между узлами описывается математической формулой.

На рис. 15 изображена линия с опорными точками А, Б, В, Г и Д.

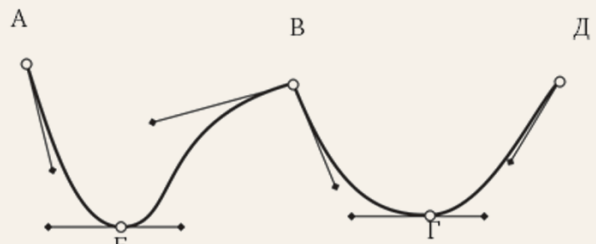






Рис. 15

У каждой из этих точек есть “рычаги” (*управляющие линии*), перемещая концы этих рычагов можно регулировать кривизну всех участков кривой. Если оба рычага находятся на одной прямой, получается гладкий узел (Б и Г), если нет — то угловой узел (В). Таким образом, форма кривой полностью задается координатами опорных точек и координатами рычагов. Кривые, заданные таким образом, называют *кривыми Безье* в честь их изобретателя французского инженера Пьера Безье.

В редакторе *Word* кривую можно нарисовать с помощью инструментов *Кривая*  и *Полилиния* . Для того чтобы редактировать кривую, нужно щелкнуть правой кнопкой мыши по кривой и выбрать команду *Начать изменение узлов* в контекстном меню. При этом появляется возможность

- перемещать, удалять и добавлять узлы;
- изменять тип узла (с гладкого на угловой и обратно);
- изменять положение рычагов, регулирующих кривизну.

В *OpenOffice.org Writer* для рисования кривых служит инструмент *Объект кривых*  на панели *Рисование*. Кнопка *Изменение геометрии*  на этой же панели включает режим редактирования кривой.

Контрольные вопросы

1. В чем отличие векторной графики от растровой?
2. Расскажите о преимуществах и недостатках векторных рисунков.

3. Можно ли сохранить фотографию в векторном формате?

4. Как вы думаете, в каких задачах лучше использовать растровую графику, а в каких — векторную:

- создание коллажей;
- проектирование зданий;
- составление плана помещений;
- подготовка обложки журнала мод;
- подготовка чертежей и схем;
- спутниковая съемка поверхности Земли;
- разработка географических карт.

5. Какая информация сохраняется в файле для векторного рисунка?

6. От чего зависит размер файла, в котором хранится векторный рисунок?

7. Почему векторные рисунки не искажаются при изменении размеров?

8. Что такое примитивы? Зачем они нужны?

9. Какие способы заливки можно использовать в векторных рисунках?

10. Как выделить сразу несколько объектов на рисунке?

11. Как изменять порядок расположения элементов рисунка? Зачем это нужно?

12. Предложите другие способы выполнения операции, показанной на рис. 11.

13. Как выровнять объекты по нижней границе?

14. Как распределить объекты на равном расстоянии по вертикали?

15. Зачем нужна группировка?

16. Как переместить сложный объект?

17. Как изменить размеры сложного объекта?

18. Как разделить сгруппированный объект на отдельные элементы?

19. Зачем нужны кривые?

20. Какая информация о векторных кривых сохраняется в файле?

21. Какие два основных типа узлов вы знаете?

22. Что можно сделать с кривой во время редактирования?

Темы сообщений:

а) Растровые и векторные рисунки: сравнение;

б) Векторная графика вокруг нас;

в) Векторные графические редакторы;

г) Форматы векторных рисунков;

д) Векторный редактор *OpenOffice.org Draw*;

е) Векторный редактор *Inkscape*;

ж) Векторная графика в редакторе GIMP;

з) Векторная графика на веб-страницах.

Интересные сайты:

а) www.inkscape.org — редактор *Inkscape*;

б) inkscape.paint-net.ru — уроки по редактору *Inkscape*;

в) wiki.linuxformat.ru/wiki/LXF74-75:Inkscape — уроки по редактору *Inkscape*;

г) corel.demiart.ru — уроки по редактору *Corel Draw*;

д) coreldrawgromov.ru — уроки по редактору *Corel Draw*.

Выводы из прочитанного:

- Изображения хранятся в памяти компьютера в виде двоичных кодов.
- Существует два типа цифровых изображений: растровые и векторные.
- Растровое (точечное) изображение состоит из отдельных *пикселей*, каждый из которых имеет свой цвет. Для создания и обработки растровых изображений служат графические редакторы.
 - Растровые изображения вводятся в компьютер (*оцифровываются*) с помощью сканера, цифрового фотоаппарата или веб-камеры.
 - Растровые редакторы позволяют рисовать линии, отрезки, прямоугольники, овалы и другие фигуры, делать текстовые надписи. После того как фигура или текст зафиксированы в рисунке, они превращаются в набор пикселей.
 - При рисовании используются два цвета: основной и фоновый. Эти цвета можно выбрать из палитры.
 - В растровых редакторах можно выделить *фрагмент* (часть рисунка) прямоугольной или произвольной формы. Фрагмент можно переместить, скопировать, удалить, изменить его размеры, сохранить в виде файла.
 - В текстовые документы можно вставлять рисунки из файлов или через буфер обмена. Режим обтекания рисунка определяет, как он будет расположен по отношению к окружающему тексту.
- Векторный рисунок хранится в памяти как множество простейших геометрических фигур (графических примитивов) с заданными свойствами контура и заливки внутренней области.
 - Качество векторных изображений не снижается при изменении их размеров.
 - Элементы векторного рисунка — это независимые объекты, каждый из которых можно редактировать отдельно от других.
 - Для каждой векторной фигуры определяется контур (цвет, толщина, стиль линии, стрелки) и заливка (сплошной цвет, градиент, текстура или растровый рисунок).
 - Элементы векторного рисунка можно перемещать (как в плоскости рисунка, так и по уровням) друг относительно друга, выравнивать, распределять равномерно по горизонтали и вертикали.
 - Для удобства работы с составными объектами их можно сгруппировать.
 - Векторные кривые хранятся в памяти как набор узлов. Узлы могут быть гладкие и угловые. Форма кривой между узлами рассчитывается по математическим формулам, коэффициенты которых определяются расположением узлов.



Шесть вопросов по информатике для конкурсов “Что? Где? Когда?” и “Брейн-ринг”

Д.М. Златопольский,
Москва

1. Изображение этого уже неиспользуемого сейчас носителя информации используется в качестве значка для кнопки. О каком носителе идет речь?

2. Аббревиатура известного носителя информации — DVD — изначально расшифровывалась, да иногда расшифровывается и сейчас, как “Digital Video Disc” (цифровой видеодиск). Что в таком варианте, так сказать, “расшифровки” неверно?

Вариант вопроса

Аббревиатура известного носителя информации — DVD — изначально расшифровывалась, да иногда расшифровывается и сейчас, как “Digital Video Disc” (цифровой видеодиск), поскольку данный формат первоначально разрабатывался как замена видеокассетам. Что в таком варианте, так сказать, “расшифровки” неверно?

3. Профессиональный праздник “День программиста” отмечается 13 или 12 сентября. Почему именно в эти даты?

Указанные даты соответствуют 256-му дню года. В невисокосный год этот день приходится на 13 сентября, в високосный год — на 12 сентября. А число 256 выбрано потому, что, как известно, это количество различных значений, которые можно выразить с помощью байта, или с помощью восьми бит, — 2^8 .

Вариант вопроса

Профессиональный праздник “День программиста” отмечается 13 или 12 сентября. Почему возможны варианты?

4. Вы, наверное, знаете, что носителями информации в электронно-вычислительных машинах первых

поколений были так называемые “перфоленты” (или перфорированные ленты) и “перфокарты” (или перфорированные карты), на которых информация представлялась в двоичном виде с помощью отверстий.

При использовании перфокарт программы представляли собой их комплект (колоду).

Вопрос: “В чем преимущество использования перфокарт по сравнению с перфолентами, кроме того, что из-за более плотной бумаги они были долговечнее перфолент?”

Вариант вопроса

Вы, наверное, знаете, что носителями информации в электронно-вычислительных машинах первых поколений были так называемые “перфоленты” (или перфорированные ленты) и “перфокарты” (или перфорированные карты), на которых информация представлялась в двоичном виде с помощью отверстий.

На одной ленте представлялась вся программа, а при использовании перфокарт программы представляли собой их комплект (колоду).

Вопрос: “В чем преимущество использования перфокарт по сравнению с перфолентами, кроме того, что из-за более плотной бумаги они были долговечнее перфолент?”

5. Вы, наверное, знаете, что носителями информации в электронно-вычислительных машинах первых поколений были так называемые “перфокарты” (или перфорированные карты), на которых информация представлялась в двоичном виде с помощью отверстий*.

Можно сказать, что программы представляли собой комплект (колоду) перфокарт.

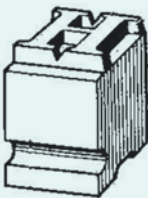
А теперь — внимание! Вопрос: “Что общего у технологии использования колоды перфокарт и работе текстовых редакторов?”

6. Вы, конечно, знаете, что такое азбука Морзе. Рассказывают, что, создавая свой код, Морзе отправился в ближайшую типографию, в которой в те времена, да и в течение длительного времени потом, текст в книгах, газетах и т.п. набирался вручную из отдельных литер. Литера — металлический, пластмассовый или деревянный брусок прямоугольного сечения с рельефным изображением буквы или знака на одной из его сторон (участникам конкурса демонстрируется изображение литеры).

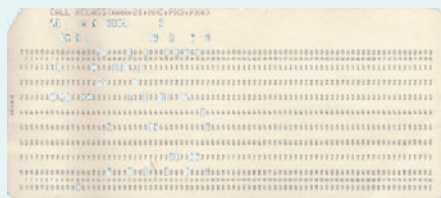
Зачем Морзе ходил в типографию?

Вариант вопроса

Вы, конечно, знаете, что такое азбука Морзе. Рассказывают, что, создавая свой код, Морзе отправился в ближайшую типографию, в которой в те времена, да и в течение длительного времени потом, текст в книгах, газетах и т.п. набирался вручную из отдельных литер. Там он подсчитал число литер в наборных кассах. Литера — металлический, пласт-



* Участникам конкурса демонстрируется изображение перфокарты (см. рисунок вверху страницы).



массовый или деревянный брусок прямоугольного сечения с рельефным изображением буквы или знака на одной из его сторон (участникам конкурса демонстрируется изображение литеры). Наборная касса — ящик, в котором помещались литеры.

Зачем Морзе ходил в типографию?

Ответы

1. Речь идет о так называемой “дискете” — гибком магнитном диске. Ее изображение используется в качестве значка для кнопки и пункта меню *Сохранить*.

2. Неверным в “расшифровке” является слово *Video*. DVD-диски могут быть использованы для хранения не только видео, но и дру-

гих видов информации. Поэтому через некоторое время после их появления многие стали расшифровывать DVD как Digital Versatile Disc (цифровой многоцелевой диск). Фирма Toshiba, заведующая официальным сайтом DVD Forum, использует именно обозначение “Digital Versatile Disc”. К консенсусу специалисты не пришли до сих пор, поэтому сегодня “DVD” официально вообще никак не расшифровывается.

3. Указанные даты соответствуют 256-му дню года. В невисокосный год этот день приходится на 13 сентября, в високосный год — на 12 сентября. А число 256 выбрано потому, что, как известно, это количество различных значений, которые можно выразить с помощью байта, или с помощью восьми бит, — 2^8 .

Ответ на вариант вопроса

Профессиональный праздник программистов отмечается в 256-й день года (число 256 выбрано потому, что, как известно, это количество различных значений, которые можно выразить с помощью байта, или с помощью восьми бит, — 2^8). В невисокосный год этот день приходится на 13 сентября, в високосный год — на 12 сентября.

4. Главным преимуществом перфокарт было удобство работы с программой в случае наличия ошибки и вообще, при необходимости, внесения изменений — в любом месте колоды можно было добавить карты, удалить, заменить одни карты другими.

5. Главным преимуществом перфокарт было удобство работы с программой в случае наличия ошибки и вообще, при необходимости, внесения изменений — в любом месте колоды можно было добавить карты, удалить, заменить одни карты другими, то есть фактически выполнять многие функции, позже реализованные в текстовых редакторах.

6. Морзе подсчитал число литер в наборных кассах (наборная касса — ящик, в котором помещались литеры). Буквам и знакам, для которых литер в этих кассах было припасено больше, он сопоставил более короткие кодовые обозначения (ведь эти буквы повторяются в тексте чаще). Так, например, в русском варианте азбуки Морзе буква *e* передается одной точкой, а редко встречающаяся буква *ц* — набором из четырех символов. Поэтому азбука Морзе относится к так называемым “неравномерным” системам кодирования символов, то есть таким, в которых разным символам соответствуют коды различной длины.

Ответ на вариант вопроса (отличается началом)

Буквам и знакам, для которых литер в наборных кассах было припасено больше, ...



ЗАДАЧНИК

Фальшивые монеты, отравленное вино и... СИСТЕМЫ СЧИСЛЕНИЯ

Д.М. Златопольский

Рассмотрим ряд задач.

Задача 1

Имеются 64 одинаковые по виду золотые монеты. 63 из них настоящие, а одна — фальшивая, которая легче настоящих. За какое наименьшее число взвешиваний на рычажных весах с чашками и без гирь можно определить, какая из монет фальшивая?

Прежде чем описывать решение, заметим, что фраза “наименьшее число взвешиваний” в условии означает, что должны рассматриваться “худшие” варианты, а не варианты типа “а вдруг повезет”. Например, вариант, когда при первом взвешивании сравниваются две монеты и одна из чашек весов перевешивает (“повезло!”), рассматриваться не должен. Аналогично, не должен рассматриваться, например, такой вариант:

№ взвешивания	Берем монеты	1-я чашка весов	2-я чашка весов	Допустим, что перевесила
1		32	32	2-я чашка
2	С 1-й чашки	15	15	1-я чашка
3	Со 2-й чашки	7	7	Весы в равновесии

— при котором за три взвешивания устанавливается, что фальшивая монета — 15-я из той группы, которая оказалась легче при втором взвешивании.

Самое простое¹ решение: последовательно делить все количество монет пополам, на весах сравнивать их, затем брать более легкую кучку монет, делить ее пополам — и т.д., пока при последнем взвешивании не останется сравнить между собой

¹ Но, как будет показано ниже, не самое рациональное.

две монеты. Более легкая из них и есть фальшивая. Сказанное можно оформить в виде таблицы (синим цветом и жирным начертанием выделены кучки монет, содержащие фальшивую монету, а также сама эта монета):

№ взвешивания	Берем монеты	1-я чашка весов	2-я чашка весов	Допустим, что перевесила
1		32	32	2-я чашка
2	С 1-й чашки	16	16	1-я чашка
3	Со 2-й чашки	8	8	1-я чашка
4	Со 2-й чашки	4	4	2-я чашка
5	С 1-й чашки	2	2	1-я чашка
6	Со 2-й чашки	1	1	2-я чашка

Возможный ответ: потребуется шесть взвешиваний.

Задания для самостоятельной работы

1. Нарисуйте блок-схему, описывающую все возможные варианты взвешиваний по рассмотренной методике при 16 монетах.

2. Определите, какое минимальное число взвешиваний потребуется для поиска фальшивой монеты среди 2^n монет.

Задача 2

Есть восемь монет, из которых семь настоящих, каждая из которых весит 10 г, и одна фальшивая весом 9 г. Необходимо, используя электронные весы, показывающие вес с точностью 1 г, найти фальшивую монету за минимальное число взвешиваний.

Решение во многом аналогично предыдущей задаче.

При первом взвешивании определим вес любых четырех монет, например, 1, 2, 3 и 4. Если весы покажут 40 г, то фальшивая монета находится среди оставшихся монет, если 39 г — среди взвешиваемых. Затем нужно узнать вес любых двух монет из соответствующей группы и выявить пару монет, среди

которых есть фальшивая. Третье взвешивание покажет, какая монета из этой пары фальшивая.

Задание для самостоятельной работы

3. Нарисуйте блок-схему, описывающую все возможные варианты взвешиваний по рассмотренной методике решения задачи 2.

Интересно, что для решения задачи можно применить... двоичную систему счисления!

Пронумеруем монеты: 1, 2, ..., 8 — и представим номера монет в виде трехразрядных двоичных чисел:

Номер монеты	1	2	3	4	5	6	7	8
Условный номер	0	1	2	3	4	5	6	7
Двоичное представление	000	001	010	011	100	101	110	111

В первом взвешивании должны участвовать монеты, содержащие единицу в первом разряде, во втором — монеты с единицей во втором разряде, в третьем взвешивании — с единицей в третьем разряде. Результаты измерений будем записывать так: если общий вес монет равен 39 г, то во взвешиваемой группе содержится фальшивая монета — запишем 1 в соответствующий номеру измерения разряд, в противном случае запишем 0. Полученное двоичное число будет однозначно определять номер фальшивой монеты.

Пример. Пусть фальшивая монета имеет номер 6. Ее условный номер: $5_{10} = 101_2$, значит, эта монета будет участвовать в первом и третьем взвешиваниях. При них весы покажут 39 г, а при втором взвешивании — 40 г, т.е. в результате получится двоичное число 101, соответствующее условному номеру фальшивой монеты.

Обоснование

Обсудим вопрос: “В каком случае при каком-то взвешивании весы покажут 39 г?”. Ответ — если среди взвешиваемых монет есть фальшивая. А 40 г? — Если ее там нет. Число взвешиваний равно количеству двоичных разрядов в записи номеров. Фальшивая монета будет участвовать в тех взвешиваниях, номера которых соответствуют номеру разряда в ее двоичной записи с единицами. И именно при этих взвешиваниях весы покажут 39 г (а мы запишем в соответствующем разряде 1). В остальных взвешиваниях фальшивая монета участвовать не будет, то есть при них весы покажут 40 г (а мы запишем 0 в соответствующем разряде). Следовательно, двоичное число, составленное по результатам всех взвешиваний, будет таким же, как двоичный номер искомой монеты.

Задание для самостоятельной работы

4. Подумайте, будет ли работать описанная методика, когда общее число монет равно:

- 1) 16;
- 2) 14.

Задача 3

Имеются девять одинаковых по виду золотых монет. Восемь из них настоящие, а одна — фальшивая, которая легче настоящих. За какое наименьшее число взвешиваний на рычажных весах с чашками и без гирь можно определить, какая из монет фальшивая?

Если решать задачу так же, как задачу 1, то понадобится три взвешивания (убедитесь в этом самостоятельно). Но, оказывается, можно решить ее всего за два взвешивания!

Для этого можно разделить все монеты не на две, а на три кучки, после чего сравнить на весах две любые кучки. Если весы в равновесии, то фальшивая монета — в третьей кучке из трех монет, в противном случае — среди тех, которые на весах перевесили. Таким образом, в любом случае мы выделим кучку из трех монет, среди которых есть искомая. Сравнив на весах две любые монеты из этой кучки, мы этим, вторым, взвешиванием установим, какая из монет фальшивая.

Задание для самостоятельной работы

5. Решите задачу: “Купец получил за товар 64 золотые монеты. Потом он узнал, что одна из этих монет фальшивая, которая весит больше настоящих. За какое наименьшее число взвешиваний на рычажных весах с чашками и без гирь купец сможет определить, какая из монет фальшивая?”

Указания по выполнению. Разделяйте монеты, среди которых есть фальшивая, на три части (в том числе неравные, если число монет не кратно 3). Например, первоначальную кучу из 64 монет можно разделить на три кучки из 21, 21 и 22 монет. В любом случае необходимо, чтобы две из этих трех кучек были равными, а третья может от них отличаться, при этом размер самой большой кучки должен быть как можно меньше.

Оказывается, что и в решении задачи 3 нам опять поможет система счисления, но не двоичная, а троичная!

Закодируем номера монет в троичной системе счисления:

Номер монеты	1	2	3	4	5	6	7	8	9
Условный номер	0	1	2	3	4	5	6	7	8
Троичное представление	00	01	02	10	11	12	20	21	22

Задачу можно решить за два взвешивания. Для первого взвешивания рассмотрим левый разряд троичных чисел — 1 в нем будет означать, что монету нужно положить на левую чашку весов, 2 — на правую чашку, а 0 — что монета не участвует во взвешивании. Затем аналогично для второго разряда.

Запишем результаты каждого i -го взвешивания ($i = 1, 2$) в соответствующий разряд по правилу: если перевесила левая чашка — 1, если правая — 2, если весы уравновешены — 0. Полученное троичное число однозначно определяет номер фальшивой монеты.

Проверьте сказанное самостоятельно на примере какой-то монеты. Почему так получается?

Задание для самостоятельной работы

6. Решите задачу 3 для случая, когда монет — 27.

Задача 4

Когда-то в Древнем Риме некий патриций решил устроить праздник и для этого приготовил четыре бочки вина. Однако он узнал, что кто-то из его врагов подсыпал яд в одну из бочек. Про яд известно, что человек, который его выпил, умирает в течение 24 часов. До начала праздника осталось 25 часов. У патриция есть рабы, которыми он готов пожертвовать, чтобы узнать, в какой именно бочке яд. Какое минимальное число рабов необходимо “использовать”, чтобы патриций узнал это?

И здесь, прежде чем описывать решение, обратим внимание на то, что фраза “минимальное число” в условии означает, что должны рассматриваться “худшие” варианты, а не варианты типа “а вдруг патрицию повезет”. Например, вариант, когда один раб пьет вино из одной из бочек и умирает (“Патрицию повезло!”), рассматриваться не должен.

Самое простое решение — “использовать” трех рабов, каждый из которых пьет вино из какой-то одной бочки (все — из разных). Если через 24 часа никто не умрет, то отравленное вино — в 4-й бочке, в противном случае один из рабов ценой своей жизни “укажет” на бочку с ядом.

А если трех рабов у патриция нет? Можно ли решить задачу в этом случае? Оказывается, можно! Для этого опять следует применить двоичную систему счисления!

Пронумеруем² бочки: 1, 2, 3, 4 — и представим номера бочек в виде двухразрядных двоичных чисел:

Номер бочки	1	2	3	4
Условный номер	0	1	2	3
Двоичное представление	00	01	10	11

Двух имеющихся рабов тоже пронумеруем. Раб с номером 1 должен выпить вино из тех бочек, у кото-

² Мы используем глаголы “пронумеруем” и другие в 1-м лице исключительно для более понятного изложения методики решения задач. Естественно, что мы считаем недопустимым использование людей для решения подобных задач.

рых в первом разряде их двоичного условного номера стоит 1 (то есть из бочек с номерами 10 и 11), а раб с номером 2 — из тех бочек, у которых единица стоит во втором разряде их двоичного номера (01 и 11).

Если через 24 часа никто из рабов не умер (ура!), то яд в бочке с номером 1 (с двоичным условным номером 00).

Если умер раб с номером 1, то яд в бочке с двоичным условным номером 10, если раб с номером 2 — в бочке с номером 01.

К сожалению, возможен также и вариант, когда погибнут оба раба...

Задания для самостоятельной работы

7. Определите, какое минимальное число рабов необходимо “использовать”, чтобы патриций решил задачу, когда общее число бочек равно:

- 1) 8;
- 2) 16;
- 3) 2^k .

8. Определите, можно ли при восьми бочках решить задачу с меньшим числом рабов, но за большее время.

9. Сравните два варианта решения аналогичной задачи, когда общее число бочек равно 8, а до начала пирши осталось 49 часов:

1) при котором в первые 24 часа “используются” два раба, а затем, при необходимости, “эксперимент” (повторим — недопустимый) проводится в следующие 24 часа;

2) при котором в первые 24 часа “используются” три раба.

Можно ли сделать вывод о том, какой вариант “гуманнее”?

Задача 5

Когда-то в Древнем Риме некий патриций решил устроить праздник и для этого приготовил девять бочек вина. Однако он узнал, что кто-то из его врагов подсыпал яд в одну из бочек. Про яд известно, что человек, который его выпил, умирает в течение



Паоло Веронезе “Пир в доме Левия”

24 часов. До начала праздника осталось 49 часов. У патриция есть рабы, которыми он готов пожертвовать, чтобы узнать, в какой именно бочке яд. Какое минимальное число рабов необходимо “использовать”, чтобы патриций узнал это?

Если решать задачу так же, как задачу 4, то “понадобится” четыре раба (убедитесь в этом самостоятельно). Но, оказывается, можно решить ее всего с двумя рабами! Как это сделать, вы, конечно, уже догадались — следует применить не двоичную систему счисления, а троичную.

Закодируем номера бочек в троичной системе счисления:

Номер монеты	1	2	3	4	5	6	7	8	9
Условный номер	0	1	2	3	4	5	6	7	8
Троичное представление	00	01	02	10	11	12	20	21	22

Рабов тоже пронумеруем (1 и 2). Раб с номером 1 должен выпить вино из тех бочек, у которых в первом разряде их троичного условного номера стоит 1 (то есть из бочек с номерами 10, 11 и 12), а раб с номером 2 — из тех бочек, у которых единица стоит во втором разряде их троичного номера (01, 11 и 21).

Если через 24 часа умрут оба раба, то яд находится в бочке, из которой они пили, то есть с троичным номером 11.

В случаях, когда погибнет только один раб, станут известны две бочки, из которых он пил, а второй раб останется живым. Значит, за вторые 24 часа можно найти “нужную” бочку.

Если же через 24 часа никто не умрет, то яд находится в одной из четырех непроверенных бочек. И в этом случае за вторые 24 часа можно решить данную задачу (см. задачу 4).

Задания для самостоятельной работы

10. Определите, какое минимальное число рабов необходимо “использовать”, чтобы патриций решил задачу 5, когда общее число бочек равно:

- 1) 27;
- 2) 81;
- 3) 3^k ;
- 4) n .

11. Заполните таблицу с результатами определения необходимого числа рабов в случае использования для расчета двоичной и троичной систем счисления (без учета фактора времени):

Число бочек n	Необходимое число рабов		Разность
	Использование двоичной системы	Использование троичной системы	
8			
16			
32			
...			

Определите, при каком значении общего числа бочек n разность значений во втором и третьем

столбцах станет равна 5 (допустив, что такое число бочек возможно).

12. (Задание для читателей, владеющих каким-либо языком программирования.) Разработайте компьютерную программу, с помощью которой можно решить задачу о патриции при любом числе бочек n и при заданном количестве часов, оставшихся до пира (25 или 49).

Результаты присылайте в редакцию (можно выполнять не все задания). Авторы лучших ответов будут награждены дипломами.

Литература

1. Златопольский Д.М., Усенков Д.Ю. Задачи на взвешивания. / “В мир информатики” № 171 (“Информатика” № 17/2011).

Где пожар?

В пожарную часть поступил звонок, в котором были два сообщения: “У нас пожар — скорее приезжайте!” и “Пожар в городе Тотактосяк”. Звонить могли из одного из трех городов: Правдинска, жители которого говорят только правду, Ложенца, жители которого всегда лгут, или Тотактосяк — в котором обитатели говорят попеременно правду и ложь (то есть из каждых двух высказанных ими утверждений одно всегда истинно, а другое — ложно). В какой город должна была приехать пожарная машина, чтобы потушить пожар?

Шесть преподавателей

В педагогическом институте Аркадьева, Бабанова, Корсакова, Дашков, Ильин и Флеров преподают экономическую географию, английский язык, немецкий язык, историю, французский язык и математику.

Известно, что:

- 1) преподаватель немецкого языка и преподаватель математики в студенческие годы занимались художественной гимнастикой;
- 2) Ильин старше Флерова, но стаж работы у него меньше, чем у преподавателя экономической географии;
- 3) будучи студентками, Аркадьева и Бабанова учились вместе в одном университете. Все остальные окончили педагогический институт;
- 4) Флеров — сын преподавателя французского языка, но студентом у него не был;
- 5) преподаватель французского языка — самый старший из всех по возрасту и по стажу работы. Он работает в педагогическом институте с тех пор, как окончил его. Преподаватели математики и истории — его бывшие студенты;
- 6) Аркадьева старше преподавателя немецкого языка.

Кто какой предмет преподает?

Литература

1. Богомолова О.Б. Логические задачи. М.: БИНОМ. Лаборатория знаний, 2005.

Какое число задумано?

Задумано трехзначное число, у которого с любым из чисел 543, 142 и 562 совпадает один из рядов, а два других не совпадают. Какое число задумано?

“Лимонадная” статистика

Одна фирма провела опрос любителей лимонада — желтого и красного. Вот ее отчет:

- общее число опрошенных: 300;
- количество потребителей желтого лимонада: 234;
- число потребителей красного лимонада: 213;
- количество потребителей любого лимонада: 144;
- число “непотребителей” лимонада: 0.

Найдите ошибку в отчете.

ПОИСК ИНФОРМАЦИИ

Три вопроса

Ответы на заданные вопросы найдите в Интернете или по другим источникам информации.

1. В каком городе Уильям Боинг в 1916 году построил авиазавод?
2. Между какими командами матч в высшей лиге чемпионата СССР по футболу 1979 года закончился со счетом 5:2?
3. Какой породы была собака, о которой главный герой романа Владимира Набокова “Лолита” рассказывал, что “...мы едва не сбили навязчивую пригородную собаку (из тех, что устраивают засады автомобилям)”?

Можно отвечать не на все вопросы.

СЕМИНАР

Еще две замечательные кривые

В недавней публикации [1] была приведена программа, с помощью которой можно получить на экране изображение двух кривых — улитки Паскаля и кардиоиды. Существуют и другие кривые, которые называют “замечательными”. Они носят, как правило, “звучные” имена, например, “астроида”, “локон Аньези”, “окружность Аполлония”, “трактриса” и т.п.

В этой статье мы рассмотрим две замечательные кривые — эпициклоиду и гипоциклоиду. Они изображены, соответственно, на рис. 1 и 2.

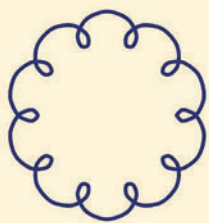


Рис. 1

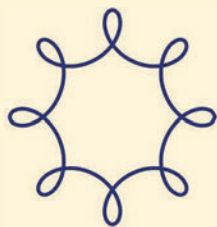


Рис. 2

Заметим, что это не просто красивые картинки. Оба изображения имеют “геометрический” смысл — это линия, которую образует точка, закрепленная в плоскости некоторого круга радиуса r (производящий круг), когда круг катится без скольжения по неподвижной окружности радиуса R_n (направляющая). На рис. 3 красным цветом показана часть AM эпициклоиды, по которой перемещается точка M производящего круга.

Когда окружности касаются внешним образом, линия называется “эпициклоидой” (от греческих слов $\epsilon\pi\acute{\iota}$ — на, над, при и $\kappa\acute{o}\kappa\lambda\omicron\varsigma$ — круг, окружность), когда касание внутреннее — “гипоциклоидой” (от греческих слов $\upsilon\pi\acute{o}$ — под, внизу и $\kappa\acute{o}\kappa\lambda\omicron\varsigma$).

Обе кривые имеют также варианты в зависимости от того, где находится точка M , которая “образует” линии [2]. Обозначим расстояние от этой точки до центра производящего круга — d . Если $d = r$ (точка M

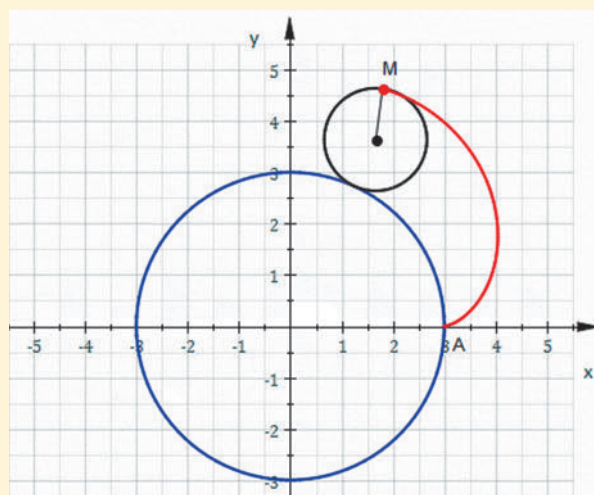


Рис. 3

расположена на окружности производящего круга), то такие эпициклоида и гипоциклоида называются “обыкновенными” (на рис. 3 как раз и показан такой вариант). Когда точка M взята внутри производящего круга ($d < r$) — линии называются “укороченными”, когда вне его ($d > r$) — “удлиненными”.



Мухаммед Насирэддин ат-Туси и Николай Коперник

После такого теоретического введения приступим к разработке программ, с помощью которых можно получить изображения эпициклоиды и гипоциклоиды.

Как и в случае с улиткой Паскаля [1], это удобно делать с помощью параметрических уравнений кривой. Для эписциклоиды они следующие:

$$x = (R_n + r)\cos(\varphi) - d\cos\left(\frac{R_n+r}{r}\varphi\right);$$

$$y = (R_n + r)\sin(\varphi) - d\sin\left(\frac{R_n+r}{r}\varphi\right).$$

Напомним, что параметрическими такие уравнения называются потому, что определяют значения координат x и y каждой точки кривой в зависимости от некоторого параметра, в нашем случае от параметра φ — угла наклона отрезка, соединяющего эту точку с началом координат. Кроме того, в уравнениях есть величины R_n , r и d , смысл которых раскрыт выше.

Используя параметрические уравнения линии, можно получить ее изображение в программе — для этого надо рассчитать значения координат x и y для всех углов, скажем, от 1 до 360 градусов через 1 градус, и поставить точку в соответствующем месте экрана. Программа на школьном алгоритмическом языке, решающая такую задачу для эписциклоиды, имеет вид:

```

алг Эписциклоида
нач цел  $x$ ,  $y$ , угол,  $x0$ ,  $y0$ ,
      вещ  $R_n$ ,  $r$ ,  $d$ , угол2
|Характеристики кривой
вывод "Задайте значение  $r$  "
ввод  $r$ 
вывод "Задайте значение  $R_n$  "
ввод  $R_n$ 
вывод "Задайте значение  $d$  "
ввод  $d$ 
|Иницилируем графический режим
...
|Координаты центра экрана
 $x0 := \text{int}(\text{макс}X/2)$ ;  $y0 := \text{int}(\text{макс}Y/2)$ 
|Для всех целых значений углов
нц для угол от 1 до 360
  |Переводим угол в радианы
   $\text{угол2} := 6.28 * \text{угол}/360$ 
  |Рассчитываем координаты
  |соответствующей точки кривой
   $x := x0 + \text{int}((R_n + r) * \cos(\text{угол2}) -$ 
     $d * \cos((R_n + r)/r * \text{угол2}))$ 
   $y := y0 - \text{int}((R_n + r) * \sin(\text{угол2}) -$ 
     $d * \sin((R_n + r)/r * \text{угол2}))$ 
  |и изображаем эту точку
  точка( $x$ ,  $y$ )
кц
кон
  
```

Примечания

- $x0$ и $y0$ — координаты центра экрана (с учетом этих координат рассчитываются значения x и y); значения $x0$ и $y0$ зависят от величин $\text{макс}X$ и $\text{макс}Y$, равных, соответственно, максимальному значению координат x и y в выбранном режиме работы экрана.
- угол2 — величина угла в радианах.

3. Функция `int` возвращает целую часть ее вещественного аргумента.

4. Ось Y на экране направлена вниз, поэтому в формуле для расчета координаты y использован знак “минус”.

Для построения эписциклоид и гипоциклоид можно использовать также электронную таблицу Microsoft Excel или подобную программу. Верхняя часть листа, на котором можно сделать это применительно к эписциклоиде, показана на *рис. 4* (необходимые формулы запишите самостоятельно):

	A	B	C	D	E	F	G
1	Угол в градусах	Угол в радианах	x	y		Rn =	100
2	0	0,00	90,00	0,00		d =	20
3	5	0,09	98,10	-6,80		r =	10
4	10	0,17	115,15	0,29			
5	15	0,26	125,57	23,25			
...							

Рис. 4

По полученным расчетным данным можно построить график (тип — **Точечная с гладкими кривыми**).

Задание для самостоятельной работы

1. На изучаемом вами языке программирования разработайте программу, с помощью которой можно получить изображение гипоциклоиды. Ее параметрические уравнения:

$$x = (R_n - r)\cos(\varphi) + d\cos\left(\frac{R_n+r}{r}\varphi\right);$$

$$y = (R_n - r)\sin(\varphi) - d\sin\left(\frac{R_n+r}{r}\varphi\right).$$

2. Оформите листы электронной таблицы Microsoft Excel для получения эписциклоид и гипоциклоид. Необходимые формулы определите самостоятельно.

3. Установите, что определяют значения величин R_n , r и d на изображении обсуждаемых замечательных кривых.

4. Определите, как будут выглядеть:

а) укороченные и удлиненные эписциклоиды и гипоциклоиды:

- при $R_n = r$;
- при $R_n/r = 2$;

- б) обыкновенная эписциклоида и гипоциклоида:
 - при $R_n = r$;
 - при $R_n/r = 4$.

Результаты, пожалуйста, присылайте в редакцию (можно выполнять не все задания). Фамилии всех приславших ответы будут опубликованы, а лучшие ответы мы поощрим.

В заключение приведем краткую историческую справку о рассмотренных замечательных кривых [2].

Чтобы объяснить попятные движения планет, древнегреческие астрономы, следуя Гиппарху (II век до н.э.), приписывали им равномерное движение по окружности (*эписцикл*),

центр которой равномерно движется по другой окружности (*деферент*). Линия, описываемая точкой при этих условиях, является эпициклоидой. Мы не знаем, однако, какие геометрические ее свойства были известны ученым древности. В середине XIII века выдающийся мусульманский астроном и математик Мухаммед Насирэддин ат-Туси (1201–1274) установил, что точка окружности, катящейся по неподвижной окружности вдвое большего радиуса, касаясь ее изнутри, описывает диаметр неподвижной окружности. Это свойство независимо от Насирэddина было найдено великим польским астрономом Николаем Коперником (1473–1543); оно содержится в его знаменитом труде “Об обращениях небесных кругов”, опубликованном в 1543 г. Теорема Насирэddина — Коперника нашла широкое применение в прикладной механике.

Начало систематического изучения эпициклоид и гипоциклоид было положено в 1525 г. знамени-

тым немецким художником Альбрехтом Дюрером (1471–1528), широко применявшим геометрические методы в изобразительном искусстве. Однако математикам исследования Дюрера остались неизвестными.

В середине XVII века Ж.Дезарг (1593–1662), у которого глубина математических идей сочеталась с талантами конструктора, изучал свойства эпициклоид в связи с задачей создания зубчатых колес с наименьшим трением. Результаты этих, как и многих других, исследований Дезарга не были им опубликованы, но они были известны в кругу его друзей.

Ла Гир, продолживший исследования Дезарга, опубликовал в 1675 г. “Трактат об эпициклоидах и их применении в механике”. В книге установлен ряд важных свойств эпициклоид.

Литература

1. О кардиоиде и о сердце. / “В мир информатики” № 203 (“Информатика” № 12/2014).
2. *Выгодский М.Я.* Справочник по высшей математике. М.: Наука, 2000.

“ЛОМАЕМ” ГОЛОВУ

Вычислительные устройства

А.А. Мячев,
Москва

В приведенной ниже таблице записаны названия ручных и механических вычислительных устройств и фамилии их изобретателей. Читать можно по всем направлениям — слева направо, справа налево, снизу вверх и сверху вниз, но не по диагоналям. Комментарии к названиям и фамилиям даются. Найдите их. При необходимости используйте Интернет или другие источники информации.

Е	Р	А	Р	И	Ф	М	О	М	Е	Т	Р
К	Ь	Л	Е	Й	Б	Н	И	Ц	Ш	И	Т
А	Л	Ч	К	И	Н	Е	П	Е	Р	К	О
С	А	О	И	Т	Е	Л	Ь	К	С	К	М
С	К	Л	Л	А	Н	С	У	У	Ч	А	А
И	С	А	С	Б	А	Б	А	М	Е	Р	С
Б	А	П	И	О	К	А	Н	М	Т	Д	О
А	П	Ы	Ч	Р	О	С	Ь	Е	П	Г	Д
К	А	Т	С	Ь	Н	А	П	Р	Л	Ю	Н
Й	Н	Е	Ч	С	О	М	А	С	А	Н	Е
Е	И	Ш	А	М	А	Н	И	Т	С	Т	Р
Н	И	Л	Д	Е	Р	Т	У	О	Р	Е	Б
З	Ю	О	С	А	Г	И	В	С	Н	У	Р

Комментарии

1. Общее название счетных приборов древности.
2. Китайские счеты.
3. Японские счеты.
4. Предшественник счетов — старинный русский вычислительный прибор — дощаной...
5. Изобретатель Генрих..., разработавший счетный прибор, в котором при сложении осуществляется перенос единицы в соседний (старший) разряд,

а при вычитании осуществляется заимствование единицы из старшего разряда.

6. Счетный прибор инженера, описанный в п. 5.
7. Вычислительное устройство В.Буныковского.
8. Шотландский барон, изобретатель счетного прибора.
9. Приспособление для быстрого умножения изобретателя из п. 8.
10. Счетный прибор, который иногда используют для обучения счету в начальной школе, — счетная...
11. Первая вычислительная...
12. Изобретатель одной из первых вычислительных машин — Блез...
13. Изобретатель одной из первых вычислительных машин — Готфрид...
14. Профессор Вильгельм..., автор первой вычислительной машины (которая, правда, не сохранилась).
15. Эдмунд..., предложивший в вычислительном устройстве логарифмическую шкалу, которая использовалась вместе с двумя циркулями-измерителями.
16. Логарифмическая...
17. Конструктор вычислительного прибора из п. 16.
18. Роберт..., конструктор прямоугольного варианта вычислительного прибора из п. 16.
19. Механическое вычислительное устройство для умножения многозначных чисел (на котором можно также эти числа складывать, вычитать и делить нацело с остатком).
20. Конструктор Карл..., предложивший название устройства из п. 19 и разработавший соответствующее устройство.
21. Изобретатель вычислительного прибора — Вильгодт...

22. Марка вычислительного прибора из п. 19, выпускавшегося в Германии в г. Брауншвейг и поставлявшегося в Россию в начале XX века.

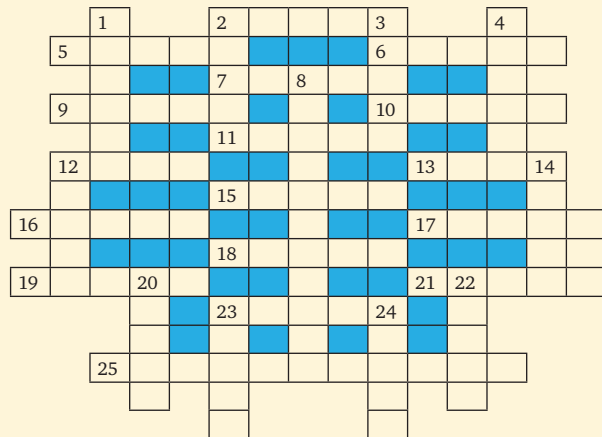
23. Марка вычислительного прибора из п. 19, выпускавшегося в СССР наряду с моделями “Феликс” и др.

Литература

1. Златопольский Д.М. Музей истории вычислительной техники / “Информатика” № 2/2010.

Кроссворд

Решите, пожалуйста, кроссворд.



По горизонтали

- 2. Одна из наиболее популярных услуг, предоставляемых телекоммуникационными сетями.
- 5. Последовательность букв и цифр, ограниченная с обоих концов пробелами, запятыми, точками, дефисами и т.п.
- 6. Программа, обладающая способностью к самовоспроизведению.
- 7. Последовательность символов, предназначенная для чтения человеком.
- 9. В мониторах так называют три пятнышка люминофора разного цвета (красного, зеленого и синего), а размер этого элемента является важным параметром монитора.
- 10. Список учитываемых предметов (имущества, документов и т.п.).
- 11. Первая составная часть сложных слов, имеющая значение “относящийся к слову 8 по вертикали”.
- 12. Структура данных, в которых применен принцип “последним пришел — первым вышел”.
- 13. Единица измерения частоты (колебаний и т.п.).
- 15. Изобретатель системы кодирования информации, использующей два символа — точку и тире.
- 16. Взломщик компьютерных программ.
- 17. Первая буква греческого алфавита.
- 18. Инструментальная программная оболочка, в которой происходит работа пользователя, а также день недели.
- 19. Координаты ячейки на листе электронной таблицы.

21. В текстовом редакторе Microsoft Word — текст, набранный до нажатия клавиши “Enter”.

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

25. Замена одного другим, например, одной части слова на другую.

По вертикали

- 1. Программа, обращающаяся с запросами к другой программе.
- 2. Непрерывная последовательность данных.
- 3. Человек — разработчик программы.
- 4. Указатель места ввода очередного символа на экране.
- 8. Наука об общих закономерностях получения, хранения, передачи и преобразования информации в сложных управляющих системах, будь то машины, живые организмы или общество.
- 12. Элемент презентации программы Microsoft PowerPoint.
- 14. Знак, обозначающий число.
- 20. Андрей Петрович... — советский академик, один из основоположников школьной информатики.
- 22. Пластинка с надписью (на товарах, на багаже, грузах).
- 23. Цифра восьмеричной системы счисления.
- 24. Сверхъестественное существо из скандинавской мифологии — карлик, великан, ведьма. В компьютерных играх они представляют собой горных духов, ассоциируемых с камнем, обычно враждебных человеку. Соответствующее понятие используется также применительно к форумам в Интернете. (Внимание! В этом слове вместо двух одинаковых букв следует вписать одну.)

Двуносый чайник и три чашки

Мы продолжаем публиковать задачи с необычным чайником — с двумя носиками. В данном случае предлагаем читателям решить задачи, связанные с заполнением трех чашек разного объема.

Задача 2

Как, используя двуносый чайник, полностью заполнить три чашки:

- 1) вместимостью 10, 12 и 14 условных единиц;
- 2) вместимостью 4, 4 и 16 условных единиц;
- 3) вместимостью 5, 5 и 25 условных единиц.

Во всех случаях принять, что в чайнике имеется достаточно большое количество воды.

Алгоритмы решения задач, пожалуйста, оформите в виде таблицы:

№	Действие	В 1-й чашке	Во 2-й чашке	В 3-й чашке
Исходное состояние		0	0	0
1	Налить в...			
2				
...				

Числовые ребусы в троичной системе. Часть 5

В приведенных ниже ребусах зашифрованы числа, записанные в троичной системе счисления. Одинаковым буквам соответствуют одинаковые цифры. Звездочкой (“*”) может быть любая цифра.

$$1. \begin{array}{r} + \quad * \quad * \\ \quad * \quad D \\ \hline D \quad D \end{array}$$

$$2. \begin{array}{r} + \quad * \\ \quad * \quad A \\ \hline A \quad * \quad * \end{array}$$

$$3. \begin{array}{r} + \quad C \\ \quad * \quad * \\ \hline C \quad * \quad * \\ * \quad * \end{array}$$

Ответы присылайте в редакцию (можно решать не все ребусы).

“АДРЕС” и “СРЕДА”

На листе бумаги нарисованы девять клеток (рис. 1).

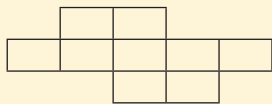


Рис. 1

На них положили пять карточек с буквами, образующих слово “АДРЕС” (рис. 2):

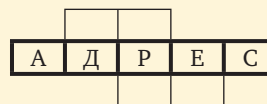


Рис. 2

Необходимо, перекладывая карточки на соседнюю свободную клетку листа, за минимальное количество переключений получить слово “СРЕДА” (рис. 3):

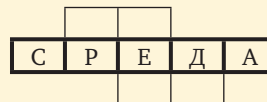


Рис. 3

Алгоритм решения задачи, пожалуйста, оформите с использованием символов “>” (перемещение карточки вправо), “<” (влево), “V” (вниз), “^” (вверх) в виде:

1. Р ^.
2. Е <.

Числовой ребус с “АМУР”ом

Решите, пожалуйста, числовой ребус:

$$АМУР^2 = ****МУАР.$$

Одинаковыми буквами обозначены одинаковые цифры, разными буквами — разные цифры. Звездочкой (“*”) может быть любая цифра.

Добейтесь равенства

Дано выражение:

$$987654321 = 100.$$

Сделайте его правильным, используя четыре знака “+” или “-” в его левой части.

ЯПОНСКИЙ УГОЛОК



Два sudoku

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	

ВНИМАНИЕ! КОНКУРС!

Конкурс № 111 “Переpravы”

Напомним, что задания данного конкурса представляют собой задачи на переpravы. Конкурс проводится в несколько туров, а его итоги будут подводиться с учетом всех туров в целом.

Тур 5 (заключительный)

Ответ с обоснованием отправьте в редакцию до 10 февраля 2015 года по адресу: 121165, Москва, ул. Киевская, д. 24, “Первое сентября”, “Информатика” или по электронной почте: vmi@1september.ru.

Задача 5.1

Три вора — Трус, Балбес и Бывалый, каждый с несколькими баулами, — хотят переpravиться че-

рез реку. Известно, что Трус обворует любой баул Балбеса, если баул останется без присмотра кого-нибудь из остальных. Так же Балбес обворует баул Бывалого, а Бывалый — баул Труса. Есть трехместная лодка, место занимает человек или баул. Грести может только Трус. Как им всем переправиться и перевезти баулы, чтобы никто никого не обворовал? (На пустынном берегу баулы в безопасности.)

Задача 5.2

К переправе подошли царевна Несмеяна и строй из семи богатырей. Лодка одна, в ней могут плыть двое или трое (в одиночку плыть нельзя). Каждый согласен плыть только вдвоем с другом или втроем с двумя друзьями. Какое наибольшее число из них сможет переправиться, если каждые двое рядом стоящих богатырей — друзья, Несмеяна дружит с ними всеми, кроме среднего из богатырей, а все остальные пары не дружат?

Ответы с обоснованием отправьте в редакцию до 10 февраля.

Головоломка “Получить верное равенство”

Напомним, что следовало установить, как в неверном равенстве $9 + 1 = 8$, составленном из спичек, переложив одну спичку, сделать так, чтобы равенство стало верным.

Решение

Снять спичку с цифры 8 (получится цифра 9) и переложить ее в первую цифру 9 (получится 8):

$$8 + 1 = 9.$$

Заметим, что вариант ответа, приводящий к равенству $9 - 1 = 8$, не подходит, так как в условии шла речь не о снятии, а о перекладывании спички.

Ответы представили:

— Абдувахидова Алина, Абдувахидова Софья, Викторов Даниил, Галкина Эвелина, Карпов Иван, Милушкин Дмитрий, Никулин Даниил, Строкин Константин, Филиппова Наталья и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Авдеев Андрей и Яковлева Мария, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Алексеева Полина, Волкова Алина и Тимошенкова Ульяна, Смоленская обл., г. Демидов, школа № 1, учитель **Кордина Н.Е.**;

— Багрова Анастасия, г. Пионерский Калининградской обл., школа № 2, учитель **Багрова О.А.**;

— Васильев Кирилл и Жабин Игорь, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Градова Валентина и Землянская Надежда, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

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

Конкурс № 112

В качестве заданий данного конкурса предлагаем выполнить задания для самостоятельной работы, предложенные в статье “Еще две замечательные кривые” в рубрике “Семинар”. Срок представления ответов — 10 февраля.

Конкурс № 113

В качестве заданий этого конкурса предлагаем выполнить задания для самостоятельной работы, предложенные в статье “Фальшивые монеты, отравленное вино и... системы счисления”. Можно выполнять не все задания. Срок представления ответов — 1 марта.

Пожалуйста, четко укажите в ответах свои фамилию и имя, населенный пункт, номер и адрес школы, фамилию, имя и отчество учителя информатики.

— Жарова Екатерина и Птицына Мария, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Зеленцова Виктория, Ломтева Елизавета и Окунева Екатерина, средняя школа села Ошминское Тоншаевского р-на Нижегородской обл., учитель **Попов Г.Н.**;

— Калина Екатерина, Лукина Светлана и Хомутова Елена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Кляузер Адам, г. Орел, лицей № 4 им. Героя Советского Союза Г.Б. Злотина, учитель **Чапкевич И.М.**;

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

— Минченко Валерия, средняя школа села Дохновичи Брянской обл., учитель **Клопова Е.В.**;

— Новиков Филипп и Цыплаков Евгений, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Огнев Руслан, г. Орел, лицей № 4 им. Героя Советского Союза Г.Б. Злотина, учитель **Чапкевич И.М.**

Задача “Профессор и студенты”

Напомним, что требовалось установить, какое количество студентов могло присутствовать на лекции, чтобы из шести записанных на доске утверждений профессора четыре оказались бы неверными.

Ответ

Условию удовлетворяют три значения: 10, 20 и 30.

Правильный ответ прислали:

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

— Байкова Римма, Дубинина Анна, Иванова Виолетта и Левченко Ирина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

— Владимиров Виталий, Владимирова Снежана, Семенова Екатерина и Яковлева Анастасия, основная школа села Именево, Республика Чувашия, Красноармейский р-н, учитель **Тимофеева И.А.**;

— Данильчук Сергей, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Карданова Аминат, Коломина Нонна, Медяникова Аделина, Остроухова Валерия и Уткина Ксения, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

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

— Лютикова Светлана, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

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

— Новиченко Владислав и Шатова Елена, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Филиппенко Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Задача “Верблюды и дядя Федор”

Решение

Так как пять верблюдов плюнули по три раза каждый, то всего было сделано 15 плевков. Каждый из пяти верблюдов получил по два плевка, то есть в верблюдов попало 10 плевков. Следовательно, дяде Федору “достались” остальные $15 - 10 = 5$ плевков.

Ответ: 5 плевков.

Правильные ответы представили:

— Абдувахидова Алина, Абдувахидова Софья, Галкина Эвелина, Карпов Иван, Милушкин Дмитрий, Филиппова Наталья и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Авдеев Андрей и Яковлева Мария, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

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

— Васильев Кирилл и Жабин Игорь, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Владимиров Виталий, Владимирова Снежана, Семенова Екатерина и Яковлева Анастасия, основная школа села Именево, Республика Чувашия, Красноармейский р-н, учитель **Тимофеева И.А.**;

— Гаронин Лев, Новиков Филипп и Цыплаков Евгений, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Градова Валентина и Землянская Надежда, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Донцов Юрий и Шаурина Анастасия, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Жарова Екатерина и Птицына Мария, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Зеленцова Виктория и Окунева Екатерина, средняя школа села Ошминское Тоншаевского р-на Нижегородской обл., учитель **Попов Г.Н.**;

— Калина Екатерина, Лукина Светлана и Хомутова Елена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**

Задача “Четыре друга”

Напомним условие: “Четыре друга — Миша, Коля, Саша и Дима — проживали по следующим адресам: Лесная, 37; Цветочная, 25; Лесная, 25. Узнайте, в каком доме и на какой улице жил каждый из мальчиков, если известно, что Миша и Коля жили на одной улице, Саша и Коля жили в домах с одинаковыми номерами, а Миша и Дима были родными братьями”.

Решение

Так как на одной улице жили трое ребят — Миша, Дима (они братья) и Коля, — то четвертый мальчик (Саша) жил на Цветочной улице в доме 25. Тогда адрес Коли — Лесная, 25. По оставшемуся адресу жили братья Миша и Дима.

Ответ (адреса ребят): Коля: Лесная, 25; Саша: Цветочная, 25; Миша: Лесная, 37; Дима: Лесная, 37.

Правильные ответы прислали:

— Абакумова Владислава, Донцов Юрий, Куликова Елена и Шаурина Анастасия, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Абдувахидова Алина, Абдувахидова Софья, Викторов Даниил, Галкина Эвелина, Карпов Иван, Кротов Олег, Милушкин Дмитрий, Миноцкий Ян, Семенова Милена, Филиппова Наталья и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

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

— Алексеева Полина, Волкова Алина и Тимошенкова Ульяна, Смоленская обл., г. Демидов, школа № 1, учитель **Кордина Н.Е.**;

— Барина Мария, Лосева Нина и Храмова Анна, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Батурин Илья, Дикий Иван, Мисюра Алексей, Пак Александра, Панасенко Дарья, Пантелюк Руслан, Приходько Геннадий, Сысоев Александр и Шикалович Ростислав, г. Пионерский Калининградской обл., школа № 2, учитель **Багрова О.А.**;

— Бирюкова Елена, Васильев Кирилл и Жабин Игорь, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Владимиров Виталий, Владимирова Снежана, Семенова Екатерина и Яковлева Анастасия, ос-

новная школа села Именево, Республика Чувашия, Красноармейский р-н, учитель **Тимофеева И.А.**;

— Гаронин Лев, Новиков Филипп, Тимошина Наталья и Цыплаков Евгений, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Градова Валентина и Землянская Надежда, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Зеленцова Виктория, Ломтева Елизавета, Окунева Екатерина и Полякова Наталия, средняя школа села Ошминское Тоншаевского р-на Нижегородской обл., учитель **Попов Г.Н.**;

— Иванова Виолетта, Казакова Валерия, Левченко Ирина и Якушкина Яна, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

— Калина Екатерина, Лукина Светлана, Макаров Сергей и Хомутова Елена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Карданова Аминат, Коломина Нонна, Медяникова Аделина, Остроухова Валерия и Уткина Ксения, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6, учитель **Рябченко Н.Р.**;

— Кошелева Анастасия, Куминская средняя школа, Тюменская область, Ханты-Мансийский автономный округ — Югра, Кондинский р-н, учитель **Шишигина О.В.**;

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

— Никонова Валентина, Куминская средняя школа, Тюменская область, Ханты-Мансийский автономный округ — Югра, Кондинский р-н, учитель **Шишигина О.В.**

Числовые ребусы в троичной системе. Часть 1

Приведем решение ребуса 3.

$$\begin{array}{r} + \quad * \\ + \quad * \\ \hline \mathbf{B} \quad \mathbf{B} \end{array}$$

Так как цифра **В** не может быть равна нулю, то **В = 1** или **В = 2**. Второй вариант не подходит, поскольку число 22 не может быть суммой двух однозначных троичных чисел (максимальная такая сумма равна $2 + 2 = 11$). Анализ показывает, что последнее равенство и является единственным искомым решением.

Ответы представили:

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

— Букин Владислав и Хомутова Елена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Владимирова Виталий, Владимирова Снежана, Семенова Екатерина и Яковлева Анастасия, ос-

новная школа села Именево, Республика Чувашия, Красноармейский р-н, учитель **Тимофеева И.А.**;

— Гаронин Лев, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Лютикова Светлана, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Марков Петр, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

— Филиппенко Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Хрекова Мария, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

Числовой ребус “ОДЕЯЛО из ЛОСКУТОВ”

Решение

Запишем ребус “в столбик”:

$$\begin{array}{r} \text{Л О С К У Т} \\ + \text{Л О С К У Т} \\ \hline \text{Л О С К У Т} \\ \hline \text{О Д Е Я Л О} \end{array}$$

Исследуем возможные значения цифры **Т** и соответствующие значения **О** (по последнему справа разряду), значения **3О** (по второму слева разряду) и **Л** (по первому слева разряду):

Т	О	3О	В “уме” из разряда десятков тысяч переходит	Л	Допустим ли вариант?
1	3	9	0	1	Нет (Т = Л)
2	6	18	1	5	Да
3	9	27	2	9	Нет (О = Л)
4	2	6	0	4	Нет (Т = Л)
5	5				Нет (Т = О)
6	8	24	2	2	Да
7	1	3	0	7	Нет (Т = Л)
8	4	12	1	1	Да
9	7	21	2	5	Да

Проверка четырех допустимых вариантов показывает, что возможен только вариант при **Т = 6**, то есть решением ребуса является: $283\ 176 \times 3 = 849\ 528$.

Правильное решение прислали:

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

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

— Владимирова Виталий, Владимирова Снежана, Семенова Екатерина и Яковлева Анастасия, основная школа села Именево, Республика Чувашия, Красноармейский р-н, учитель **Тимофеева И.А.**;

— Гаронин Лев, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Дикий Данил, г. Пионерский Калининградской обл., школа № 2, учитель **Багрова О.А.**;

— Лютикова Светлана, средняя школа поселка Осинка, Алтайский край, учитель **Евдокимова А.И.**;

— Марков Петр, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Никонова Валентина, Куминская средняя школа, Тюменская область, Ханты-Мансийский автономный округ — Югра, Кондинский р-н, учитель **Шишигина О.В.**;

— Филиппенко Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Головоломка “Суперлабиринт «Термины информатики»”

Напомним, что требовалось по приведенным комментариям к 66 терминам, связанным с информатикой и компьютерами, вписать соответствующие термины в таблицу так, чтобы первая буква очередного термина совпадала с последней буквой предыдущего.

Ответы

1. Компьютер. 2. Растр. 3. Редактор. 4. Раскладка. 5. Адаптер. 6. Разряд. 7. Дискковод. 8. Документ. 9. Трекбол. 10. Лента. 11. Адрес. 12. Слово. 13. Откат. 14. Таблица. 15. Абзац. 16. Цикл. 17. Лэптоп. 18. Пиксел(ь). 19. Линейка. 20. Архитектура. 21. Анимация. 22. Язык. 23. Копия. 24. Ярлык. 25. Клавиатура. 26. Ада. 27. Алфавит. 28. Текстура. 29. Алгоритм. 30. Модем. 31. Модуляция. 32. Ячейка. 33. Абонент. 34. Триггер. 35. Регистр. 36. Ряд. 37. Дюйм. 38. Масштаб. 39. Браузер. 40. Разъем. 41. Монитор. 42. Рамка. 43. Атрибут. 44. Текст. 45. Телекс. 46. Сайт. 47. Трафик. 48. Кэш. 49. Шаблон. 50. Набор. 51. Реверс. 52. Сервер. 53. Репитер. 54. Развертка. 55. Аналог. 56. График. 57. Корзина. 58. Арбитраж. 59. Журнал. 60. Логика. 61. Апплет. 62. Таймер. 63. Режим. 64. Матрица. 65. Абак. 66. Карандаш.

Ответы представили:

— Астахов Максим, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

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

— Баранов Владимир, средняя школа поселка Осинка, Алтайский край, учитель **Евдокимова А.И.**;

— Власова Ольга и Дружинина Анастасия, средняя школа поселка Ерофей Павлович, Амурская обл., Сквородинский р-н, учитель **Краснёнкова Л.А.**;

— Гаронин Лев, Диков Андрей и Яшина Елена, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Марков Петр, Назаренко Иван и Шукина Елена, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

— Сосновский Артем, средняя школа села Дожновичи Брянской обл., учитель **Клопова Е.В.**;

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

— Хрекова Мария, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

Решения японских головоломок “судоку”, опубликованных в сентябрьском выпуске “В мир информатики”, прислали:

— Астахов Максим, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

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

— Деревянченко Дарья, Тихомирова Елизавета и Тюфелева Мария, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Кошелева Анастасия, Куминская средняя школа, Тюменская область, Ханты-Мансийский автономный округ — Югра, Кондинский р-н, учитель **Шишигина О.В.**;

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

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

— Панасенко Дарья и Пантелюк Руслан, г. Пионерский Калининградской обл., школа № 2, учитель **Багрова О.А.**;

— Рябиков Григорий, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

Решение задачи “Статистика”, анализ которой был начат в рубрике “Крепкий орешек”, представили:

— Астахов Максим, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Зеленцова Виктория и Окунева Екатерина, средняя школа села Ошминское Тоншаевского р-на Нижегородской обл., учитель **Попов Г.Н.**;

— Марков Петр, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

Спасибо всем!

ЦИФРОВОЙ МИР

Цена округления

Как известно, дроби переводятся из десятичной системы в двоичную не всегда точно. Погрешности, возникающие при переводе, могут привести к неприятным последствиям не только в финансовых вычислениях. Например, рассказывают следующую историю [Гашков С.Б. Запоминальная компьютерная арифметика. Математика и искусство счета на компьютерах и без них. М.: URSS, 2012]. В 1991 году во время войны в Персидском заливе американская ракета “Пат-

риот” не смогла перехватить иракскую ракету “Скад” из-за ошибки в определении времени. При вычислениях нужно было умножать время, отсчитываемое тактовым генератором бортового компьютера, на 1/10. В компьютере это число представлялось 24-разрядной двоичной дробью. Естественно, с ошибкой, хотя и маленькой. За сто часов работы компьютера ошибка в определении времени достигла 0,34 сек. Так как скорость ракеты “Скад” составляла 1700 м/сек., то ошибка в определении положения иракской ракеты достигла 500 м. Естественно, ракету перехватить не удалось, и она попала в казарму американских солдат. Погибло 28 человек...

Работа с данными типа “время” в электронной таблице Microsoft Excel

Предисловие

В разделе “В мир информатики” уже отмечалось (см. [1]), что в электронной таблице Microsoft Excel и других подобных программах можно работать не только с числами и текстами, но и с датами. Даты можно сравнивать между собой, складывать и вычитать, а также использовать в других вычислениях. Например, можно вычислить число дней между двумя датами, определить, какой день недели приходился на ту или иную дату, и т.п.

В данной статье представлены задачи на обработку данных еще одного типа — “время”.

Для решения приведенных задач необходимо узнать особенности внутреннего представления времени в программе Excel. Это можно сделать на примере следующего задания.

Введите в ячейки A1, A2 и A3 значения времени, равные, соответственно, 12 часам, 6 часам и 9 часам, то есть 12:00, 6:00 и 9:00. Скопируйте диапазон ячеек A1:A3 в B1:B3. Измените формат представления данных в диапазоне на **Общий**. Проанализируйте появившиеся значения. Что вы можете сказать о внутреннем представлении времени в программе Microsoft Excel? Что представляет собой разность двумя значениями времени, представленная в общем формате?

Обращаем внимание на то, что, как правило, для того чтобы ввести в ячейку время, следует указать число часов, число минут и число секунд через двоеточие (10:12:45 и т.п.).

Задачи

1. В ячейке B2 получить число 0,75 (указанное число ни в одну из ячеек не вводить).

	A	B	C
1			
2		18.00	
3			

2. В ячейках B2 и B3 получить число 0,65659722 (указанное число ни в одну из ячеек не вводить).

	A	B	C
1			
2		15:45:30	
3		3:45:30 PM	
4			

3. В две ячейки введите два разных значения времени. Определите, сколько минут прошло между этими моментами времени.

4. В две ячейки введите два разных значения времени. Определите, сколько часов прошло между этими моментами времени.

5. В две ячейки введите два разных значения времени. Определите, сколько секунд прошло между этими моментами времени.

6. В одной из ячеек укажите некоторое время суток и определите:

- сколько часов осталось до конца суток;
- сколько минут осталось до конца суток;
- сколько секунд осталось до конца суток.

7. В одной из ячеек укажите некоторое время и определите длительность отрезка времени (в виде n час, m мин., s сек.), оставшегося до конца дня.

8. В одной из ячеек указано время прибытия поезда на станцию по расписанию, в другой — фактическое время прибытия. Определите, на сколько минут опоздал поезд, если он прибыл на станцию в тот же день, в который должен был прибыть по расписанию.

9. Занятие продолжается с 9:00 до 11:30 без перерыва. Определите продолжительность занятия в минутах. Сколько академических часов оно продолжается (академический час равен 45 мин.)?

10. В ячейках B2 и B3 указано значение одного и того же момента времени, но в разных часовых поясах.

Определите, на сколько часов отличается время в этих часовых поясах (ответ получите в виде числа). Принять, что указанные в ячейках B2 и B3 значения относятся к одному и тому же дню.

11. Известно время вылета самолета из Москвы и время прибытия его в аэропорт назначения (оба значения — по московскому времени). Определите продолжительность рейса, если самолет прибыл в аэропорт назначения в тот же день.

12. Известно время отправления поезда из Москвы и время прибытия его на конечную станцию. Определите время нахождения поезда в пути. Следует учесть, что на конечную станцию поезд может прибыть на следующий день (при этом время нахождения поезда в пути меньше суток).

13. На обложке компакт-диска с музыкальными записями указано время проигрывания каждой записи в минутах и секундах: 6:50, 4:20 и 5:45. Вычислите общее время звучания в секундах.

14. В ячейках B2:B10 указано время проигрывания каждой музыкальной записи в минутах и секундах в виде ММ:СС, где ММ — число минут, СС — число секунд. Вычислите общее время звучания в минутах.

15. На листе указаны время прибытия поездов по расписанию и величина опоздания каждого поезда:

	A	B	C	D
1	№ поезда	Время прибытия по расписанию	Величина опоздания, мин.	
2	41	20:21	10	
3	15	21:07	3	
4	22	22:31	4	
5	256	21:06	2	
6	...			

Необходимо определить время фактического прибытия на конечную станцию (столбец D).

16. Известны результаты, показанные лыжниками на каждом из четырех этапов Кубка мира по лыжам:

№	Фамилия	1-й этап	2-й этап	3-й этап	4-й этап
1					
2					
3					
...					

Время прохождения каждого этапа записано в таблице в формате *время* в виде ЧЧ:ММ:СС. Для каждого спортсмена определите общее время, затраченное на все этапы соревнований. Известно, что для каждого лыжника искомое значение меньше суток.

17. В одну из ячеек введите значение времени, соответствующее моменту времени 19 час. 25 мин. 10 сек. Определите время через 0,1 суток после указанного. Результат представьте в формате *время* (в виде ЧЧ:ММ:СС).

18. В одну из ячеек введите значение времени, соответствующее моменту времени 1 час 33 мин. 55 сек. Определите время через 22,5 часа после указанного. Результат представьте в формате *время* (в виде ЧЧ:ММ:СС).

19. В одну из ячеек введите значение времени, соответствующее моменту времени 13 час. 3 мин. 5 сек. Определите время через 481 минуту после указанного. Результат представьте в формате *время* (в виде ЧЧ:ММ:СС).

20. В одну из ячеек введите значение времени, соответствующее моменту времени 9 час. 15 мин. 30 сек. Определите время через 5000 секунд после указанного. Результат представьте в формате *время* (в виде ЧЧ:ММ:СС).

21. В одной из ячеек указано время прибытия поезда на станцию по расписанию, в другой — на сколько минут опоздал поезд (в виде числа). Определите фактическое время прибытия поезда. Результат представьте в формате *время* (в виде ЧЧ:ММ).

22. В одной из ячеек указано время прибытия поезда на станцию по расписанию, в другой — величина отклонения фактического времени прибытия от времени прибытия по расписанию (в минутах, в виде числа). Определите фактическое время прибытия поезда. Результат представьте в формате *время* (в виде ЧЧ:ММ). Необходимо учесть, что указанное отклонение может быть и отрицательным (поезд прибыл на станцию раньше положенного времени).

23. На листе представлено расписание движения пригородных поездов:

	A	B	C	D
1	№ поезда	Время отправления	Время в пути	Время прибытия на конечную станцию
2	001	10:21	1:45	
3	015	11:07	2:02	
4	023	12:31	1:04	
5	041	14:06	3:58	
6	047	16:00	3:18	
7	191	18:45	4:29	
8	241	20:10	3:00	
9				

Необходимо определить время прибытия на конечную станцию (столбец D).

24. Предыдущую задачу решите для случая, когда время в пути задано в часах в виде вещественного числа (2,5; 3,15 и т.п.).

25. Оформите на листе расписание школьных звонков:

	A	B	C	D
1	№ урока	Начало урока	Конец урока	Перемена
2	1	8:30	9:15	
3				0:15
4	2	9:20	10:05	
5				0:15
6	3	10:20	11:05	
7				0:15
8	4	11:20	12:05	
9				0:20
10	5	12:25	13:10	
11				0:10
12	6	13:20	14:05	
13				0:15
14	7	14:20	15:05	
15				0:10
16	8	15:15	16:00	
17				

Указания по выполнению:

1. В качестве исходных данных необходимо использовать время начала первого урока, время окончания каждого урока и продолжительность перемен.

2. Время начала второго–восьмого уроков определить по формулам.

3. Значения в ячейках B6, B8, B10, B12, B14 и B16 получить путем копирования в эти ячейки формулы, введенной в ячейку B4.

26. Оформите на листе расписание школьных звонков:

	A	B	C	D
1	№ урока	Начало урока	Конец урока	Перемена
2	1	8:30		
3				0:15
4	2			
5				0:15
6	3			
7				0:15
8	4			
9				0:20
10	5			
11				0:10
12	6			
13				0:15
14	7			
15				0:10
16	8			
17				
18	Продолжительность урока:			
19				

Продолжительность урока (в минутах) должна задаваться в ячейке B18.

27. На листе представлены сведения о времени отправления поездов по расписанию и о величине задержки отправления, а также о времени прибытия их на конечную станцию по расписанию и о величине опоздания:

	A	B	C	D	E
1	№ поезда	Время отправления		Время прибытия	
2		По расписанию	Задержка отправления, мин.	По расписанию	Величина опоздания, мин.
3	25	9:25	5	23:40	10
4	71				
5	...				

Для каждого поезда нужно определить время нахождения в пути. Принять, что каждый поезд прибыл на конечную станцию в день отправления.

Решения, пожалуйста, присылайте в редакцию (можно решать не все задачи). Авторы лучших ответов будут награждены дипломами.

Литература

1. Работа с датами в электронной таблице Microsoft Excel. / “В мир информатики” № 202 (“Информатика” № 11/2014).

ЭТО ПОЛЕЗНО ЗНАТЬ

Правила этикета электронной почты

Соблюдение приведенных ниже правил общения по электронной почте делает ведение переписки более приятным и эффективным.

1. Следите за тоном письма. Текст требует больше вежливых слов и точности выражений, чем устная речь, в которой смысловые оттенки отчасти передаются интонацией.
2. Избегайте сокращений, специальных терминов и жаргонных слов, если вы не уверены, что они понятны адресату — человеку, которому вы пишете.
3. Не злоупотребляйте прописными буквами (написанный так текст воспринимается как крик). То же относится к восклицательным и вопросительным знакам.
4. Старайтесь не допускать ошибок в правописании. Если вы не уверены в написании тех или

иных слов, используйте справочники, учебники, специальные программы и т.п.

5. Всегда указывайте тему письма. Она помогает адресату понять, о чем ваше письмо. Старайтесь ее формулировать максимально четко и понятно. Неудачная формулировка: “Пожалуйста, посмотрите”. Гораздо лучше “Реферат по истории”.
6. Обязательно в конце письма укажите свое имя (имя и фамилию). “Пустые” письма — без текста и указания имени, а только с одним вложением — признак неуважения к адресату.
7. Внимательно просматривайте текст своего письма перед тем, как его отправить. В отправленном письме уже ничего не исправить.
8. По возможности сразу же отвечайте на полученные письма. Если вы не можете написать подробное ответное письмо, сообщите написавшему вам, что письмо получено и что по существу вы ответите позже. Иногда достаточно сообщить: “Татьяна, здравствуй. Фотографии получил. Спасибо. Андрей” или т.п.



ПРИИЫ

В будущем неграмотным будет считаться не тот, кто не умеет читать, а тот, кто не умеет обучаться.

Э.Тофлер

Человек — это дробь, в числителе которой — его действительные возможности, а в знаменателе — его мнение о себе.

Лев Толстой

Древний японский метод вычисления произведений

Евгений Яковлев,
ученик школы № 1360 г. Москвы
(научный руководитель — учитель алгебры
и информатики О.Б. Богомолова)

В Японии еще в древности существовал оригинальный, удобный и наглядный способ вычисления произведений целых чисел. В других странах, в том числе в России, этот метод неизвестен. Однако с учетом предложенного в данном проекте усовершенствования японский метод умножения может с успехом быть использован для выполнения математических расчетов теми школьниками, которые испытывают затруднения при выполнении вычислений произведений чисел стандартным методом умножения в столбик.

Целью моего проекта было продемонстрировать возможность применения модифицированного японского метода вычисления произведений многозначных целых чисел для наглядного выполнения данной математической операции.

Задачи проекта

1. Исследовать принципы выполнения вычислений по японскому методу умножения целых чисел.
2. Продемонстрировать возможность применения этого метода применительно к целым числам различной длины.
3. Модифицировать японский метод умножения применительно к целым числам с нулем.

Умножение двухзначных чисел

Суть японского метода умножения заключается в правильном построении линий и отсечении нужных скоплений точек их пересечения.

Для начала рассмотрим умножение двухзначных чисел с помощью японского метода умножения.



Рис. 1

Рассмотрим вычисление произведения 13×42 . Берем первую цифру первого числа, т.е. единицу, и рисуем наклонную линию (как показано красным цветом на рисунке). Далее проделываем то же самое с остальными цифрами, но цифры следующего числа рисуем под другим наклоном (желательно — перпендикулярно предыдущим цифрам, для большей наглядности и легкости решения в дальнейшем).

Итак, мы представили все цифры чисел в виде линий. Далее мы смотрим на получившийся рисунок, начинаем отсекают скопления точек пересечения линий и подсчитываем их — с каждого края по одному скоплению (как показано на рис. 1).

Потом мы подсчитываем, сколько пересечений имеется в центральной (оставшейся) части рисунка. Определив количество пересечений, записываем это число рядом или под этими скоплениями: в данном случае справа у нас получилось 6 пересечений, в центре — 14, слева — 4.

Проверяем, получились ли числа больше или равно 10 (в нашем случае такое число есть). В японском методе начинают смотреть цифры *справа налево*, поэтому первое число — 6 и оно меньше 10, его мы оставляем без изменения. Число 14 — больше 10: в таких случаях всегда оставляют количество единиц, а количество десятков переносят в следующее число. Следовательно, из числа 14 мы получаем число 4, а единица прибавляется к следующему числу 4 — получаем число 5 (оно меньше 10, поэтому оставляем его без изменения). И, наконец, последний пункт: в качестве ответа записываем получившиеся числа *слева направо*. В итоге получилось число 546, которое и является ответом.

Умножение трехзначных чисел

Теперь рассмотрим умножение трехзначных чисел.

Рассмотрим пример: 125×231 . Так же, как и при умножении двухзначных чисел, сначала требуется превратить все цифры чисел в линии. В итоге должно получиться такое же изображение, как на рис. 2. Далее аналогично начинаем

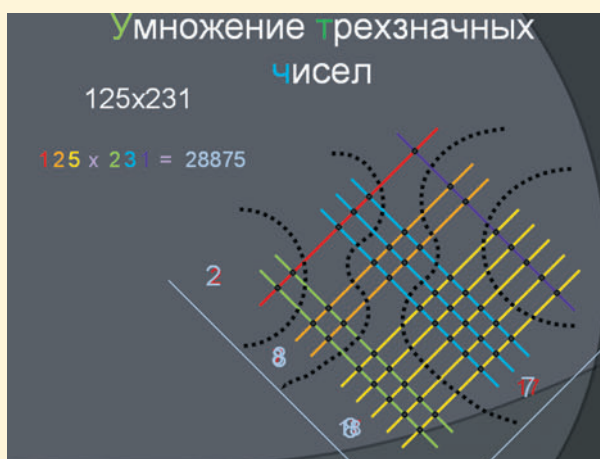


Рис. 2

отсекать крайние скопления: сначала по одному скоплению по бокам, а затем — по два. У нас останется центральная часть рисунка, которая будет состоять уже из трех скоплений пересечений. После этого мы опять подсчитываем количество точек пересечения в каждом из отмеченных нами скоплений и записываем их под выделенными скоплениями.

Если где-то получились числа больше 10, то мы делаем то же самое, что и в случае умножения двухзначных чисел, а именно переносим десятки в следующее число (не забываем, что на данном этапе нужно проверять числа строго *справа налево*). Получившиеся числа записываем, наоборот, *слева направо*. В итоге получаем число 28 875, которое и является ответом.

Умножение двухзначных чисел с нулем

Когда японцы создавали этот метод умножения, они не использовали цифру 0 в своих вычислениях. Мне показалось это странным, и я решил модифицировать этот метод, чтобы умножение с использованием нуля было возможно.



Рис. 3

Как выяснилось после решения большого количества примеров, достаточно цифру 0 в числе изображать не обычной, а *воображаемой* линией. То есть мы подразумеваем, что у нас есть линия, но, когда дело доходит до подсчета точек пересечения, мы не учитываем точки, которые образуются при пересечении с данной воображаемой линией.

Рассмотрим пример: 13×40 . Делаем все как и раньше, но 0 изображаем пунктирной линией (как показано на рис. 3). Далее, отсекая и подсчитывая точки пересечения, мы не включаем в подсчет точки на пересечениях с “нулевой” линией.

Остальное делается аналогично предыдущим примерам. В ответе получаем число 520.

Умножение трехзначных чисел с нулем

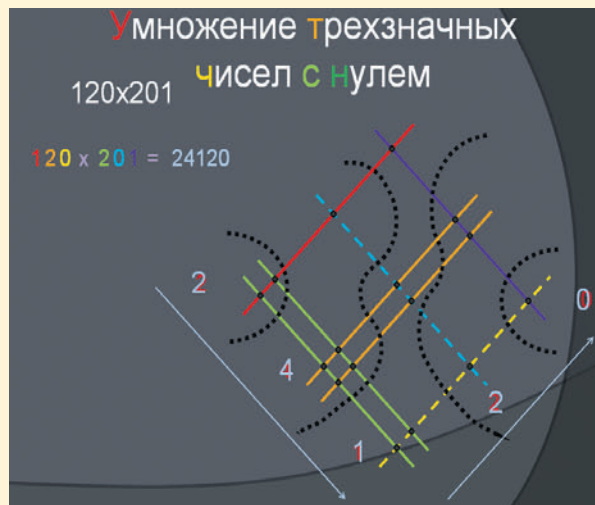


Рис. 4

С трехзначными числами — то же самое. Рассмотрим пример: 120×201 (см. рис. 4). Цифру 0 мы представляем как воображаемую линию, а остальные цифры — обычными линиями. Проводим подсчеты аналогично предыдущим примерам. В итоге получаем число 24 120, что и является ответом в данном примере.

Выводы

1. Таким образом, японский метод умножения в отличие от традиционного умножения в столбик гораздо более нагляден. Это поможет тем, кто испытывает трудности в выполнении вычислений.

2. Предлагаемый модифицированный японский метод отличается от первоначального тем, что в нем добавлена возможность использования цифры 0 (которой у японцев не существовало в те времена, когда они создавали этот метод умножения).

Предлагаю читателям самостоятельно определить, используя описанный метод, произведения:

- 1) 15×35 ;
- 2) 463×271 ;
- 3) 18×70 ;
- 4) 450×30 .

От редакции

1. Ответы на предложенные задания (с описанием методики вычислений), пожалуйста, присылайте в редакцию.

2. Дорогие ребята! Присылайте также в редакцию результаты своих проектов по информатике. Лучшие работы будут опубликованы.

Проверка факта двоичного представления чисел в компьютере

Из уроков информатики известно, что в компьютере данные хранятся и обрабатываются в двоичной системе счисления. А действительно ли это так? В собственных программах, в электронной таблице Microsoft Excel и в других случаях мы вводим десятичные числа и получаем на экране десятичный результат. Можем ли мы быть уверенными, что компьютер производил действия именно в двоичной системе? Как это проверить?

Чтобы убедиться в этом, давайте проведем небольшой эксперимент.

Наберем несложную программу, которая на школьном алгоритмическом языке имеет вид:

```
алг Проверка_двоичного_представления_чисел
нач цел m, n, i, вещ h, sum
  m := 128
  вывод нс, " n          Разность "
  нц для n от m - 1 до m + 1
    h := 1/n
    sum := 0
    нц для i от 1 до n
      sum := sum + h
    кц
  вывод нс
  вывод n, " ", sum - 1
кц
кон
```

Прокомментируем ее. “Наружный” оператор цикла с параметром n обеспечивает трехкратное исполнение программы для значений n от $m - 1$ до $m + 1$ (в нашем случае это 127, 128 и 129). Нетрудно заметить, что речь идет о числе, являющемся точной степенью двойки (т.е. 128) и двух “соседей”. Далее для каждого из трех чисел n раз происходит суммирование величины $1/n$ и печатается

разность, показывающая, насколько полученное число (по сути, равное $n * 1/n$) отличается от 1. Запустив программу, мы увидим на экране следующий результат

n	Разность
127	...
128	0.0000000000E+00
129	...

Если вы подготовите аналогичную программу на известном вам языке программирования, то увидите, что только для числа $1/128$ результат тождественно равен нулю. А вот для двух других чисел результат получился хоть и маленький, но ненулевой! Почему? Все дело в том, что число $1/128$ представляется в двоичной системе точно, а числа $1/127$ и $1/129$ — нет.

Может быть, это случайность? Проверьте это для других значений m , например, 512 или 2048. В то же время при $m = 126$ все три результата будут отличными от нуля, т.е. степени двойки действительно являются “особенными”.

Вывод. Полученные результаты можно объяснить единственным способом: компьютер действительно хранит числа и производит вычисления в двоичной системе. Именно поэтому только для степеней двойки ему удастся получить точный результат.

Задание для самостоятельной работы

Определите значения разности, указанной в комментариях к программе, для чисел 127, 129 и троек чисел, “в центре” которых находятся числа 512, 2048 и 126. Определите также (без использования программы), как в двоичной системе выглядит число $1/128$. Ответы, пожалуйста, присылайте в редакцию. Фамилии всех приславших правильные ответы будут опубликованы.

Литература

1. Еремин Е.А. Популярные лекции об устройстве компьютера. СПб.: “БХВ-Петербург”, 2003.

ЗАДАЧНИК

Неутомимая гусеница

В публикации [1] рассказывалось о гармониче-

ских числах — суммах вида $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} = \sum_{k=1}^n \frac{1}{k}$

при целом $n \geq 0$, а в статье [2] решалась задача с их использованием, связанная с игральными картами. Вот еще одна задача, в которой мы имеем дело с гармоническими числами.

Гусеница ползет по куску резины, стремясь достичь противоположного конца. Ползет она со скоростью 1 см в минуту. Кусок имеет длину 7 см и может растягиваться до любой длины. Через минуту вы вытягиваете резину еще на 7 см (то есть она

стала 14 см в длину). Гусеница прочно держится на поверхности и продолжает двигаться, пока вы тянете резину. Она ползет все с той же скоростью. Еще через минуту вы снова вытягиваете резину так, что ее первоначальная длина утраивается (то есть она становится равной 21 см). Гусеница продолжает ползти, а вы продолжаете каждую минуту тянуть резину, в четвертый раз уже удлив ее в четыре раза. Доберется ли гусеница когда-нибудь до конца резины, ведь каждую минуту длина последней существенно увеличивается?

Ответ, оказывается, положительный! Чтобы подтвердить это, прежде всего заметим, что существует формула для вычисления количества слагаемых в сумме, приведенной в начале статьи, при котором достигается некоторое заданное число. Предполо-

жим, нужно узнать, сколько слагаемых (их называют членами гармонического ряда) нужно взять, чтобы получить в сумме N . Возьмите число 2,71828 и умножьте его само на себя N раз, а затем разделите ответ на 1,781. Значение, полученное в результате, и будет числом членов гармонического ряда, необходимым для получения в сумме N . Итак, нужная нам формула:

$$\frac{2,71828^N}{1,781}$$

Теперь вернемся к гусенице и разберемся, что же происходит при ее движении. За первую минуту наша “путешественница” проползла 1 см, то есть $1/7$ длины пути. В течение следующей минуты она проползает еще 1 см, что составит $1/14$ длины резины. Таким образом, в общей сложности пройдено $1/7 + 1/14$ длины резины, что можно переписать в виде: $(1 + 1/2)/7$. После удлинения резины до 21 см гусеница преодолет еще 1 см, что составит $1/21$ часть длины. Теперь гусеница в общей сложности проползла $(1 + 1/2 + 1/3)/7$ часть общего пути.

Нетрудно понять, что за t минут гусеница проползет

$$(1 + 1/2 + 1/3 + 1/4 + \dots + 1/t)/7$$

часть пути. Здесь явно просматривается гармонический ряд. Значит, чтобы приведенное только что выражение стало равно $7/7 = 1$ (то есть был достигнут конец резины), нужно, чтобы гармоническое число в скобках было равно 7. Вспомнив о формуле для расчета суммы слагаемых, мы можем сказать, что гусеница доберется до “финиша”

через $t = \frac{2,71828^7}{1,781}$ минут. Для расчетов можно

использовать электронную таблицу Microsoft Excel или разработать компьютерную программу. В первом случае формула для расчета такая:

$$=EXP(7)/1.781$$

В языках программирования для расчета значений $2,71828^n$, как правило, имеется функция exp ($exp(n)$).

Расчеты показывают, что в рассматриваемом случае гусеница пройдет весь путь примерно за 10 часов 26 минут, что является неплохим результатом ☺.

Задание для самостоятельной работы

Разработав компьютерную программу (на языке программирования, который вы изучаете) или используя электронную таблицу Microsoft Excel или другую подобную программу, определите, через какое время гусеница доберется до “финиша”, если первоначальная длина резины равна:

- 10 см;
- 15 см;
- 20 см.

Заметим сразу, что в двух последних случаях время “путешествия” будет измеряться годами...

Литература

1. Гармонические числа. / “В мир информатики” № 96 (“Информатика” № 20/2007).
2. Гармонические числа и... игральные карты. / “В мир информатики” № 105 (“Информатика” № 5/2008).

ЦИФРОВОЙ МИР

Одна мышь на двоих

Английская писательница Вирджиния Вулф считала, что любой романист нуждается в отдельном кабинете для уединения и вдохновения. А вот некоторые компьютерные компании в Кремниевой долине* подвергают сомнению даже необходимость отдельного рабочего стола для программиста.

Метод работы называется “парное программирование”. У сотрудников на двоих один стол и один компьютер. Один из них — “ведущий” — работает на клавиатуре. Другой, “штурман”, следит за дизайном и проверяет компьютер на предмет наличия вирусов. Причем одни работают постоянно с одним и тем же партнером, другие меняют партнеров ежедневно, называя эту практику “случайным партнерством”. Есть также другие виды партнерства, например, партнерство в стиле “пинг-понг”, когда



партнеры взаимодействуют так, будто перекидывают друг другу мяч, как в пинг-понге. Еще один вариант — парное программирование на удаленном доступе, когда программисты делят один экран через Интернет.

Парное программирование впервые приобрело популярность после выхода в 1999 г. книги Кента Бека “Экстремальное программирование”. Из этой книги следует, что компьютерная программа — это

* Кремниевая долина (Силиконовая долина) — комплекс крупных научных и внедренческих фирм в долине Санта-Клара в штате Калифорния (США), к юго-востоку от Сан-Франциско. Этот комплекс в конце XX века стал крупнейшим в мире центром разработки и производства электронной техники и информационных технологий.

продукт, который быстро создается, но потом долго совершенствуется. И быстрее всего с этой задачей справляется команда из двух человек.

У парного программирования есть сторонники в разных фирмах. Они произносят много высокопарных слов о пользе совместной работы, утверждая, что вместе программисты лучше выявляют дорогостоящие ошибки программирования. Кроме того, тратят меньше времени на свободный поиск в Интернете чего-либо, не связанного с работой. “И общение выходит на иной, более глубокий уровень. Мы не используем больше слова. Достаточно хмыкнуть и ткнуть пальцем в экран”, — заявляет

один из убежденных сторонников метода парного программирования.

Но не всегда все так хорошо. Одна из программисток была недовольна своими отношениями с партнером по программированию. И однажды решила с ним об этом поговорить: “Послушай, мне кажется, мы работаем с тобой вразнобой. Можно сказать, что я жму на газ, а ты — на тормоз”.

Ряд фирм позволяет своим инженерам выбирать, хотят они постоянно работать в паре или нет. В результате половина программистов работает в паре на постоянной основе, а еще половина предпочитает объединяться в пары время от времени.

МОДЕЛИРОВАНИЕ

Лисы и зайцы

Как известно, если два биологических вида живут рядом, то они так или иначе взаимодействуют друг с другом в борьбе за существование. Например, один вид служит пищей для другого, как в случае, например, лисиц и зайцев. Зайцы питаются дарами земли, а лисицы поедают зайцев. Математическая модель, описывающая изменение в численности популяций указанных животных, следующая:

$$\begin{aligned} \text{Изм_Зайцев} = & A \times \text{Кол_Зайцев} - B \times \\ & \times \text{Кол_Зайцев} \times \text{Кол_Зайцев} - \\ & - C \times \text{Кол_Зайцев} \times \\ & \times \text{Кол_Лисиц}; \end{aligned}$$

$$\begin{aligned} \text{Изм_Лисиц} = & -D \times \text{Кол_Лисиц} + E \times \\ & \times \text{Кол_Лисиц} \times \text{Кол_Зайцев}, \end{aligned}$$

где *Изм_Зайцев*, *Изм_Лисиц* — соответственно, изменение в численности зайцев и численности лисиц за некоторый период времени;

Кол_Зайцев — имевшееся количество зайцев,

Кол_Лисиц — имевшееся количество лисиц,

A, B, C, D и *E* — величины, зависящие от особенностей окружающей среды.



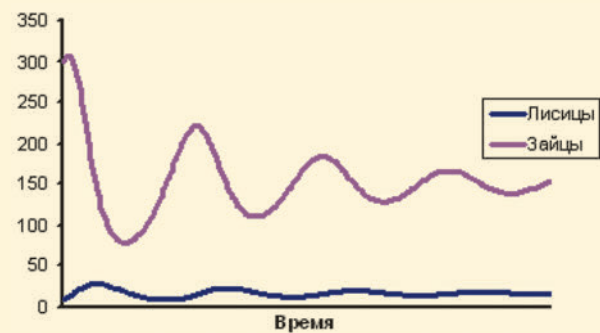
Задания для самостоятельной работы

1. Оформив лист электронной таблицы Microsoft Excel по описанной математической модели, получите ответы на следующие вопросы:

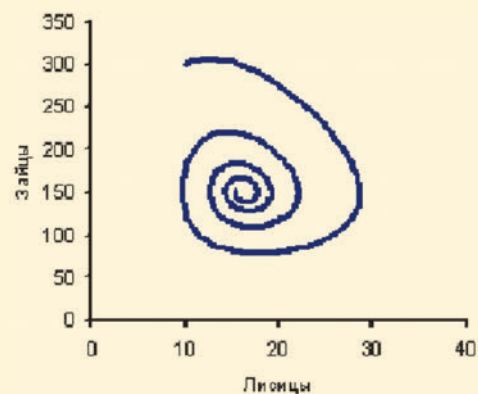
- 1) на каком значении установилась бы численность зайцев, если бы лисиц не было;
- 2) что произошло бы, если бы зайцев вообще не было?

Примите следующие значения параметров модели: $A = 0,04$, $B = 0,00005$, $C = 0,002$, $D = 0,03$, $E = 0,0002$, — а также начальные значения численности зайцев и лисиц, соответственно, 300 и 15.

2. Получите на листе следующий график:



3. Получите на листе следующий график:



4. Прокомментируйте полученные графики. Результаты присылайте в редакцию (можно выполнять не все задания).

Литература

1. Косневски Ч. Занимательная математика и персональный компьютер. М.: Мир, 1987.

2015 год

**Новая эра сотрудничества в образовании –
новые горизонты для учителей России**

**«Просвещение»
и «Первое сентября» –
стратегические партнеры**

Два крупнейших издательства – учебного книгоиздания
и учебной периодики – объединяют усилия для разработки
и продвижения электронных учебников второго поколения.

Апробация учебников – на сайте «Просвещения» и в рамках
общероссийского проекта «Школа цифрового века».

Участвуйте!

prosv.ru

digital.1september.ru