

Учителю информатики: памятные даты и события апреля

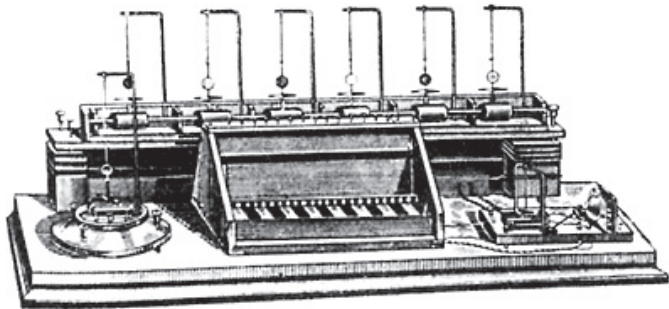
16 апреля 1768 года в городе Ревеле (ныне Таллин) родился Павел Львович Шиллинг (1768–1837) — изобретатель электромагнитного телеграфа.

Первую телеграмму, состоящую из десяти слов, принял лично Павел Львович Шиллинг: первая публичная демонстрация нового телеграфа происходила в октябре 1832 года на его квартире в Санкт-Петербурге.

В 1835 году он с успехом демонстрировал телеграф на съезде естествоиспытателей и врачей в Бонне. Вскоре русское правительство образовало “Комитет для рассмотрения электромагнитического телеграфа” (под председательством морского министра), предложивший Шиллингу установить телеграф в здании Главного Адмиралтейства для длительных испытаний его в условиях, близких к эксплуатационным. Аппараты располагались в противоположных концах длинного здания, провода были проложены частично под землей, частично под водой. Однако из-за неполадок линию так и не ввели в действие. В мае 1837 года Комитет поручил Шиллингу устроить телеграфное сообщение между Петергофом и Кронштадтом и для этого составить проект и смету. Выполнить поручение ученый не успел: летом 1837 года Павел Львович Шиллинг скончался.



Барон Павел Львович Шиллинг фон Канштадт



Телеграфный аппарат Шиллинга “образца 1828 года”

27 апреля 1791 года в Чарльзтауне (штат Массачусетс) родился Сэмюэл Морзе (1791–1872) — американец, внесший существенный вклад в усовершенствование электромагнитного телеграфа, создатель знаменитой “азбуки Морзе”.

Иногда пишут, что электромагнитный телеграф изобрел Морзе. Однако хорошо известно, что телеграф, основанный на принципе электромагнитного воздействия, был изобретен Шиллингом. Причина путаницы проста [2]: в одном случае термин *электромагнитный* употребляется для указания принципа действия, а в другом — для указания факта применения электромагнита: в конструкции приемника Морзе использовал электромагнит.

30 апреля 1916 года в небольшом городе Петоски (штат Мичиган) родился Клод Элвуд Шеннон (1916–2001) — “человек, который придумал бит”.

Самой известной работой Клода Элвуда Шеннона является опубликованная в 1948 году “Математическая теория связи”, где представлены соображения, касающиеся созданной им новой науки —



Сэмюэл Морзе

Продолжение на с. 2

СОДЕРЖАНИЕ НОМЕРА

МОДЕЛИРОВАНИЕ

Р.В. Бирих, Е.А. Еремин, В.И. Чернатынский. Компьютерные модели школьных физических задач 3–10

ПРЕДЛАГАЮ КОЛЛЕГАМ

И.Н. Фалина. Тема “Машина Тьюринга” в школьном курсе информатики ... 11–16

КОНКУРС: “КАБИНЕТ — МОЙ ВТОРОЙ ДОМ”

Итоги V тура 17–18
М.М. Барсукова. Кабинет — второй дом 18–19

В.В. Меньшиков. Функциональность и уют кабинета информатики 19–20

Н.И. Ермакова. Новый стенд? Легко! 20–22

К.В. Чернобабова. Творческая рабочая мастерская 22–23

Е.Н. Смирнова. Мои представления об идеальном кабинете 23–24

В.В. Пичугин. Секреты комфортного кабинета 24–25

И.С. Исакова, Л.Н. Крылова. Наш кабинет — наша гордость! 25–27

Ж.В. Можарова. Стараюсь, чтобы кабинет не был скучным и “холодным” 27–29

ИНФОРМАЦИЯ

Набор слушателей на курсы повышения квалификации 10

Подписка-2006. Второе полугодие 30

Приглашаем на курсы Ротландского университета! 36

“НАЧАЛКА” № 8

Газета-клуб для всех, кто учит информатике маленьких детей

В.А. Козлова. Потрогать информатику руками. Часть 2 31–35

“В МИР ИНФОРМАТИКИ” № 73

Газета для пытливых учеников и их талантливых учителей

Семинар
Признаки делимости 37

Школа программирования
Н.М. Тимофеева. Основы программирования на Visual Basic 38–40

Задачник
Ответы, решения, разъяснения 40–42

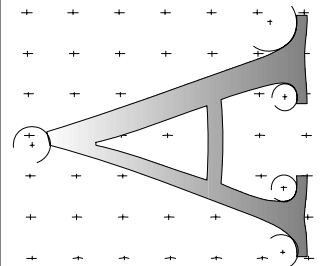
“Ломаем” голову
Как получается молоко .. 43

Внимание! Конкурс
Итоги пятого тура конкурса № 41 43–46

Конкурс № 48 для учащихся 46

№ 8 (513)

16–30 апреля 2006



Методическая газета для учителей информатики

ИНФОРМАТИК

Учителю информатики: памятные даты и события апреля

Продолжение. См. с. 1

теории информации [3, 4]. Шеннон определил основную единицу количества информации (названную потом битом) как сообщение, представляющее один из двух вариантов: *герб* — *число*, *да* — *нет* и т.п. Бит можно представить как 1 или 0, или как присутствие или отсутствие тока в цепи.

19 апреля 1931 года в Москве родился Андрей Петрович Ершов (1931–1988) — выдающийся программист и математик, академик, автор первой в мировой практике монографии по автоматизации программирования.

Для научной деятельности Андрея Петровича Ершова характерно сочетание теоретических исследований и практических работ [5]. Творческое наследие Ершова обширно, оно включает около 200 работ (в том числе ряд монографий), не считая большого числа предисловий, редакционных статей, отзывов и т.п.

Как и большинство программистов 1950-х годов, он начинал с создания алгоритмов численных методов и стандартных программ. Первая работа Ершова была посвящена предложенному им методу обращения матриц. В середине 1950-х годов происходило становление системного и так называемого “теоретического” программирования. Развитие системного программирования началось с работ в области, названной тогда автоматизацией программирования и связанной с созданием языков программирования и методов их трансляции. Первые отечественные языки программирования были основаны на операторных схемах, предложенных А.А. Ляпуновым (1911–1973). Под руководством Ершова разрабатывались одни из первых отечественных программирующих программ — для машин БЭСМ и “Стрела” (программирующими программами назывались в то время “интегральные разработки” языка и системы программирования).

Идейно к работам по автоматизации программирования прилегают и труды Ершова по теоретическому программированию, “чьим объектом изучения являются математические абстракции программ” [6].

Обобщая свой опыт руководства большими программными проектами, Ершов выдвинул несколько тезисов, касающихся организации работы программистов, сформулировал ряд общих принципов программирования как нового и своеобразного вида научной деятельности, затронул тот аспект, который впоследствии будет назван *дружественностью к пользователю*, одним из первых в стране поставил задачу создания технологии программирования. [Сам русский термин “технология программирования” (*software engineering*) был предложен Ершовым.]

Благодаря своему умению видеть новые перспективы в программировании Ершов часто выступал как инициатор новых направлений исследований. Сюда относятся, например, методы использования естественных языков для взаимодействия с ЭВМ, автоматизация редакционно-издательской деятельности [7].

Большое внимание Ершов уделял подготовке кадров, причем в последний период жизни Андрея Петровича Ершова привлекло обучение программированию и, более широко, информатике в школе. Он стал одним из создателей так называемой “школьной информатики” и признанным лидером отечественной школьной информатики, вошел в число ведущих мировых специалистов в этой области. В подготовленной им вместе с Г.А. Звенигородским и Ю.А. Первиным работе “Школьная информатика. Концеп-



Андрей Петрович Ершов

ции, состояние, перспективы” определены перспективы развития данного направления на годы вперед. Андрей Петрович был одним из авторов и редакторов первого школьного курса информатики, а также методического пособия к этому курсу. Незадолго до его кончины появился новый учебник информатики [8], одним из авторов и редактором которого он являлся. Ершов готовил телевизионный курс по этой дисциплине, руководил созданием школьных систем программирования и школьного программного обеспечения вообще, вел большую организационную работу [5]. Главным делом последних лет его жизни, которому он отдал много сил, было создание школьной энциклопедии по информатике.

25 апреля 1903 года в Тамбове родился Андрей Николаевич Колмогоров (1903–1987) — один из крупнейших математиков XX века, построивший (в 1933 году) широко теперь известную систему аксиоматического обоснования теории вероятностей.

Андрей Николаевич создал большую школу в области теории вероятностей и теории функций. Академик Колмогоров являлся иностранным членом Парижской академии наук, Лондонского королевского общества и ряда других зарубежных академий (в Голландии, Польше, Румынии, США), научных учреждений и обществ [9].

30 апреля 1777 года в городе Брауншвейге родился Карл Фридрих Гаусс (1777–1855) — крупнейший немецкий математик, астроном и физик, член Лондонского королевского общества, Парижской и Петербургской академий наук.

Его именем названа единица измерения магнитной индукции (*гаусс*) в системе СГС.

В апреле 1976 года было опубликовано второе “Открытое письмо любителям” Билла Гейтса, в котором он порицает незаконное копирование программ.

В конце 1975 года разработчики программного обеспечения — интерпретатора языка Бейсик — для компьютера “Альтаир” Билл Гейтс и Пол Аллен обнаружили, что при росте сбыта “Альтаиров” объем продаж интерпретатора резко уменьшился. Сначала они не понимали, в чем дело, но потом пришли к выводу, что их разработку просто бесплатно заимствуют. Возмущенный Гейтс опубликовал в одном из компьютерных изданий “Открытое письмо любителям”, которое быстро растиражировали другие издания. Начиналось оно так [1]: “Как, должно быть, известно многим любителям, большая часть из вас пользуется похищенным вами же программным обеспечением. За аппаратуру приходится платить, а программы — это что-то такое, чем можно поделиться. Кого интересует, платили ли за него люди, которые с ним работают?”, “Кто станет работать даром? Какой любитель готов потратить три года на разработку, отладку, документирование программы, чтобы затем убедиться, что она расходуется бесплатно?”

Скорее всего автор “Открытого письма” не ожидал реакции, последовавшей за публикацией. Любители, к которым он обращался, привыкли в своих компьютерных клубах к тому, что любящая информация, независимо от того, каких трудов и средств стоило ее получить, не скрывалась и распространялась безвозмездно. Они сочли эти обвинения неуместными и оскорбительными (а один клуб вообще собирался подать на Гейтса в суд за то, что он назвал всех любителей ворами).

В апреле 1976 года Гейтс опубликовал второе “Открытое письмо любителям”, где уже несколько смягчил свою позицию, но по-прежнему порицал незаконное копирование программ: “...Возможно, настоящая дилемма возникла из-за того, что многие просто не понимают, что ни Microsoft, ни кто угодно не может разрабатывать большие программы, не получая разумной прибыли...”. Кстати, в этом письме Гейтс предложил “зашивать” интерпретатор в постоянное запоминающее устройство (ПЗУ), и эта идея стала впоследствии общепризнанной.

Окончание на с. 48

Компьютерные модели школьных физических задач

Р.В. БИРИХ, Е.А. ЕРЕМИН, В.И. ЧЕРНАТЫНСКИЙ,
г. Пермь

В этом году в серии “Жаркое лето” будет опубликовано наше пособие по курсу компьютерного моделирования. В качестве объектной области выбрана физика, предоставляющая множество проблем, которые могут быть эффективно решены методами математического моделирования. Предполагается не просто представить читателю набор отдельных задач, а изложить научный подход к моделированию физических процессов и продемонстрировать его приложение к школьной физике.

Публикация будет состоять из нескольких связанных частей: общая стратегия физического моделирования, особенности технологии математического моделирования на компьютере и, наконец, реализация этих принципов в школьных физических задачах. Будут рассмотрены вопросы постановки математической задачи на базе фундаментальных физических законов, выбора масштабов физических величин, анализа влияния погрешностей вычислений на достоверность физических результатов, особенности представления результатов на экране дисплея в графической и табличной формах и другие. Сформулированные общие положения будут проиллюстрированы на конкретных физических примерах. Далее описан один из них.

Движение в центральном поле тяготения

Постановка задачи. Задача о движении тела в центральном поле тяготения является хорошим примером, демонстрирующим возможности использования ПК для изучения поведения объекта, подчиняющегося некоторым общим физическим законам.

Процесс выведения спутника на орбиту обычно разбивается на два этапа. На первом этапе спутник поднимается над атмосферой практически вертикально на некоторую высоту. Затем обычно последняя ступень ракетносителя придает спутнику необходимую горизонтальную скорость, и далее он движется по инерции.

Рассмотрим инерционный полет небольшого тела (спутника) около притягивающего центра с большой массой (Земли). Будем интересоваться тем, какие траектории спутника возможны, какой должна быть его минимальная скорость вблизи поверхности Земли, чтобы он, двигаясь по круговой траектории, не упал на Землю (*первая космическая скорость*),

какой должна быть минимальная начальная скорость спутника, чтобы получилась незамкнутая траектория и спутник ушел от Земли (*вторая космическая скорость*). В численном эксперименте можно также проверить законы Кеплера: 1) притягивающий центр расположен в одном из фокусов орбиты; 2) радиус-вектор от Солнца до планеты “заметает” равные площади за равные промежутки времени; 3) квадраты периодов обращения спутников вокруг притягивающего центра относятся как кубы больших полуосей эллипсов их орбит.

В основу модели мы положим закон всемирного тяготения. Будем считать, что на тело, движение которого рассматривается, действует только сила тяготения и уравнение Ньютона имеет вид:

$$m\mathbf{a} = -\frac{GMm}{r^3}\mathbf{r}. \quad (1)$$

Здесь m и M — масса спутника и масса притягивающего центра, G — гравитационная постоянная, \mathbf{r} — радиус-вектор, задающий положение спутника относительно притягивающего центра, \mathbf{a} — ускорение спутника. Заметим, что хотя закон всемирного тяготения в форме (1) записан для материальных точек, он имеет такую же форму для сферически-симметричных тел.

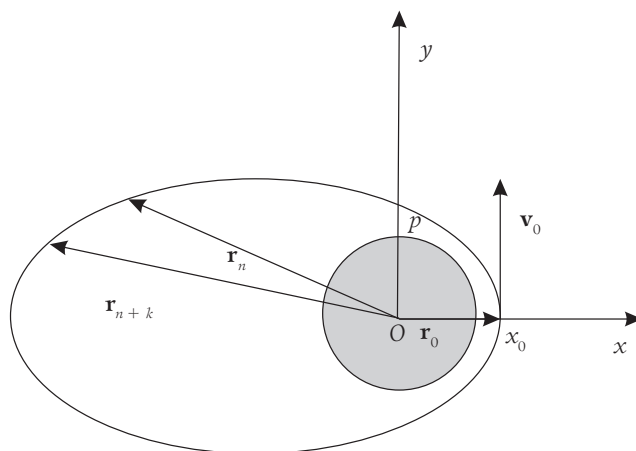


Рис. 1. Система координат и возможная траектория спутника

Движение тела под влиянием центральной силы происходит в одной плоскости, положение которой определяется векторами \mathbf{r}_0 и \mathbf{v}_0 , задающими началь-

ное положение тела и его начальную скорость. Декартову систему координат с началом в центре тяготения и начало отсчета времени выберем так, чтобы движение происходило в плоскости Oxy и в начальный момент скорость тела была перпендикулярна оси x (рис. 1).

Тогда начальные условия можно записать в виде:

$$t = 0: x = x_0, y = 0, v_x = 0, v_y = v_0 \quad (2)$$

Уравнение (1) вместе с условиями (2) полностью определяют траекторию спутника и все ее свойства.

Численная модель. Численный анализ задачи удобно проводить, используя в качестве единиц измерения характерные масштабы задачи. В качестве единицы длины удобно взять x_0 . Если разговор идет о спутнике Земли, то эта величина имеет порядок радиуса Земли R и равняется $R + h$, где h — высота спутника над поверхностью Земли. Всякое расстояние теперь будет задаваться числом, которое показывает, сколько раз в нем укладывается x_0 . Безразмерное x будет равняться x , измеренному в метрах, деленному на x_0 , также измеренному в метрах. Единицу времени удобно построить, используя гравитационную постоянную и характеристики притягивающего центра. Из уравнения (1) легко видеть, что множитель GM/r^2 имеет размерность ускорения (m/c^2). Вместо расстояния r возьмем x_0 и сформируем выражение с размерностью времени (c): $(GM/x_0^3)^{-1/2}$. Его и выберем в качестве единицы времени. В качестве единицы скорости тогда естественно взять $x_0 / (GM/x_0^3)^{-1/2}$, т.е. $(GM/x_0)^{1/2}$.

Измеренные в этих единицах проекции ускорения определяются следующими уравнениями (здесь и далее для безразмерных физических величин использованы те же обозначения, какие использовались для соответствующих размерных величин):

$$a_x = -\frac{x}{(x^2 + y^2)^{3/2}}, a_y = -\frac{y}{(x^2 + y^2)^{3/2}}, \quad (3)$$

а начальные условия принимают вид:

$$t = 0: x = 1, y = 0, v_x = 0, v_y = V_0, \quad (4)$$

где $V_0 = v_0 \cdot (x_0/GM)^{1/2}$.

Все физические величины измеряются теперь в относительных единицах и будут одинаковыми для всех систем “спутник — притягивающий центр”. Уменьшилось также число параметров задачи. Единственный безразмерный параметр V_0 , который остался в задаче, показывает, как соотносятся между собой кинетическая и потенциальная энергии спутника в начальный момент. Действительно, кинетическая энергия спутника в начальный момент равна $K = mv_0^2/2$, потенциальная энергия для ньютонов-

ского закона тяготения равна $\Pi = GMm/x_0$, и $V_0^2 = 2K/\Pi$.

Для нахождения в различные моменты времени проекций скорости спутника и его координат на временной оси выберем дискретные точки t_n , отстоящие друг от друга на малые интервалы Δt . Тогда проекции скорости $v_x^{(n+1)}$ и $v_y^{(n+1)}$ в момент времени t_{n+1} будут приближенно (считаем, что ускорение на этом интервале времени не изменилось) представляться выражениями

$$v_x^{(n+1)} = v_x^{(n)} + \Delta t \cdot a_x^{(n)}, \quad (5)$$

$$v_y^{(n+1)} = v_y^{(n)} + \Delta t \cdot a_y^{(n)}, \quad (6)$$

а координаты в этот момент будем вычислять, как при равномерном движении (опять считая, что интервал времени Δt мал, и скорость в течение него такая, как в конце интервала):

$$x^{(n+1)} = x^{(n)} + \Delta t \cdot v_x^{(n)}, \quad (7)$$

$$y^{(n+1)} = y^{(n)} + \Delta t \cdot v_y^{(n)}. \quad (8)$$

В начальный момент времени проекции скорости и координаты спутника известны:

$$t_0 = 0: x^{(0)} = 1, y^{(0)} = 0,$$

$$v_x^{(0)} = 0, v_y^{(0)} = V_0.$$

Система (5)–(8) позволяет шаг за шагом, при малом Δt , достаточно точно вычислить траекторию спутника и все ее характеристики.

Для классификации траекторий удобно вычислять их эксцентриситет e . В выбранной системе координат для вычисления эксцентриситета необходимо определить координату $y = p$ точки на траектории, для которой $x = 0$. Тогда $e = |p - 1|$. Если $e < 1$, то траекторией является замкнутая кривая (окружность при $e = 0$ и эллипс при $e \neq 0$), при $e = 1$ (в численном эксперименте это значение точно получено быть не может) траекторией является парабола, а при $e > 1$ траектория есть гипербола.

Площадь, “заметаемую” радиус-вектором за некоторое время $T_k = t^{(n+k)} - t^{(n)} = k \cdot \Delta t$, можно вычислить следующим образом:

$$S_k = \frac{1}{2} \sum_{i=n}^{n+k} (x^{(i)} \cdot v_y^{(i)} - y^{(i)} \cdot v_x^{(i)}) \Delta t. \quad (9)$$

Интересно сравнить эти площади, вычисленные на разных участках траектории. По второму закону Кеплера они должны быть равны.

По третьему закону Кеплера квадраты периодов обращения относятся как кубы больших полуосей орбит, т.е. для каждой траектории величина $K3 = T^2/a^3$ должна оставаться постоянной. Для нахождения длины большой полуоси эллипса необхо-

дим найти отрицательную координату x_{\min} при $y = 0$. Момент времени t_m , когда это произойдет первый раз, соответствует половине периода обращения спутника. Таким образом, величина большой полуоси равна: $a = (1 + |x_{\min}|)/2$, а период $T = 2t_m$.

Моделирование задачи в среде Delphi. Современная визуальная система программирования Delphi предоставляет удобное средство для создания среды моделирования любой физической задачи. Ее визуальная часть, будучи необычайно простой и естественной, позволяет быстро и без особых усилий создать на экране компьютера современный диалоговый интерфейс, который не уступает применяемому в профессиональном программном обеспечении. Лежащий в основе системы алгоритмический язык Паскаль является классическим учебным языком, так что запрограммировать на нем небольшую вычислительную задачу не представляет особой трудности. Наконец, встроенные в систему стандартные компоненты для поддержки таблиц и графики служат приятным дополнением к перечисленным выше достоинствам.

По Delphi выпущено огромное число весьма объемных книг, поэтому среди целей нашей публикации не будет подробного описания системы и простейших приемов взаимодействия с ней с помощью мыши. В то же время отдельные важные вопросы программирования в системе (например, связанные с оптимизацией рисования графиков, формированием таблиц и организацией взаимодействия модулей), напротив, будут опи-

саны максимально подробно. Иными словами, при изложении материала предполагается достаточным знание читателями основных правил записи программы на языке Паскаль и простейшие навыки работы с Delphi, но ничего сверх этого не потребуется.

Перейдем к описанию среды моделирования нашей задачи. Как и любое Windows-приложение, она будет состоять из окон. Для наших целей естественным образом напрашивается создание четырех окон:

- главного с основными органами управления процессом численного моделирования,
- окна построения графиков
- и двух окон с таблицами наиболее важных физических характеристик: в первое помещаются интегральные характеристики процесса в целом, а во второе — промежуточные числовые результаты расчетов по каждому из графиков.

Возможный вид реализации виртуальной лабораторной среды приведен на рис. 2.

Поскольку подавляющее большинство органов управления сосредоточено в главном окне, имеет смысл обратить основное внимание на его рассмотрение. Из приведенного рисунка устройство главного окна видно вполне отчетливо.

В левой части рассматриваемого окна располагается панель с 4 полями ввода, используя которые можно регулировать отображаемую в графическом окне область. Рядом со схематическим прямоугольником графического экрана предусмотрены поля для ввода минимального и максимального значения по

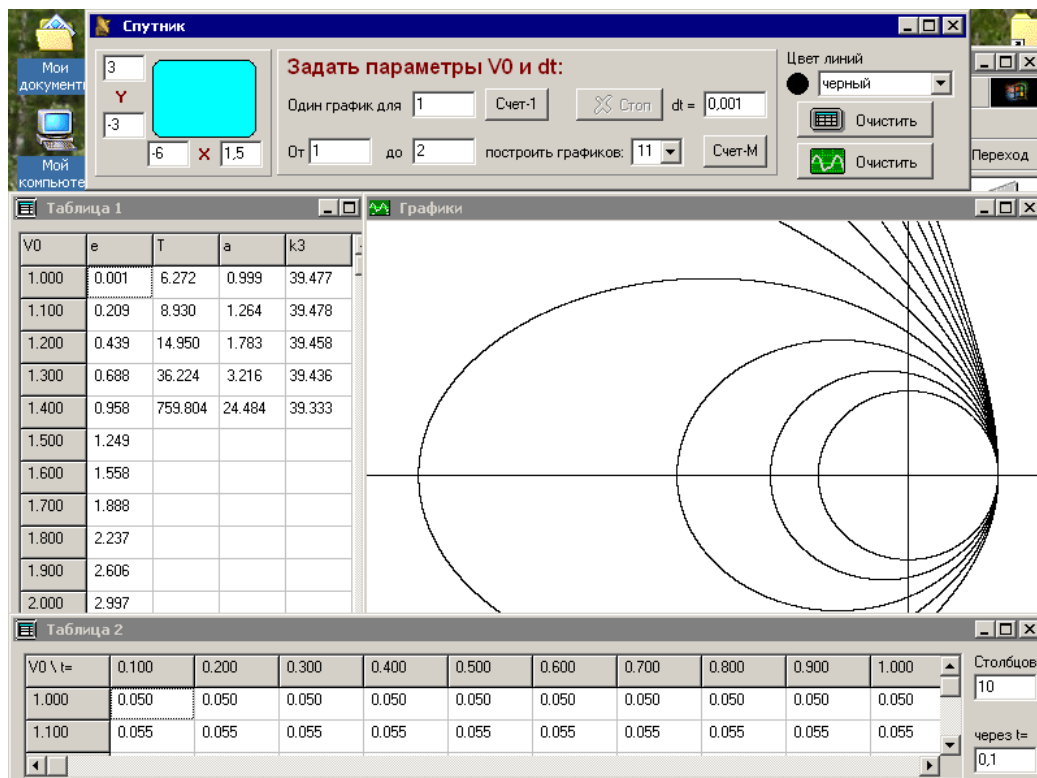


Рис. 2. Возможный вид виртуальной лабораторной среды

каждой из осей X и Y . Отметим, что координаты, как и все остальные вводимые в главном окне параметры, указываются непосредственно в “физических” единицах; необходимый для рисования перевод в пиксели экрана берет на себя программа.

Следующая панель, занимающая основную часть главного окна в его центре, служит для задания параметров задачи — скорости запуска спутника V_0 и шага по времени dt (Δt — в постановке задачи) — а также запуска и прерывания расчетов. Предусмотрено задание значения V_0 двумя способами. В первом случае строится график для одного значения параметра: оно задается в поле правее слов “Один график для” и программа запускается на счет кнопкой “Счет-1”. Во втором производится автоматическое вычисление сразу нескольких траекторий: вводятся начальное и конечное значения V_0 , для которых необходимо произвести расчет, из списка выбирается количество требуемых траекторий (их может быть от 2 до 11) и нажимается кнопка “Счет-М”. Сочетание двух указанных режимов проведения расчетов позволяет производить численные эксперименты значительно удобнее, получая любые требуемые траектории. В обоих случаях можно в любой момент прервать расчет нажатием кнопки “Стоп”.

Примечание. В Delphi при вводе дробных чисел в компоненты полей ввода целую и дробную части следует разделять символом, который определяется национальными настройками ОС Windows, а именно: в русской версии необходимо использовать запятую, а в английской — точку. Для языка программирования подобное решение проблемы кажется несколько непоследовательным (согласно их синтаксису в вещественных числах всегда однозначно используется точка); тем не менее, если вспомнить электронные таблицы Excel, то там тоже имеет место аналогичный “двойственный” подход.

В правой части главного окна есть еще три полезных органа управления: выпадающий список для выбора цвета рисования последующих траекторий и две кнопки очистки графического окна и значений в таблицах (они различаются по нанесенным на них пиктограммам).

Таким образом, описанные выше органы управления позволяют провести полное исследование физической модели спутника. Можно пробовать рисовать различные траектории и стирать их, подбирая именно те значения параметра, которые требуется найти в соответствии с целью исследования, в частности, в соответствии с поставленным учителем заданием.

Окна с графиками функций и табл. 1 интегральных результатов для каждой траектории никаких органов управления не содержат. Зато для построения табл. 2 в правой части окна помещены два поля для ввода параметров построения данной таблицы. В первое вводится количество столбцов табл. 2, которое с точки зрения физического содержания задачи равняется числу точек траектории, для которых будет зафиксировано значение некоторой интересующей нас характеристики (в данной задаче это площадь сектора). Что касается второго поля, то в него задается значение вре-

мени, через которое эти значения будут заноситься в таблицу. Например, при указанных на рисунке значениях в таблицу для каждой траектории попадет по 10 точек для значений времени 0,1, 0,2, ..., 1.

Конкретные детали реализации органов управления, которые определяются программами, обрабатывающими их нажатия, будут приведены в летнем выпуске.

Эксперимент на модели в среде Delphi. Проведем экспериментальное исследование движения спутника с помощью описанной выше модели. Для проведения численного эксперимента нужно прежде всего выбрать шаг по времени dt , с которым будем строить траекторию. По смыслу метода он должен быть достаточно малым, чтобы на этом интервале можно было пренебречь изменением ускорения при вычислении скорости и изменением скорости при вычислении новых координат. Шаг не должен быть и слишком маленьким, так как в этом случае для построения траектории потребуется большое время. Поскольку в качестве масштабов мы выбрали характерные параметры системы, то типичный временной интервал жизни системы равен 1. Малый интервал dt поэтому должен быть много меньше единицы; выберем его, например, равным 0,01. Второй параметр, который нужно задать, это начальная скорость. Опять, пользуясь тем, что расчет ведется в собственных масштабах задачи, для первого эксперимента естественно положить $V_0 = 1$. Теперь можно построить одну траекторию, нажав кнопку “Счет-1”. Посмотрим на полученную траекторию. Она похожа на окружность, но не замкнулась при обороте спутника вокруг притягивающего центра (внутренняя кривая на графиках рис. 3). Это, по-видимому, связано с недостаточной точностью счета. Давайте уменьшим величину dt в два раза: $dt = 0,005$ и изменим цвет кривой. Экран и таблицы чистить не будем, чтобы сравнить результаты, полученные для разных dt . Нажимаем кнопку “Счет-1”. Результат лучше, но требуется еще более мелкий шаг. Сделаем $dt = 0,001$.

Траектория получилась замкнутая. Если мы теперь посмотрим на интегральные характеристики построенных кривых (табл. 1 на рис. 3), то заметим, что эксцентриситет e с увеличением точности счета уменьшался и приблизился к нулю. Может быть не ноль, потому что еще грубо считаем? Уменьшим dt до 0,0001. Новый цвет кривой и кнопка “Счет-1”. (На рис. 3 приведено состояние окон после проведенных расчетов.) Вот теперь хорошо! Эксцентриситет равен нулю, большая полуось равна единице — значит, траектория окружность. Первая космическая скорость в принятых единицах равна 1, а период обращения — 2π . В нижней таблице видно, как хорошо соблюдается для этой траектории второй закон Кеплера: площадь, “заметаемая” радиус-вектором за одно и то же время на разных участках траектории, одинакова.

Теперь можно посмотреть, какие еще бывают траектории спутника. Построим серию траекторий из 11 линий с диапазоном начальных скоростей от 1 до 2. Очистим графическое поле и таблицу, наберем необходимые параметры и кнопкой “Счет-М” запускаем программу. На *рис. 4* приведены результаты этого эксперимента. Видно, что с увеличением начальной скорости эллипсы вытягиваются вдоль оси x . Притягивающий центр (начало координат) находится в правом фокусе эллипсов. Последняя замкнутая траектория получилась для $V_0 = 1,4$ (она так сильно вытянута, что не поместилась на экране). Для этой траектории эксцентриситет близок 1, а значит, соответствующая ей начальная скорость близка ко второй космической скорости. Анализ данных таблиц показывает, что второй и третий законы Кеплера хорошо выполняются.

Моделирование задачи в среде MS Excel. Рассмотрим теперь реализацию модели с помощью MS Excel. Общий алгоритм состоит из трех блоков. Первый — формирование таблицы значений координат спутника в последовательные моменты времени; второй — построение графика с помощью встроенного мастера диаграмм; третий — оформление интерфейса.

Первый блок содержит следующие шаги:

- определение параметров таблицы,
- определение структуры записи, т.е. данных, которые должны быть расположены в строке таблицы,
- определение количества записей,
- заполнение таблицы расчетными формулами и необходимыми значениями.

Шаги, образующие второй блок, однозначно определяются стандартным мастером диаграмм Excel.

Третий блок определяет удобное для проведения исследований расположение параметров, таблиц значений и графиков зависимостей.

Рассмотрим перечисленные шаги более подробно.

Параметры таблицы. В качестве параметров следует выбирать величины, задание которых определяет размеры таблицы и числовые значения данных, получаемых в ее ячейках. Предполагается, что значения параметров не изменяются в процессе вычислений; они размещаются в фиксированных ячейках, при обращении к которым используется абсолютная адресация. В сформулированной задаче значения координат спутника в любой момент времени однозначно определяются начальными условиями, поэтому их естественно опре-

делить в качестве параметров. Выбранный метод пошагового решения задачи содержит в качестве основной характеристики шаг по времени dt , который также удобно включить в список параметров. Таким образом, таблицу определяют четыре физических параметра — x_0, y_0, vx_0, vy_0 и один вычислительный — dt .

Структура записи. В ячейках строки таблицы (слева направо) следует расположить такие данные: время t , проекция скорости на ось x , проекция скорости на ось y , координата x , координата y , мгновенная секториальная скорость и две вспомогательные ячейки. Все данные (кроме секториальной скорости) вычисляются по формулам численного метода, которые представляют собой рекуррентные соотношения. С их помощью значение в данной ячейке определяется значениями величин в предыдущий момент времени, т.е. в предыдущей строке. Вспомогательные ячейки используются для выделения значений координат x и y , которые определяют такие параметры траектории, как эксцентриситет e и большая полуось a .

Количество записей. Размер таблицы, очевидно, определяется количеством шагов по времени, которое, в свою очередь, зависит от величины шага по времени dt . Поэтому мы должны сделать выбор до

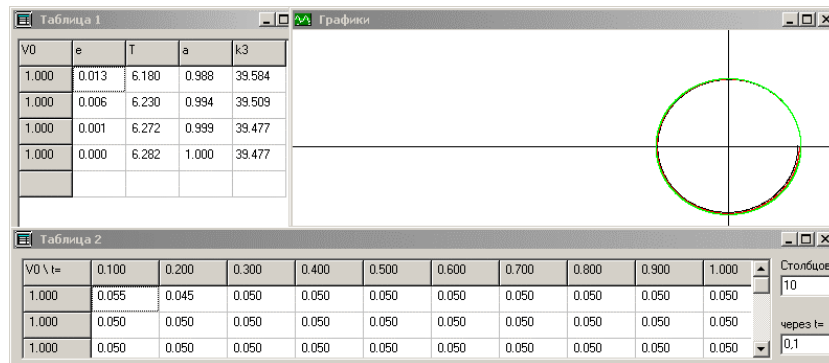


Рис. 3. Состояние окон с результатами после эксперимента по выбору dt

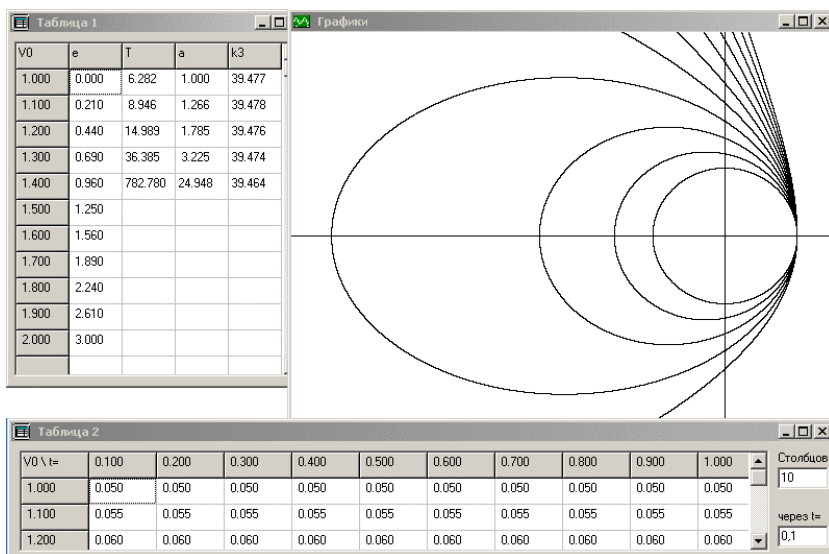


Рис. 4. Состояние окон с результатами после расчета серии траекторий

решения задачи. Практически шаг по времени определяется устойчивостью вычислительной процедуры и точностью решения, а максимальное число шагов задается в процессе решения каким-либо дополнительным условием (условием замыкания траектории, ограничением значений координат для незамкнутых траекторий и т.д.). Обойти эти трудности можно двумя путями. Первый состоит в изменении количества записей в зависимости от траектории, сохраняя постоянный шаг по времени. Другой способ заключается в создании большой таблицы для маленького шага по времени, используя для определения количества записей круговую траекторию в качестве эталона; обследовать траектории, соответствующие разным периодам обращения спутника, можно, изменяя шаг по времени.

Заполнение таблицы. Для заполнения таблицы нужно первоначально сформировать две строки. В первой строке соответствующим величинам присваиваются начальные значения. Вторая строка содержит вычислительные формулы, в которых значения величин берутся из ячеек предыдущей строки. Для ссылок на эти ячейки используется относительная адресация. Заполнение оставшихся строк таблицы производится простым копированием второй строки.

Построение траекторий. Для построения траектории рекомендуется использовать диаграмму точечного типа, сглаженную, без координатных точек. Чтобы получить правильные пропорции кривой, следует определить оси системы координат одинакового размера, выбирая начало координат по горизонтали с учетом того, что максимальное значение этой координаты равно 1. Для увеличения полезной площади диаграммы легенду рекомендуется удалить.

Организация интерфейса. Для более удобной работы предлагается начальные данные с графиком траектории и таблицу результатов разместить на отдельном листе Книги MS Excel, а громоздкую таблицу с расчетами значений координат точек траектории — на следующем.

Следует обратить внимание на то, что математическая модель, создаваемая в MS Excel, оказывается менее гибкой и наглядной, чем модель, полученная с помощью среды программирования Delphi. В частности, в последней можно создать иллюзию движения спутника по траектории, представить для сравнения множество траекторий в одних осях, удобно

провести вычисления характеристик траекторий для проверки законов Кеплера.

Эксперимент на модели в среде MS Excel. Расположим интерфейсную часть модели, т.е. параметры, диаграмму и характеристики траекторий, на первом листе книги, а таблицу вычисленных значений переменных на втором листе. Пусть параметры x_0 , y_0 , vx_0 , vy_0 и dt в перечисленном порядке располагаются в ячейках B2, C2, D2, E2 и F2 Листа 1, соответственно. Тогда первые две строки таблицы значений на Листе 2, начиная со строки 4, заполняются формулами, приведенными в табл. 1.

Последующие записи получаются копированием строки 5, предварительно вычислив требуемое количество строк.

Размер таблицы определим из следующих соображений. При движении по окружности радиуса, равного 1, с характерной скоростью, равной 1, полный оборот совершается за промежуток времени, равный $2\pi \approx 6,28$. Шаг по времени выберем равным 0,001, так что таблица будет содержать 6280 строк. Для заполнения такой большой таблицы рекомендуется использовать при копировании строк поле адреса строки ввода.

Для вычисления характеристик траектории и проверки законов Кеплера таблица значений дополнена тремя вспомогательными столбцами — F, G, H. В столбце F, начиная с первой строки таблицы, вычисляется комбинация проекций скорости и координат спутника для нахождения средней секториальной скорости на заданном промежутке времени. В столбце G с помощью функции ветвления ЕСЛИ и логического произведения И выделяются значения координаты $y > 0$ (столбец E4) в малой окрестности точки $x = 0$. Остальные ячейки столбца обнуляются. Таким образом, создается структура значений, позволяющая с помощью интервальной функции МАХ найти приближенно координату точки, в которой траектория пересекает ось y , и вычислить эксцентриситет e . Аналогичная структура данных создается во вспомогательном столбце H для вычисления большой полуоси. При этом, поскольку выделяется малый диапазон отрицательных значений x , необходимо использовать функцию MIN. Соответствующие формулы приведены в табл. 2, являющейся фрагментом строки 4.

Таблица 1

| | A | B | C | D | E |
|-----|------------------|--|--|----------------------|----------------------|
| ... | t | vx | vy | x | y |
| 4 | 0 | = Лист1!\$B\$2 | = Лист1!\$C\$2 | = Лист1!\$D\$2 | = Лист1!\$E\$2 |
| 5 | =A4+Лист1!\$F\$2 | =B4+ Лист1!\$F\$2* (-D4/КОРЕНЬ(СТЕПЕНЬ ((D4*D4+E4*E4);3))) | =C4+ Лист1!\$F\$2* (-E4/КОРЕНЬ (СТЕПЕНЬ ((E4*E4+D4*D4);3))) | =D4+ Лист1!\$F\$2*B4 | =E4+ Лист1!\$F\$2*C4 |

Функции MIN и MAX используются в столбцах *a* и *e* таблицы результатов.

| F | G | H |
|--------------|-----------------------------------|-----------------------------------|
| =D4*C4-E4*B4 | =ЕСЛИ(И(ABS(D4)<0.001;E4>0);E4;0) | =ЕСЛИ(И(ABS(E4)<0.001;D4<0);D4;0) |

Таблица 2

Вычисления указанных характеристик производятся в таблице результатов, которая также включается в интерфейс. Период обращения определяется как значение ячейки A6280, когда получена замкнутая траектория. Среднее значение секториальной скорости, вычисленное путем усреднения формулы (9) для 20 точек в окрестности значений горизонтальной координаты, соответствующих максимальному и минимальному удалению от силового центра, помещается в раздел K2 таблицы результатов.

Для реализации возможности сопоставления траекторий, соответствующих разным начальным состояниям, и проверки третьего закона Кеплера были созданы таблицы для двух траекторий, данные из которых представлялись на одной диаграмме (см. рис. 5). Таким образом, интерфейс представлял собой две строки параметров, две строки результатов обработки числовых данных и диаграмму с двумя графиками.

Таблица начальных значений

| v0 | u0 | x0 | y0 | Dt |
|----|-----|----|----|---------|
| 0 | 1.3 | 1 | 0 | 0.0062 |
| 0 | 1.1 | 1 | 0 | 0.00145 |

Таблица результатов

| e | T | a | K2 | K3 |
|----------|---------|----------|----------|----------|
| 0.698758 | 38.9112 | 3.359113 | 0.650249 | 0.655714 |
| 0.212295 | 9.1002 | 1.272125 | 0.550012 | 0.551804 |

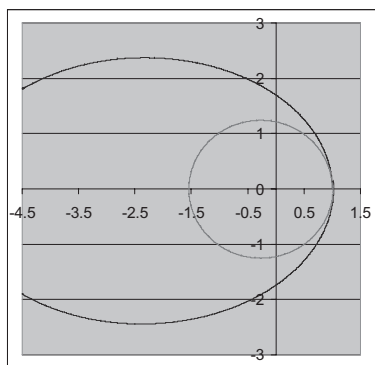


Рис. 5. Результаты эксперимента на модели MS Excel

Примеры вопросов для самостоятельного решения

1. Какие в проведенных исследованиях получают размерные первая и вторая космические скорости для спутника Земли?

Ожидаемый ответ. Нужно вычислить единицу

скорости для спутника у поверхности Земли: $(GM/x_0)^{1/2} = (6,7 \cdot 10^{-11} \text{ м}^3 / (\text{кг} \cdot \text{с}^2)) \cdot 6,0 \cdot 10^{24} \text{ кг} / (6,4 \cdot 10^6 \text{ м})^{1/2} = 7,9 \text{ км/с}$. Для первой космической скорости это число нужно умножить на 1, а для второй — на 1,4.

2. Какие траектории спутника получатся, если начальная безразмерная скорость его меньше единицы?

Ожидаемый ответ. Траекториями будут эллипсы с Землей в левом фокусе эллипса. Если начальная точка была недостаточно высоко над Землей, то траектория пересечет поверхность Земли, т.е. произойдет падение спутника на Землю.

3. Как изменятся значение площади, заметаемой радиус-вектором, и средняя секториальная скорость, если интервал времени, за который эти величины считаются, увеличить вдвое? В 5 раз?

Ожидаемый ответ. Площадь увеличится в соответствующее число раз. Убедиться в этом можно, анализируя содержимое табл. 2 в разных экспериментах. Средняя секториальная скорость не изменится.

Примеры заданий для самостоятельного исследования

1. Построить 11 траекторий для значений начальной скорости в диапазоне от 0,8 до 1,8. Используя режим расчета одиночной траектории, нанести на рисунок красным цветом круговую траекторию, а розовым — самую вытянутую траекторию, целиком уместившуюся на экране.

2. Произведите расчет для значений начальной скорости 0,4. Получилась ли траектория замкнутой? Как добиться, чтобы она стала замкнутой (уменьшить шаг по времени)?

3. Постройте 6 траекторий для значений начальной скорости в диапазоне от 1,4 до 1,42. Определите, сколько из них будут замкнутыми. Найдите в таблице значение большой полуоси для самой вытянутой замкнутой траектории. Попробуйте подобрать масштабы по осям *x* и *y*, чтобы эта траектория поместилась в окне графика.

4. Подобрать такую начальную скорость спутника, чтобы период его обращения *T* был вдвое больше, чем при обращении по круговой орбите.

5. Исследовать, как влияет величина шага по времени на значение константы третьего закона Кеплера для разных траекторий.

Примечание. Количество траекторий при моделировании средствами MS Excel может быть уменьшено.



ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ «ПЕРВОЕ СЕНТЯБРЯ»
 ГАЗЕТА «ИНФОРМАТИКА»
 ОТДЕЛЕНИЕ ПЕДАГОГИЧЕСКОГО ОБРАЗОВАНИЯ ФГП МГУ ИМ. М.В. ЛОМОНОСОВА

ОБЪЯВЛЯЮТ НАБОР СЛУШАТЕЛЕЙ НА КУРСЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ НА 2006/2007 УЧЕБНЫЙ ГОД, ПЕРВЫЙ ПОТОК

Курсы проводятся в режиме дистанционного обучения (взаимодействие со слушателями производится посредством обычной или, при наличии у слушателя возможности, электронной почты). Продолжительность обучения — 7 месяцев, нормативный срок освоения учебного материала — 72 часа. Лекционный материал (8 лекций) и контрольные (2 работы) будут публиковаться на страницах газеты «Информатика» (для курса 07-007) или отправляться по почте (для остальных курсов). Итоговую работу слушатели будут выполнять в своих учебных заведениях.

После успешного окончания курсов слушатели получают удостоверение установленного образца о прохождении курсов повышения квалификации от Педагогического университета «Первое сентября» и Отделения педагогического образования ФГП МГУ им. М.В. Ломоносова.

Стоимость обучения составляет 790 рублей за один курс при оплате до 30 июня 2006 г. (990 рублей при оплате с 1 июля до 30 октября).

В 2006/2007 учебном году мы предлагаем четыре курса по вашей специальности:

| Код | Курс |
|--------|---|
| 07-001 | И.Г. Семакин. Информационные системы в базовом и профильном курсах информатики |
| 07-002 | Е.В. Андреева. Методика обучения основам программирования на уроках информатики |
| 07-006 | А.А. Дуванов. Основы web-дизайна и школьного «сайтостроительства» |
| 07-007 | И.Н. Фалина, В.Ф. Бурмакина. Как готовиться к тестированию по проверке ИКТ-компетенции школьников |

Мы также предлагаем один общепедагогический курс, предназначенный для всех работников образования:

| | |
|--------|--|
| 21-001 | С.С. Степанов. Теория и практика педагогического общения |
|--------|--|

Для зачисления на курсы необходимо прислать в Педагогический университет «Первое сентября» заявку. Пожалуйста, используйте только приведенный ниже бланк или его ксерокопию. Регистрация слушателей производится с 1 апреля по 30 сентября 2006 г. После регистрации вам будет выслан комплект документов с правилами обучения и счетом для оплаты. Вы оплатите счет лишь в том случае, если вас устроят предлагаемые условия (факт подачи заявки ни к чему не обязывает).

ЗАЯВКА Прошу выслать мне комплект документов для зачисления на курсы повышения квалификации. 07-08

ФАМИЛИЯ

ИМЯ

ОТЧЕСТВО

ИНДЕКС

АДРЕС

Телефон (с кодом города): () _____

Электронный адрес (если есть): _____

Место работы: _____

Должность: _____ Стаж работы по специальности: _____

ВНИМАНИЕ! К обучению на курсах повышения квалификации допускаются сотрудники образовательных учреждений, работающие по соответствующей специальности.

Я хочу пройти обучение по курсам (укажите коды выбранных вами курсов):

—

 —

 —

 —

Если вы обучались в 2005/2006 году на наших курсах, укажите, пожалуйста, ваш идентификатор:

**Заявки следует направлять по адресу: ул. Киевская, д. 24, г. Москва, 121165,
 Педагогический университет «Первое сентября». Справки по тел.: (495) 240-26-38**

Тема “Машина Тьюринга” в школьном курсе информатики

И.Н. ФАЛИНА,
Москва

Во многих учебниках по информатике при изучении понятия и свойств алгоритма присутствуют фразы такого содержания: “...существует много разных способов для записи одного и того же алгоритма, например, запись в виде текста, запись в виде блок-схемы, запись на каком-либо алгоритмическом языке, представление алгоритма в виде машины Тьюринга или машины Поста...”. К сожалению, такого типа фразы являются единственными, где упоминается машина Тьюринга. Без сомнения, объем часов, отводимых на изучение алгоритмов, не позволяет включать в эту тему еще и изучение способов записи алгоритма в виде машины Тьюринга. Но эта тема крайне интересна, важна и полезна для школьников, особенно увлекающихся информатикой.

Тема “Машина Тьюринга” может изучаться в 8–11-х классах в рамках темы “Информационные процессы. Обработка информации”, на факультативных занятиях, в системе дополнительного образования, например, в школах юных программистов. Изучение этой темы может сопровождаться компьютерной поддержкой, если у учителя есть программный тренажер-имитатор “Машина Тьюринга”. В классах с углубленным изучением программирования школьники могут самостоятельно написать программу “Машина Тьюринга”. В рамках этой статьи вашему вниманию предлагается практикум по решению задач на тему “Машина Тьюринга”. Теоретический материал по данной теме не раз печатался на страницах газеты “Информатика”, например, в № 3/2004 статья И.Н. Фапиной “Элементы теории алгоритмов”.

Краткий теоретический материал

Машина Тьюринга — это строгое математическое построение, математический аппарат (аналогичный, например, аппарату дифференциальных уравнений), созданный для решения определенных задач. Этот математический аппарат был назван “машиной” по той причине, что по описанию его составляющих частей и функционированию он похож на вычислительную машину. Принципиальное отличие машины Тьюринга от вычислительных машин состоит в том, что ее запоминающее устройство представляет собой бесконечную ленту: у реальных вычислительных машин запоминающее устройство может быть как угодно большим, но обязательно конечным.

Машину Тьюринга нельзя реализовать именно из-за бесконечности ее ленты. В этом смысле она мощнее любой вычислительной машины.

В каждой машине Тьюринга есть две части:

- 1) *неограниченная* в обе стороны *лента*, разделенная на ячейки;
- 2) *автомат* (головка для считывания/записи, управляемая программой).

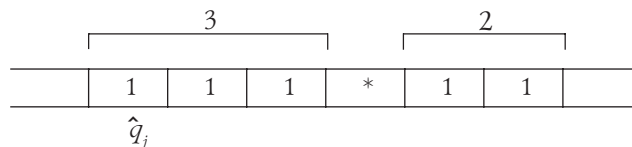
С каждой машиной Тьюринга связаны *два конечных алфавита*: алфавит входных символов

$A = \{a_0, a_1, \dots, a_m\}$ и алфавит состояний

$Q = \{q_0, q_1, \dots, q_p\}$. (С разными машинами Тьюринга могут быть связаны разные алфавиты A и Q .) Состояние q_0 называется *пассивным*. Считается, что если машина попала в это состояние, то она закончила свою работу. Состояние q_1 называется *начальным*. Находясь в этом состоянии, машина начинает свою работу.

Входное слово размещается на ленте по одной букве в расположенных подряд ячейках. Слева и справа от входного слова находятся только пустые ячейки (в алфавит A всегда входит пустая буква a_0 — признак того, что ячейка пуста).

Автомат может двигаться вдоль ленты влево или вправо, читать содержимое ячеек и записывать в ячейки буквы. Ниже схематично нарисована машина Тьюринга, автомат которой обозревает первую ячейку с данными.



Автомат каждый раз “видит” только одну ячейку. В зависимости от того, какую букву a_i он видит, а также в зависимости от своего состояния q_j , автомат может выполнять следующие действия:

- записать новую букву в обозреваемую ячейку;
- выполнить сдвиг по ленте на одну ячейку вправо/влево или остаться неподвижным;
- перейти в новое состояние.

То есть у машины Тьюринга есть три вида операций. Каждый раз для очередной пары (q_j, a_i) машина Тьюринга выполняет команду, состоящую из трех операций с определенными параметрами.

Программа для машины Тьюринга представляет собой таблицу, в каждой клетке которой записана команда.

| | a_0 | a_1 | ... | a_i | ... | a_m |
|-------|-------|-------|-----|---|-----|-------|
| q_1 | | | | | | |
| q_2 | | | | | | |
| ... | | | | | | |
| q_j | | | | $a_i \begin{Bmatrix} \Lambda \\ \Pi \\ H \end{Bmatrix} q_m$ | | |
| ... | | | | | | |
| q_p | | | | | | |

Клетка (q_j, a_i) определяется двумя параметрами — символом алфавита и состоянием машины. Команда представляет собой указание: куда передвинуть головку чтения/записи, какой символ записать в текущую ячейку, в какое состояние перейти машине. Для обозначения направления движения автомата используем одну из трех букв: “Л” (влево), “П” (вправо) или “Н” (неподвижен).

После выполнения автоматом очередной команды он переходит в состояние q_m (которое может в частном случае совпадать с прежним состоянием q_j). Следующую команду нужно искать в m -й строке таблицы на пересечении со столбцом a_i (букву a_i автомат видит после сдвига).

Договоримся, что когда лента содержит входное слово, то автомат находится против какой-то ячейки в состоянии q_1 . В процессе работы автомат будет перескакивать из одной клетки программы (таблицы) в другую, пока не дойдет до клетки, в которой записано, что автомат должен перейти в состояние q_0 . Эти клетки называются *клетками останова*. Дойдя до любой такой клетки, машина Тьюринга *останавливается*.

Несмотря на свое простое устройство, машина Тьюринга может выполнять все возможные преобразования слов, реализуя тем самым все возможные алгоритмы.

Пример. Требуется построить машину Тьюринга, которая прибавляет единицу к числу на ленте. Входное слово состоит из цифр целого десятичного числа, записанных в последовательные ячейки на ленте. В начальный момент машина находится против самой правой цифры числа.

Решение. Машина должна прибавить единицу к последней цифре числа. Если последняя цифра равна 9, то ее заменить на 0 и прибавить единицу к предыдущей цифре. Программа для данной машины Тьюринга может выглядеть так:

| | a_0 | 0 | 1 | 2 | 3 | 4 | ... | 7 | 8 | 9 |
|-------|-----------|-----------|-----------|-----------|-----------|-----------|-----|-----------|-----------|-----------|
| q_1 | 1 Н q_0 | 1 Н q_0 | 2 Н q_0 | 3 Н q_0 | 4 Н q_0 | 5 Н q_0 | ... | 8 Н q_0 | 9 Н q_0 | 0 Л q_1 |

В этой машине Тьюринга q_1 — состояние изменения цифры, q_0 — состояние останова. Если в состоянии q_1 автомат видит цифру 0..8, то он заменяет ее на 1..9 соответственно и переходит в состояние q_0 , т.е. машина останавливается. Если же он видит цифру 9, то заменяет ее на 0, сдвигается влево, оставаясь в состоянии q_1 . Так продолжается до тех пор, пока автомат не встретит цифру меньше 9. Если же все цифры были равны 9, то он заменит их нулями, запишет 0 на месте старшей цифры, сдвинется влево и в пустой клетке запишет 1. Затем перейдет в состояние q_0 , т.е. остановится.

Практические задания

1. На ленте машины Тьюринга содержится последовательность символов “+”. Напишите программу для машины Тьюринга, которая каждый второй символ “+” заменит на “-”. Замена начинается с правого конца последовательности. Автомат в состоянии q_1 обозревает один из символов указанной последовательности. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

2. Дано число n в восьмеричной системе счисления. Разработать машину Тьюринга, которая увеличивала бы заданное число n на 1. Автомат в состоянии q_1 обозревает некую цифру входного слова. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

3. Дана десятичная запись натурального числа $n > 1$. Разработать машину Тьюринга, которая уменьшала бы заданное число n на 1. Автомат в состоянии q_1 обозревает правую цифру числа. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

4. Дано натуральное число $n > 1$. Разработать машину Тьюринга, которая уменьшала бы заданное число n на 1, при этом в выходном слове старшая цифра не должна быть 0. Например, если входным словом было “100”, то выходным словом должно быть “99”, а не “099”. Автомат в состоянии q_1 обозревает правую цифру числа. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

5. Дан массив из открывающих и закрывающих скобок. Построить машину Тьюринга, которая удаляла бы пары взаимных скобок, т.е. расположенных подряд “()”.

Например, дано “) (() (()”, надо получить “) . . . ((”.

Автомат в состоянии q_1 обозревает крайний левый символ строки. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

6. Дана строка из букв “a” и “b”. Разработать машину Тьюринга, которая переместит все буквы “a” в левую, а буквы “b” — в правую части строки. Автомат в состоянии q_1 обозревает крайний левый символ строки. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

7. На ленте машины Тьюринга находится число, записанное в десятичной системе счисления. Умножить это число на 2. Автомат в состоянии q_1 обозревает крайнюю левую цифру числа. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

8. Даны два натуральных числа m и n , представленные в унарной системе счисления. Соответствующие наборы символов “|” разделены пустой клеткой. Автомат в состоянии q_1 обозревает самый правый символ входной последовательности. Разработать машину Тьюринга, которая на ленте оставит сумму чисел m и n . Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

9. Даны два натуральных числа m и n , представленных в унарной системе счисления. Соответствующие наборы символов “|” разделены пустой клеткой. Автомат в состоянии q_1 обозревает самый правый символ входной последовательности. Разработать машину Тьюринга, которая на ленте оставит разность чисел m и n . Известно, что $m > n$. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

10. На ленте машины Тьюринга находится десятичное число. Определить, делится ли это число на 5 без остатка. Если делится, то записать справа от числа слово “да”, иначе — “нет”. Автомат обозревает некую цифру входного числа. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.

Решения заданий

Задача 1

| | | | |
|-------|------------------|-----------|---|
| | a_0 | + | - |
| q_1 | $a_0 \wedge q_2$ | + П q_1 | |
| q_2 | $a_0 \vee q_0$ | + Л q_3 | |
| q_3 | $a_0 \vee q_0$ | - Л q_2 | |

В состоянии q_1 машина ищет правый конец числа, в состоянии q_2 — пропускает знак “+”, при достижении конца последовательности — останавливается. В состоянии q_3 машина знак “+” заменяет на знак “-”, при достижении конца последовательности она останавливается.

Задача 2

| | | | | | | |
|-------|-----------|-----------|-----------|-----------|-----|-----------|
| | a_0 | 0 | 1 | 2 | ... | 7 |
| q_1 | 1 Н q_0 | 1 Н q_0 | 2 Н q_0 | 3 Н q_0 | ... | 0 Л q_1 |

Решение этой задачи аналогично рассмотренному выше примеру.

Задача 3

| | | | | | | | |
|-------|-------|-----------|-----------|-----------|-----|-----------|-----------|
| | a_0 | 0 | 1 | 2 | ... | 8 | 9 |
| q_1 | | 9 Л q_1 | 0 Н q_0 | 1 Н q_0 | ... | 7 Н q_0 | 8 Н q_0 |

Состояние q_1 — уменьшаем младшую (очередную) цифру на 1. Если она не равна нулю, то после уменьшения сразу — останов, если же младшая цифра равна 0, то вместо нее пишем 9, смещаемся влево и вновь выполняем вычитание. В клетку $[a_0, q_1]$ машина Тьюринга никогда не попадет, поэтому ее можно не заполнять.

Задача 4 (усложнение задачи 3)

| | | | | | | | |
|-------|----------------|----------------|-----------|-----------|-----|-----------|-----------|
| | a_0 | 0 | 1 | 2 | ... | 8 | 9 |
| q_1 | | 9 Л q_1 | 0 Л q_2 | 1 Н q_0 | ... | 7 Н q_0 | 8 Н q_0 |
| q_2 | $a_0 \vee q_3$ | 0 Н q_0 | 1 Н q_0 | 2 Н q_0 | ... | 8 Н q_0 | 9 Н q_0 |
| q_3 | | $a_0 \vee q_0$ | | | | | |

Состояние q_1 — уменьшаем младшую (очередную) цифру на 1. Если она больше 1, то после уменьшения — сразу останов, если же младшая цифра равна 0, то вместо нее пишем 9, смещаемся влево и вновь выполняем вычитание. Если уменьшаемая цифра равна 1, то вместо нее пишем 0 и переходим в состояние q_2 .

Состояние q_2 — после записи “0” в каком-либо разряде надо проанализировать, не является ли этот ноль старшей незначащей цифрой (т.е. не стоит ли слева от него в записи выходного слова a_0).

Состояние q_3 — если записанный “0” является старшей незначащей цифрой, то его надо удалить из записи выходного слова.

Те клетки, в которые машина Тьюринга никогда не попадает, оставляем пустыми.

Задача 5

| | | | |
|-------|----------------|----------------|----------------|
| | a_0 | (|) |
| q_1 | $a_0 \vee q_0$ | (П q_2 |) П q_1 |
| q_2 | $a_0 \vee q_0$ | (П q_2 |) Л q_3 |
| q_3 | $a_0 \vee q_0$ | $a_0 \vee q_3$ | $a_0 \vee q_1$ |

Состояние q_1 : если встретили “(”, то сдвиг вправо и переход в состояние q_2 ; если встретили “ a_0 ”, то останов.

Состояние q_2 : анализ символа "(" на парность, в случае парности должны увидеть ")". Если парная, то возврат влево и переход в состояние q_3 .

Состояние q_3 : стираем сначала "(", затем ")" и переходим в q_1 .

Задача 6

Решение этой задачи обычно вызывает у школьников затруднение. При разборе решения этой задачи можно пойти, например, следующим путем.

Рассмотрите со школьниками следующие варианты входных слов и попросите их сформулировать, что должна делать машина Тьюринга, каков внешний вид выходного слова, чем с точки зрения машины Тьюринга эти варианты различаются:

$aaa \rightarrow$ выходное слово совпадает с входным, просматриваем входное слово до тех пор, пока оно не заканчивается.

$a \rightarrow$ выходное слово совпадает с входным, просматриваем входное слово до тех пор, пока оно не заканчивается.

$bbb \rightarrow$ выходное слово совпадает с входным, просматриваем входное слово до тех пор, пока оно не заканчивается.

$b \rightarrow$ выходное слово совпадает с входным, просматриваем входное слово до тех пор, пока оно не заканчивается.

$ab \rightarrow$ выходное слово совпадает с входным, просматриваем входное слово до тех пор, пока оно не заканчивается.

Результат обсуждения.

Машина Тьюринга должна "понимать", по цепочке каких букв она идет, т.е. у нее должно быть как минимум два состояния. Пусть состояние q_1 — движение по цепочке из букв "a", а q_2 — состояние движения по цепочке из букв "b". Заметим, что цепочка может состоять и из одной буквы. Если мы дошли до конца строки в состоянии q_1 или q_2 , т.е. встретили a_0 , машина должна остановиться, мы обработали всю строку.

Рассмотрим следующие варианты входных слов:

$bba \rightarrow abb$

$bbbaab \rightarrow aabbbb$

$aabbbbaab \rightarrow aaaabbbb$

Результат обсуждения. Первый вариант входного слова можно последовательно обработать следующим образом: $bba \rightarrow bbb \rightarrow$ вернуться к левому концу цепочки из букв "b" $\rightarrow abb$ (заменить первую букву в этой цепочке на "a"). Для выполнения этих действий нам потребуется ввести два новых состояния и, кроме того, уточнить состояние q_2 . Таким образом, для решения этой задачи нам нужно построить машину Тьюринга со следующими состояниями:

q_1 — идем вправо по цепочке букв "a". Если цепочка заканчивается a_0 , то переходим в q_0 ; если заканчивается буквой "b", то переходим в q_2 ;

q_2 — идем вправо по цепочке букв "b", если цепочка заканчивается a_0 , то переходим в q_0 ; если заканчивается "a", то заменяем букву "a" на "b",

переходим в состояние q_3 (цепочку вида $\underbrace{b \dots ba}_n$ заменили на цепочку вида $\underbrace{b \dots b}_{n+1}$);

q_3 — идем влево по цепочке букв "b" до ее левого конца. Если встретили a_0 или "a", то переходим в q_4 ;

q_4 — заменяем "b" на "a" и переходим в q_1 (цепочку вида $\underbrace{b \dots b}_{n+1}$ заменяем на цепочку вида $a \underbrace{b \dots b}_n$).

| | a_0 | a | b |
|-------|----------------------|-----------|-----------|
| q_1 | $a_0 \text{ H } q_0$ | a П q_1 | b П q_2 |
| q_2 | $a_0 \text{ H } q_0$ | b Л q_3 | b П q_2 |
| q_3 | $a_0 \text{ П } q_4$ | a П q_4 | b Л q_3 |
| q_4 | | | a П q_1 |

Задача 7

| | a_0 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|-------|----------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| q_1 | $a_0 \text{ Л } q_2$ | 0 П q_1 | 1 П q_1 | 2 П q_1 | 3 П q_1 | 4 П q_1 | 5 П q_1 | 6 П q_1 | 7 П q_1 | 8 П q_1 | 9 П q_1 |
| q_2 | $a_0 \text{ H } q_0$ | 0 Л q_2 | 2 Л q_2 | 4 Л q_2 | 6 Л q_2 | 8 Л q_2 | 0 Л q_3 | 2 Л q_3 | 4 Л q_3 | 6 Л q_3 | 8 Л q_3 |
| q_3 | 1 H q_0 | 1 Л q_2 | 3 Л q_2 | 5 Л q_2 | 7 Л q_2 | 9 Л q_2 | 1 Л q_3 | 3 Л q_3 | 5 Л q_3 | 7 Л q_3 | 9 Л q_3 |

состояние q_1 — поиск правой (младшей) цифры числа;

состояние q_2 — умножение очередной цифры числа на 2 без прибавления 1 переноса;

состояние q_3 — умножение очередной цифры числа на 2 с прибавлением 1 переноса.

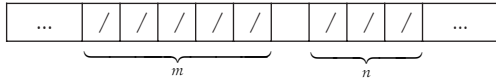
Задача 8

Машина Тьюринга для этой программы выглядит тривиально просто — в ней всего одно состояние. Такая машина Тьюринга выполняет следующие действия: стирает самый правый штрих, ищет разделитель (пустую ячейку) и в эту пустую ячейку помещает штрих, тем самым сформирована непрерывная последовательность штрихов длины $n + m$.

| | a_0 | / |
|-------|-----------|----------------------|
| q_1 | / H q_0 | $a_0 \text{ Л } q_1$ |

Однако, как ни странно, решение этой задачи вызывает большие трудности. Очень часто ученики строят машину Тьюринга, которая выполняет цик-

лические действия: последовательно пододвигают правые n штрихов к левым.



В этом случае их программа выглядит следующим образом:

| | | |
|-------|----------------------|-----------------|
| | a_0 | / |
| q_1 | $a_0 \text{ Н } q_2$ | / Λq_1 |
| q_2 | / П q_2 | a_0 П q_3 |
| q_3 | $a_0 \text{ Н } q_0$ | / Λq_2 |

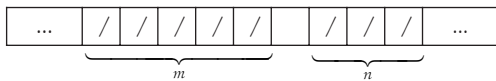
состояние q_1 — поиск разделителя;
 состояние q_2 — передвинули штрих;
 состояние q_3 — проверка на конец (все ли штрихи передвинули).

На примере этой задачи четко видно, как часто дети пытаются решить задачу уже знакомыми способами. Мне кажется, что, предлагая ученикам задачи на составление машин Тьюринга, мы развиваем способность к нахождению необычных решений, развиваем способность творчески думать!

Задача 9

Эта задача кажется школьникам достаточно легкой, но трудности возникают с остановом машины Тьюринга. Ниже приведен один из возможных вариантов машины Тьюринга для этой задачи.

Идея решения (условие останова). На ленте есть два исходных массива штрихов.



Штрихи начинаем стирать с левого конца массива m . И поочередно стираем самый левый штрих в массиве m и самый правый штрих в массиве n . Но прежде чем стереть правый штрих в массиве n , проверяем, единственный он (т.е. последний, который надо стереть) или нет.

Опишем сначала состояния машины Тьюринга, которые необходимы для решения нашей задачи, а затем составим программу-таблицу.

Состояние q_1 — поиск разделителя между массивами штрихов при движении справа налево;
 состояние q_2 — поиск левого штриха в массиве m ;
 состояние q_3 — удаление левого штриха в массиве m ;
 состояние q_4 — поиск разделителя при движении слева направо;
 состояние q_5 — поиск правого штриха в массиве n ;

состояние q_6 — проверка единственности этого штриха в массиве n , т.е. определяем, был ли он последним;

состояние q_7 — если он был последним, то останов, иначе переход на новый цикл выполнения алгоритма.

| | | |
|-------|----------------------|-------------------|
| | a_0 | / |
| q_1 | $a_0 \Lambda q_2$ | / Λq_1 |
| q_2 | a_0 П q_3 | / Λq_2 |
| q_3 | | a_0 П q_4 |
| q_4 | a_0 П q_5 | / П q_4 |
| q_5 | $a_0 \Lambda q_6$ | / П q_5 |
| q_6 | | $a_0 \Lambda q_7$ |
| q_7 | $a_0 \text{ Н } q_0$ | / Λq_1 |

Задача 10

При решении этой задачи следует обратить внимание на правильное выписывание алфавита:

$A = \{a_0, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, \Delta, \Lambda, \text{Н}, \text{Е}, \text{Т}\}$.

| | | | | | | | | | | | | | |
|-------|-------------------|-----------|-----------|-----------|-----|-----------|-----|-----------|----------|-----------|---|---|---|
| | a_0 | 0 | 1 | 2 | ... | 5 | ... | 9 | Δ | Λ | Н | Е | Т |
| q_1 | $a_0 \Lambda q_2$ | 0 П q_1 | 1 П q_1 | 2 П q_1 | ... | ... | ... | 9 П q_1 | | | | | |
| q_2 | | 0 П q_3 | 1 П q_5 | 2 П q_5 | ... | 5 П q_3 | ... | 9 П q_5 | | | | | |
| q_3 | Δ П q_4 | | | | | | | | | | | | |
| q_4 | Λ Н q_0 | | | | | | | | | | | | |
| q_5 | Н П q_6 | | | | | | | | | | | | |
| q_6 | Е П q_7 | | | | | | | | | | | | |
| q_7 | Т Н q_0 | | | | | | | | | | | | |

Состояние q_1 — поиск правого конца числа;
 состояние q_2 — анализ младшей цифры числа; если она равна "0" или "5", т.е. число делится на 5, то переход в состояние q_3 , иначе переход в состояние q_5 ;
 состояние q_3 — запись буквы "Д" справа от слова на ленте;
 состояние q_4 — запись буквы "А" справа от слова и останов машины;
 состояние q_5 — запись буквы "Н" справа от слова;
 состояние q_6 — запись буквы "Е" справа от слова;
 состояние q_7 — запись буквы "Т" справа от слова и останов машины.

Свойства машины Тьюринга как алгоритма

На примере машины Тьюринга хорошо прослеживаются свойства алгоритмов. Попросите учащихся показать, что машина Тьюринга обладает всеми свойствами алгоритма.

Дискретность. Машина Тьюринга может перейти к $(k + 1)$ -му шагу только после выполнения k -го шага, т.к. именно k -й шаг определяет, каким будет $(k + 1)$ -й шаг.

Понятность. На каждом шаге в ячейку пишется символ из алфавита, автомат делает одно движение (Л, П, Н), и машина Тьюринга переходит в одно из описанных состояний.

Детерминированность. В каждой клетке таблицы машины Тьюринга записан лишь один вариант действия. На каждом шаге результат определен однозначно, следовательно, последовательность шагов решения задачи определена однозначно, т.е. если машине Тьюринга на вход подадут одно и то же

ОТ РЕДАКЦИИ

Все приведенные в статье задачи можно решить просто в тетради, начертив информационную ленту и программу-таблицу. Но можно сделать этот процесс более увлекательным и наглядным: воспользоваться машинной реализацией — интерпретатором машины Поста и машины Тьюринга “Algo2000”, созданным Радиком Зартдиновым. Программа обладает интуитивно понятным интерфейсом, и требования у нее самые умеренные: компьютер IBM PC AT 486 и выше, наличие операционной системы Windows'95/98/NT.

Посмотрим в общих чертах, как работает “Algo2000”.

В меню программы выберем пункт **Интерпретатор** и укажем, с какой машиной мы хотим работать (в нашем случае это “машина Тьюринга”).

Перед нами появится поле машины Тьюринга.

Теперь необходимо задать внешний алфавит, т.е. в строке **Внешний алфавит** указать, какие символы в него входят (если строка **Внешний алфавит** не видна, нужно выбрать пункт меню **Вид | Внешний алфавит**). Каждый символ можно указать только один раз. После окончания ввода внешнего алфавита формируется первый столбец таблицы: он заполняется символами внешнего алфавита в том же порядке. При редактировании внешнего алфавита автоматически изменяется таблица: вставляются, удаляются или меняются местами строки.

Не забудем, что нужно как-то расставить символы внешнего алфавита по секциям ленты (можно все секции оставить пустыми) и поставить каретку против одной из секций, т.е. надо задать программу и некоторое состояние машины.

Теперь можно приступить непосредственно к записи алгоритма решения задачи. Он задается в виде таблицы: в каждый столбец верхней строчки заносятся символы внутреннего алфавита, в каждую строчку первого столбца — символы внешнего алфавита. В ячейках на пересечении других столбцов и строчек помещаются команды. Если на пересечении какой-либо строки и какого-либо столбца мы получим пустую клетку, то это означает, что в данном внутреннем состоянии данный символ встретиться не может.

Например, мы составляем алгоритм нахождения разности двух целых положительных чисел (в десятичной

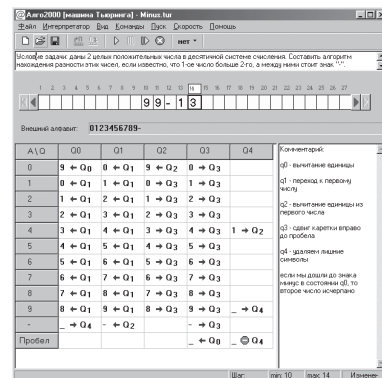
входное слово, то выходное слово каждый раз будет одним и тем же.

Результативность. Содержательно результаты каждого шага и всей последовательности шагов определены однозначно, следовательно, правильно написанная машина Тьюринга за конечное число шагов перейдет в состояние q_0 , т.е. за конечное число шагов будет получен ответ на вопрос задачи.

Массовость. Каждая машина Тьюринга определена над всеми допустимыми словами из алфавита, в этом и состоит свойство массовости. Каждая машина Тьюринга предназначена для решения одного класса задач, т.е. для каждой задачи пишется своя (новая) машина Тьюринга.

системе счисления), если известно, что первое число больше второго, а между ними стоит знак минус.

Поле программы будет выглядеть так:



Формат команды в каждой ячейке — aKq . Здесь: a — новое содержание текущей ячейки (новый символ внешнего алфавита, который заносится в текущую ячейку), K — команда лентопротяжного механизма машины Тьюринга (влево, вправо, стоп), q — новое внутреннее состояние машины Тьюринга.

Кнопка запустит программу. Если выполнение не было приостановлено, то оно всегда начинается с нулевого внутреннего состояния Q0.

Программу можно выполнить по шагам. Для этого нажмите на кнопку на панели инструментов (если кнопки не видны, нужно выбрать пункт меню **Вид | Панель инструментов**) или выберите в главном меню **Пуск | Пошагово**. Если необходимо полностью прервать выполнение программы, то это можно сделать с помощью кнопки на панели инструментов или с помощью главного меню (**Пуск | Прервать**). Пункт меню **Скорость** позволяет регулировать скорость выполнения программы.

Выполнение программы будет идти до тех пор, пока не встретится команда “Стоп” или не возникнет какая-нибудь ошибка.

При возникновении вопросов в ходе работы с программой-интерпретатором обращайтесь к справочному файлу *Algo2000.hlp*. Его, так же, как и саму программу “Algo2000”, можно найти на сайте газеты “Информатика” <http://inf.1september.ru> в разделе “Download”.

“Как это делаю я”

Методический конкурс для учителей информатики

Итоги пятого тура “Кабинет — наш второй дом”

Подводя итоги пятого тура нашего конкурса, хочется прежде всего отметить неформальное отношение наших коллег к своему кабинету. Для публикации мы выбрали восемь работ, которые, на наш взгляд, представляют весь спектр присланных материалов.

Так, М.М. Барсукова, например, четко и емко представила основные акценты кабинета информатики. В том числе сформулировала и один из вопросов, возникающих перед учителем: “Нужны ли в современном кабинете стенды? Или это “пережиток” прошлого?”. Каждый решает этот вопрос по-своему. Можно воспользоваться старыми добрыми фломастерами, а можно изготовить красочные наглядные стенды с использованием возможностей своей компьютерной техники (посмотрите, как это делает Н.И. Ермакова).

Похоже, что особой популярностью среди учителей информатики пользуется потолочная плитка :). Оказалось, что это очень удобный материал для создания легких и мобильных стендов (один из многочисленных примеров представлен в кабинете В.В. Меньшикова).

А вот счастливые обладатели мультимедийных проекторов считают, что все же учебный материал лучше продемонстрировать не на стенде, а на экране... Хорошо, конечно, когда проектор есть! Впрочем, и в таких кабинетах без стенда с текущей информацией не обходится.

Опять-таки, если позволяют средства, кабинет можно превратить в настоящую творческую мастерскую для ребят: создание мультимедийных проектов, электронных газет, поиск интересной информации в Интернете и т.д.

К сожалению, далеко не все сегодня могут позволить себе подобное. Но газету выпускать — и она станет частью кабинета — могут все! А если нет доступа к Интернету, то новую интересную информацию можно почерпнуть из газет-журналов. И этими находками поделиться со всеми.

Итак, все согласилось с тем, что кабинет должен быть комфортен и ученикам, и учителю; он должен быть уютным и функциональным одновременно. А раз так, нельзя забывать ни о каких мелочах (как справедливо заметил В.В. Пичугин, “мелочах ли?”).

Изменения в кабинетах происходят постоянно: они живут, растут и изменяются вместе с нами. Очень точно сказала Е.Н. Смирнова: “Появляются

новые возможности, приходят новые ученики, возникают новые идеи. Мне кажется, что в любом случае главное — любовь учителя к своему предмету и неиссякающий интерес к нему. Тогда и ученикам будет интересно, и кабинет станет функциональным и удобным”.

Все присланные материалы, как обычно, размещены в разделе конкурса на сайте “Информатики” <http://inf.1september.ru>.

Для публикации мы выбрали материалы:

БАРСУКОВОЙ Марии Михайловны, СОШ № 48 (г. Оренбург);

МЕНЬШИКОВА Виталия Владимировича, СОШ № 5 (г. Сегежа, Республика Карелия);

ЕРМАКОВОЙ Надежды Игоревны, Центр образования № 2 (г. Оренбург);

ЧЕРНОБАБОВОЙ Клавдии Владимировны, гимназия № 8 (г. Сочи);

СМИРНОВОЙ Елены Николаевны, СОШ № 24 (г. Брянск);

ПИЧУГИНА Виталия Владимировича, СОШ (р.п. Пинеровка, Саратовская обл.);

ИСАКОВОЙ Ирины Сергеевны и КРЫЛОВОЙ Людмилы Николаевны, СОШ № 1 (п. Лимбяха, г. Новый Уренгой);

МОЖАРОВОЙ Жаннны Вадимовны, СОШ № 17 (г. Алапаевск, Свердловская обл.).

Кстати, невозможно не заметить особого пристрастия многих хозяев компьютерных кабинетов к кактусам. Они особенно красивы? Или неприхотливы? А может, сдерживают излишне резвых учеников?:-) Что же касается способности этих растений поглощать вредные электромагнитные излучения, то на сегодняшний день она признана весьма призрачной. “Нет никаких научных подтверждений того, что кактусы обладают каким-то особым влиянием на электромагнитное излучение компьютерного монитора. Однако избавляться от них не надо, поскольку любые зеленые растения увеличивают влажность в помещении, изменяя тем самым аэромонный состав воздуха. От этого снижается уровень статистического электричества. Так что комнатные цветы вблизи рабочего места не только красивы, но и полезны” (консультант отдела санитарного надзора ТУ “Роспотребнадзор” Н.Ярикова).

Результаты шестого тура будут опубликованы в № 10/2006.

Участники пятого тура

Барсукова М.М., СОШ № 48 (г. Оренбург);
 Брюханова Т.А., СОШ № 19 (с. Лугавское, Красноярский край);
 Вяткина Л.В., СОШ (с. Нердва, Пермский край);
 Гоняева Т.А., СОШ (с. Зерновое, Иркутская обл.);
 Ермакова Н.И., Центр образования № 2 (г. Оренбург);
 Жданова О.Б., Межшкольный учебный комбинат (г. Новоуральск, Свердловская обл.);
 Жуланова С.В., СОШ № 3 (г. Березники, Пермская обл.);
 Загвоздина Л.З., СОШ № 14 (г. Братск, Иркутская обл.);
 Зайцева Л.И., школа-интернат (п. Горноправдинск, Тюменская обл.);
 Зуева И.Ю., гимназия № 7 им. В.М. Воронцова (г. Воронеж);
 Исаева А.В., СОШ (п. Строительный, Тульская обл.);
 Исакова И.С., Крылова Л.Н., СОШ № 1 (п. Лимбьяха, г. Новый Уренгой);
 Костарева Н.П., СОШ № 3 (г. Оса, Пермская обл.);
 Меньшиков В.В., СОШ № 5 (г. Сегежа, Республика Карелия);
 Можарова Ж.В., СОШ (г. Алапаевск, Свердловская обл.);
 Никанорова Л.Н., СОШ № 1194 (г. Москва);
 Новоселова Е.П., СОШ (п. Зеленоборск, Тюменская обл.);
 Обухова И.В., МНОУ "ЛИЦЕЙ" (г. Кемерово);
 Перминова О.И., СОШ № 21 (г. Канск, Красноярский край);
 Пичугин В.В., СОШ (р.п. Пинеровка, Саратовская обл.);
 Плотноикова С.В., СОШ (с. Верхнемарково, Иркутская обл.);
 Семенова З.С., СОШ № 7 (г. Заинск, Республика Татарстан);
 Синица А.А., СОШ № 2 им. Н.П. Массонова (г. Свислочь, Республика Беларусь);
 Смирнова Е.Н., СОШ № 24 (г. Брянск);
 Хмелькова О.П., СОШ № 882 (г. Москва);
 Чапкевич И.М., лицей № 4 (г. Орел);
 Чернобабова К.В., гимназия № 8 (г. Сочи);
 Широкова Л.В., Лицей № 10 (г. Волгоград);
 Ямкина Е.В., СОШ № 37 (г. Ульяновск).

Кабинет — второй дом

М.М. Барсукова,
г. Оренбург

Работая в кабинете более 30 часов в неделю, начинаешь считать его своим родным домом. А раз это второй дом — значит, он должен быть уютным, удобным и функциональным. Уют создать не сложно: чистота, красивые шторы на окнах, цветы, аквариум в качестве увлажнителя... С функциональностью сложнее. Кабинет должен быть максимально комфортным и удобным для преподавателя: все под рукой, все в нужном месте и в нужное время.

Многие учителя считают, что стенды в кабинете информатики не особенно-то и нужны. Ведь с использованием ПК можно показать все, что угодно. Но мне кажется, что это не совсем так. Необходимо использовать пространство кабинета в полной мере. Стенды — не пережиток прошлого, а необходимое содержание любого кабинета.

На мой взгляд, в каждом кабинете должны присутствовать следующие стенды:

1. Техника безопасности.
2. Алгоритмические структуры.
3. История развития вычислительной техники.
4. Творчество учащихся.
5. Экзамены.
6. Устройство компьютера.

Стенд "Техника безопасности". Будет лучше, если он будет выполнен в занимательно-шутливой форме с использованием смешных ситуаций, которые не должны происходить на уроке. Кроме того, не стоит забывать, что это все серьезно и несет опасность для жизни. Поэтому правила, которые утверждены вашим ОУ, должны быть рядом.

Стенд "Алгоритмические структуры". Несмотря на то что программирование и алгоритмизация сейчас не на первом месте, это вовсе не означает, что детей не нужно учить мыслить. Знание основных структур необходимо. Если сделать структуры доста-

точно крупными, можно использовать стенд для ответа учащихся на уроке.

Стенд “История развития вычислительной техники”. Наверное, редкий учитель не имеет такого стенда в своем классе: очень удобно использовать персоналии при объяснении теоретического материала. Очень жаль, что мне пока не встречались типографские наборы портретов ученых, имеющих отношение к информатике. Поэтому качество портретов иногда не соответствует необходимому стандарту.

Стенд “Творчество учащихся”. Ребятам всегда хочется увидеть результаты своего труда, показать свои работы одноклассникам, учителям, родителям. Называть этот стенд можно по-разному. Например, в нашем кабинете он называется “Компьютер с нами, он везде — в учебе, творчестве, игре”. Этот стенд — сменный. На его “экране” можно отражать текущие результаты проходящих конкурсов. Кроме этого, на нем располагаются примеры рефератов, сообщений учащихся. У нас по краю стенда прибиты гвоздики со шляпками, и на них ребята вешают так называемые “Тетради сбора информации”. Каждый ребенок собирает в эту тетрадь сообщения о компьютерах, которые он встретил в печатных изданиях. Материалы вклеиваются в тетрадь и вывешиваются на всеобщее обозрение (любой учащийся может прочитать представленные материалы на перемене). Учитель с определенной периодичностью проверяет их и выставляет дополнительные оценки. Наиболее интересные статьи, с точки зрения учителя или ученика, могут быть представлены на уроках в качестве сообщений.

Стенд “Экзамены”. Этот стенд в любом учебном кабинете является ориентиром для учащихся. Становится ясно, какие вопросы могут быть на экзамене и стоит ли выбирать этот экзамен, если предложено сделать выбор.

Стенд “Устройство компьютера”. Почти все учащиеся, покинув школу, будут иметь дело с компьютерами в повседневной жизни, но это вовсе не означает, что они будут заниматься “железом”. Для многих этот стенд, быть может, послужит единственным представлением о внутреннем устройстве компьютера.

Кабинет информатики — не только учебный кабинет. Он, как правило, является и центром методической работы, где находят свое воплощение и применение практически все идеи школы. Поэтому в кабинете могут находиться стенды, которые относятся к общепедагогическим технологиям. Если в вашей школе ведется работа по созданию электронных учебников, газет — выделите и для этого место в вашем кабинете. Для молодых специалистов будет полезен стенд с названием “Раздаточный материал”, где представле-

ны примеры лабораторных работ по разным темам. Учащиеся тоже с удовольствием рассматривают этот стенд, находя в нем свои задания и узнавая дальнейшие.

Функциональность и уют кабинета информатики

В.В. Меньшиков,

г. Сегежа, Республика Карелия

Современный компьютерный класс в нашей школе был открыт в 2001 году. Сейчас в нем имеются 10 компьютеров, 2 принтера (черно-белый лазерный и цветной струйный), сканер, мультимедийный проектор с экраном. Все компьютеры в кабинете объединены в локальную сеть.

По моему мнению, оформление компьютерного класса должно нести не только функциональную нагрузку. Кабинет должен быть удобным и комфортным для учителя и учеников, ну и по-домашнему уютным, только тогда учитель сможет полноценно работать, а ученики — плодотворно и с удовольствием учиться. Школа у нас старая, а компьютерный класс должен быть современным, поэтому пришлось решить ряд вопросов:

Вопрос 1. Расстановка мебели.

В классе был сделан так называемый “подиум”. Совершенно негде было установить учительский стол и организовать рабочее место учителя. После долгих раздумий было решено расположить все прямо на подиуме вполоборота к учащимся. Со временем оказалось, что это очень удобно.

Компьютерные столы расставили по периметру. В центре поставили ученические парты.

Вопрос 2. Стенды.

Как и в любом другом кабинете, в компьютерном классе должны быть стенды... Тем стендам, которые были в кабинете первоначально, уже более 30 лет. Несмотря на вполне приличный вид, я от них полностью отказался, т.к. они морально устарели. Все стенды я изготовил из потолочных плиток, которые прикрепил к стенам с помощью специального клея. Стендов у меня пять: три больших (1 × 1 м) и два маленьких (0,5 × 1 м). Маленькие стенды — навесные.

Стенд № 1 является постоянным. На нем располагается расписание звонков, график работы компьютерного класса, официальная инструкция по технике безопасности, инструкция по пожарной безопасности, комплексы упражнений для глаз, комплексы упражнений для проведения физкультминуток, правила поведения в школе.

Стенд № 2 также является постоянным.



Посвящен он правилам техники безопасности, но в данном случае все правила даны в стихотворной форме.

Стенд № 3 постоянно меняется. Здесь вывешиваются работы учащихся, варианты заданий для поступающих в высшие учебные заведения, материал для любознательных, материалы газеты “Информатика”, журналов “Информатика и образование”, “Компьютерра” и др.

Два маленьких стенда, находящиеся над доской, рассказывают о правилах набора текста и также являются постоянными.



Кроме того, эти два стенда служат маскировкой проводов, идущих над доской. Стенды подвесные: крепятся они на плотную прозрачную нить с помощью пластиковых бельевых прищепок. Смотрится очень эффектно и достаточно эстетично.

Вопрос 3. Тетради.

В кабинете есть только два встроенных шкафчика, в которых все нужно было расположить достаточно компактно. Я работаю во многих классах, и тетрадей для проверочных работ у меня тоже очень много. Чтобы все они не занимали много места, я из плотной бумаги изготовил специальные лотки (фотографии и даже чертежи © приведены на странице конкурса на сайте “Информатики” <http://inf.1september.ru>).

Уют создают цветы, которых в кабинете достаточно много. На столах учащихся в горшках растут

кактусы. Кроме того, в кабинете имеется медиатека, которая постоянно дополняется работами учащихся. Выпускаются диски с созданными учащими презентациями и web-сайтами по различным предметам и темам школьной программы.



Новый стенд? Легко!

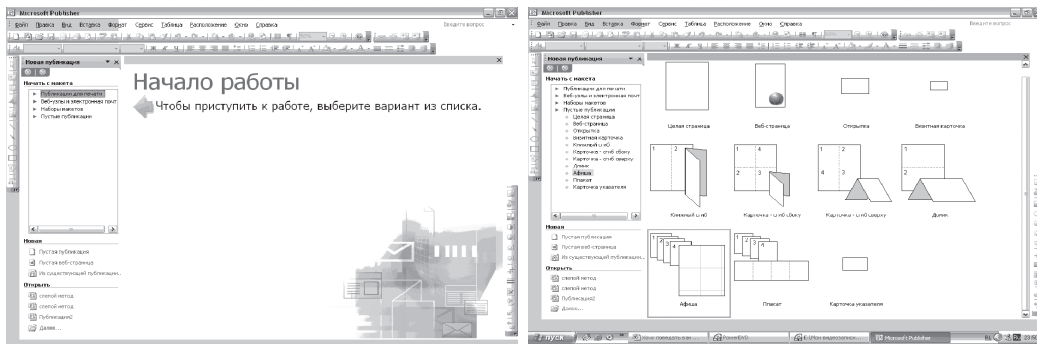
Н.И. Ермакова,
г. Оренбург

Для оформления кабинетов мы используем потолочную пенопластовую плитку. Как я только не располагала ее на стенах кабинета: по парам и по диагонали, вдоль и поперек, даже одно время в кабинете висели две “ханойские башенки” из потолочной плитки.

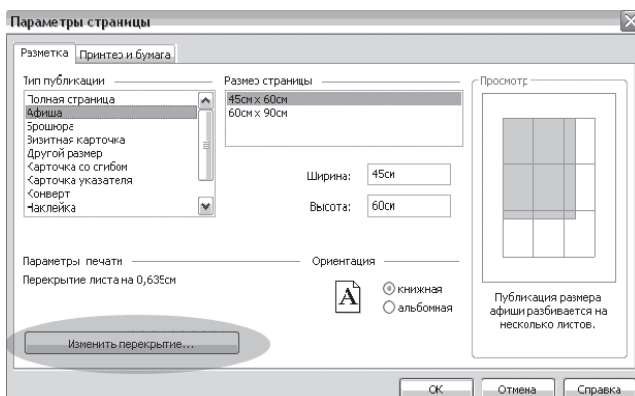
Во время последних поисков очередного удачного расположения плитки на стене кабинета я натолкнулась на забытые старенькие стенды. Информатика, изображенная на них, была классической и не потеряла своей актуальности, но внешний вид оставался желать лучшего: пожелтевший со временем ватман, потускневшие цвета. В голову пришла идея о восстановлении стенда. Но покупать ватман, фломастеры, маркеры, а тем более что-то еще рисовать... Не хотелось. Ведь есть компьютер! Можно было, конечно, распечатать информацию на отдельных листочках, а потом в художественном порядке расположить на стенде. Но хотелось бы, чтобы вся информация была на одном большом плакате. Можно ли воплотить эту идею в жизнь? Мне удалось! Причем за рекордно короткие сроки (один день!) и с относительно небольшими затратами.

На помощь пришла прикладная программа **MS Office Publisher**. Предлагаю подробный алгоритм, по которому я создаю свои “типографские” плакаты.

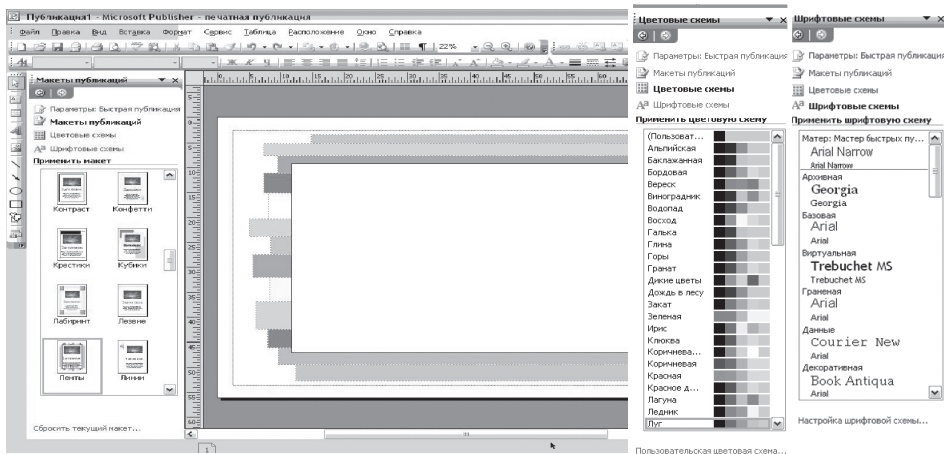
1. Откройте программу **Microsoft Publisher**.
2. В появившемся диалоговом окне выберите **Пустые публикации | Афиша**.



3. Установите параметры страницы, выбрав команду **Файл | Параметры страницы**. Здесь можно установить ширину и высоту плаката, а также выбрать ориентацию листа (книжную или альбомную). Не забудьте воспользоваться кнопкой **Изменить перекрытие** (те участки, которые вы будете склеивать между собой; по умолчанию стоят очень большие значения, достаточно установить 0,1–0,4).



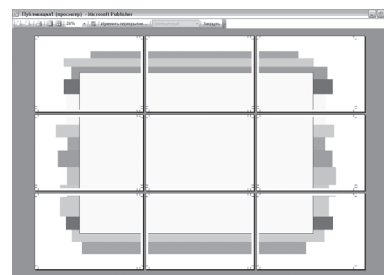
4. При желании можно применить один из предложенных **Макетов публикации** с разнообразными **Цветовыми схемами**, которые имеют экзотические названия, а также воспользоваться **Шрифтовыми схемами**. Таким образом, при необходимости можно оживить любой плакат соответствующим фоном, цветом и текстом.



5. Теперь можно получившийся фон заполнить подобранным материалом. Но не забывайте выполнять команду **Файл | Предварительный просмотр**.

6. Распечатайте полученный плакат, отрежьте лишние по-

лости бумаги и аккуратно склейте, взяв за основу принцип наклейки фотообоев. Если вы будете смотреть на плакат с левой стороны, то от каждого листа отрежьте отмеченные компьютером полоски бумаги соединения справа и сверху, а если с правой стороны, то надо отрезать слева и сверху (по умолчанию считаем, что плакат вы повесите выше уровня глаз, именно поэтому мы будем отрезать в обоих случаях сверху). Все эти хитрости необходимы для того, чтобы сделать места склейки листов практически незаметными. И пожалуйста, возьмите на вооружение клей ПВА, а не клеящий карандаш (он растягивает бумагу).



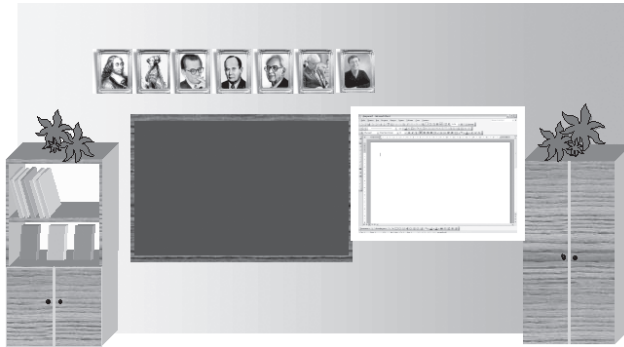
В результате вы за короткий срок и с небольшими затратами получите красочно оформленное наглядное пособие.

Благодаря такой технологии я существеннополнила свой “запас” наглядных пособий и стендов.

Обо всех рассказывать не буду, предложу вашему вниманию лишь один стенд, который играет в моем кабинете существенную роль.

К сожалению, без цифрового фотоаппарата я не смогла сделать фотографию стенда. Пришлось использовать графические возможности текстового редактора MS Word; получилась “виртуальная” фотография.

Обратите внимание на огромный плакат справа от доски: я сделала снимок с экрана (**Print Screen**) интерфейса текстового редактора и распечатала его на 24 листах.



Вы спросите меня, зачем нужен такой плакат?

Во-первых, у меня в распоряжении нет проектора, на котором можно было бы показать интерфейс текстового редактора. Во-вторых, при объяснении некоторых тем текстового редактора приходится рисовать значки на доске, а здесь достаточно показать указкой. В-третьих, плакат таких размеров будоражит воображение учащихся и учителей. Ученики интересуются процессом создания плаката и понимают, что даже на обычном принтере с подачей листов формата А4 можно создавать плакаты, афиши, постеры и т.д. Учителя-предметники в суть процесса не вникают, но атакуют заказами на изготовление наглядных пособий. В-четвертых, если ученик не может найти тот или иной инструмент, я на плакате показываю его расположение; он ищет его на экране монитора, а это уже не так просто, как если бы я навела указатель мыши на соответствующую пиктограмму на ПК. В-пятых, очень удобно проводить опрос (впрочем, как и при использовании любого другого наглядного пособия). В-шестых, такой плакат можно использовать и при изучении других прикладных программ со схожим интерфейсом (например, табличного процессора MS Excel).

И, наконец, на “рабочее поле” плаката я прикалываю опорные конспекты, другие наглядные пособия (меньшего размера) и т.д.

Надеюсь, что моя идея поможет вам в работе, и с помощью описанного выше алгоритма вы сможете заменить многие стенды на более красочные и быстро изготавливаемые наглядные пособия на любую тему и любого нестандартного размера.

Творческая рабочая мастерская

К.В. Чернобабова,
г. Сочи

Раньше кабинет информатики был чем-то таинственным и загадочным.

В первые годы изучения информатики акцент делался на алгоритмизацию и программирование (на

“Корветах” и “Ямахах” трудно было сделать что-либо другое). В кабинетах, как правило, присутствовали плакаты “Структура и принципы работы ЭВМ”, “Программный принцип работы ЭВМ”, стенды учебного характера.

Сегодня вектор развития образовательного процесса направлен на освоение информационно-коммуникационных технологий. Компьютерная грамотность выпускников должна быть достаточна для того, чтобы свободно владеть компьютером. Эта потребность продиктована временем. Учащиеся должны овладеть офисным пакетом программ, научиться работать с локальной и глобальной сетью, познакомиться с возможностями профессиональных программ по обработке графических, аудио- и видео-файлов.

На мой взгляд, теперь учебный материал лучше демонстрировать учащимся не на стенде, а на экране с помощью мультимедийного проектора. Можно разработать к уроку презентацию, которая позволит сделать процесс обучения интересным, наглядным; можно воспользоваться готовыми информационно-обучающими программами, над которыми трудились специалисты.

Можно подготовить к практическим работам индивидуальные технологические карты, которые представляют собой напечатанные с помощью ПК книжечки-инструкции.

Теперь, когда ПК используется прежде всего как инструмент информационных технологий, я стремлюсь сделать компьютерный кабинет *творческой рабочей мастерской*, в которой создается атмосфера сотрудничества и взаимопомощи. Здесь рождаются идеи и воплощаются в мультимедийные компьютерные проекты.

Я стремлюсь к тому, чтобы в кабинете было наиболее “продвинутое” программное обеспечение “на все случаи жизни” (Pinnacle Studio, Macromedia Flash, PhotoShop, PageMaker, риппинг дисков, граберы, декодеры, конверторы, разные среды программирования и др.), дополнительное оборудование (сканер, цветной принтер, CD и DVD-writer, мультимедийный проектор, наушники и микрофоны, локальная сеть, выход в Интернет) и т.д.

Здесь ребята творят, фантазируют. В нашем кабинете они могут:

- создавать проекты в виде мультимедийных презентаций, web-сайтов для конкурсов, фестивалей, выставок и др.;
- изменять форматы музыкальных файлов, чтобы на школьном концерте спеть со сцены песню под созданный аккомпанемент;
- осуществлять ежемесячный выпуск гимназической газеты “Школьный бум!”;

• работать в Интернете, осуществляя поиск информации, и др.;

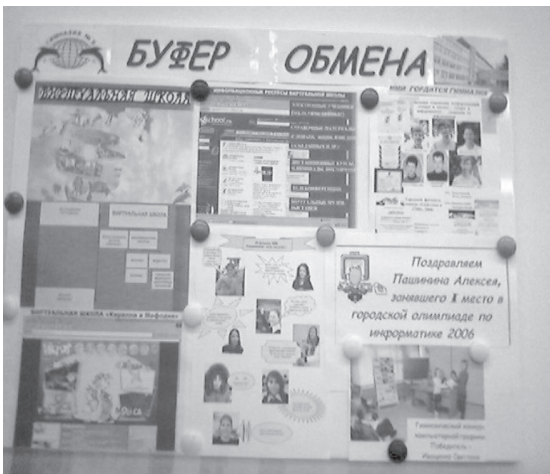
- готовиться к олимпиадам;
- обновлять стенд, выпускать тематические газеты;
- просто играть!

Всем этим ребята занимаются во внеурочное время, на кружке. Кабинет действительно становится вторым домом, местом обмена личным опытом; местом, где рождаются новые идеи, делаются интересные открытия и сообщения. Творческий процесс так увлекает, что мы с ребятами часто задерживаемся допоздна.

Мультимедийные проекты гимназистов ежегодно представляются на городской научно-практической конференции “Первые шаги в науку” и занимают призовые места.

На кабинетном стенде “Буфер обмена” размещается текущая, сменная информация:

- о победах на конкурсах и олимпиадах;
- поздравления с праздниками;
- новости компьютерного мира;
- об информатизации гимназии;
- программная наполняемость кабинета;
- содержание школьной медиатеки;
- темы проектов;
- гимназическая периодическая газета “Школьный бум!”;
- лучшие проекты;
- лучшая работа компьютерной графики;
- тематические газеты
- и др.



В кабинете также имеются:

- плакат по технике безопасности (он напечатан в газете “Информатика” № 2/2006);
- полки, на которых демонстрируются устройства ПК. Ребята могут приносить интересные находки. Так, имеются счеты, которые становятся редкостью, компьютер из теста, который для кабинета “слепила” ученица, другие поделки.

Один такой в России праздник, и, слава Богу, что один...

Николай Зиновьев



На фото слева направо: Роберт Гусовский, Татьяна Желудова, Анна Золотова, Александра Бельчицкая, Елена Морозова, Елена Кошкин, Мария Семиданова.

Елена Кошкин

Мои представления об идеальном кабинете

Е.Н. Смирнова,
г. Брянск

Мне хочется рассказать о кабинете информатики, в котором я работаю и которым заведу. За 16 проведенных вместе лет облик кабинета не раз менялся, как менялись и мои представления об идеальном предметном кабинете. Сегодня МОУ “Средняя общеобразовательная школа № 24 г. Брянска”, в которой я работаю, имеет два кабинета информатики и кабинет мультимедийных технологий. Наши кабинеты очень загружены: в первую смену уроки, во вторую — занятия кружков, элективных курсов, работают клубы. Поэтому важно организовать кабинет так, чтобы в нем было удобно и приятно находиться.

В моем кабинете можно выделить две зоны, мы с ребятами условно называем их “рабочей” и “компьютерной”. В “рабочей” зоне расположены одноместные письменные столы, обращенные к доске. В “компьютерной”, размещенной вдоль стен, находятся компьютерные рабочие места учащихся. Каждый вид работы ребята выполняют в определенном для этого месте, что дисциплинирует и делает любое занятие более целенаправленным. Учитель тоже имеет два рабочих места — компьютерное и письменное, оба находятся напротив ученических.

Я думаю, что в таком сугубо техническом кабинете, каковым является кабинет информатики, для создания уюта и атмосферы психологического комфорта необходимы зеленые уголки, поэтому я с удовольствием развожу цветы. Они у меня постоянно меняются.

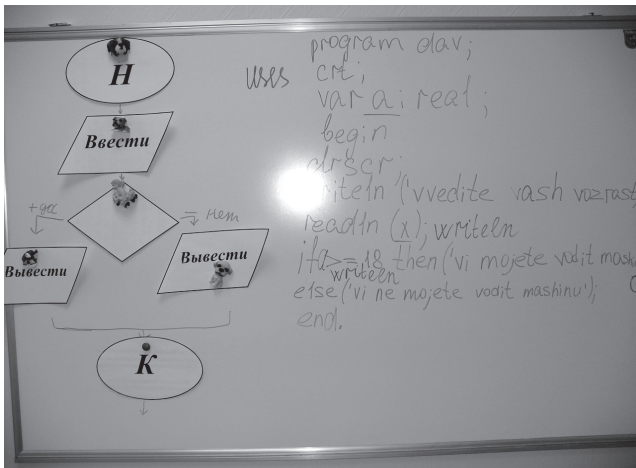
Отношение к плакатам и стендам на стенах у меня постоянно менялось. Иногда хотелось расписать все стены формулами из области информатики, был период, когда на стенах располагались творческие работы

учеников — рисунки, вышивки. Сейчас на стене висит только одна пробковая доска, на которой оперативно меняются необходимая для работы информация, материал для зачетов и занимательные материалы.



Конечно, может возникнуть вопрос о методических копилках. Если стены пустые — как и чем пользуется учитель для обеспечения наглядности объясняемого материала?! Пользуюсь как традиционными средствами — доской и мелом (только вместо мела — маркер), так и современными технологиями — мультимедийным проектором. Так же часто ребята работают с мультимедийными презентациями, компьютерными тестами и другим компьютерным раздаточным материалом в индивидуальном порядке. Для объяснения практически каждой темы я использую презентацию, разработанную мной или моими учениками.

Отдельно хочется рассказать о доске. В моем кабинете на передней стене висят две магнитные доски, на которых можно писать маркером. Это очень удобно. Часто одну из досок я использую в качестве экрана для демонстрации электронных документов и презентаций.



Очень большую помощь в методическом обеспечении качественной работы учителя оказывают мне ученики. Их рефераты — моя особая гордость. Ре-

ферат по информатике обязательно содержит теоретическую и практическую части. Уже много лет я накапливаю это интеллектуальное богатство и использую его на занятиях, не забывая новым поколениям ребят сообщить об авторстве. Это вдохновляет на выполнение аналогичной работы многих учащихся. Также я регулярно собираю тематические папки по материалам периодической печати. К этим папкам имею доступ не только я, но и мои ученики.

Конечно, изменения в интерьере и обеспечении кабинета происходят постоянно. Появляются новые возможности, приходят новые ученики, возникают новые идеи. Мне кажется, что в любом случае главное — любовь учителя к своему предмету и неиссякающий интерес к нему. Тогда и ученикам будет интересно, и кабинет станет функциональным и удобным.

Секреты комфортного кабинета

В.В. Пичугин,

р.п. Пинеровка, Саратовская обл.

Учебный кабинет должен быть комфортен и ученику, и учителю. У каждого заведующего кабинетом есть свои секреты, обеспечивающие этот комфорт на уроке и после. Удивительно, что оказалось трудным (по крайней мере мне) самому разложить на слагаемые то единое целое, что называют *кабинетом информатики*. Это и удобный всем режим работы, и доброжелательная атмосфера взаимопомощи, и элементарный санитарный порядок, и, конечно, всякие мелочи (мелочи ли?) вроде стендов, шкафов, полок, плакатов, папок...

Стенды и шкафы в моем кабинете вроде бы как у многих, но кое о чем хочется поведать.

“Умная мысль”. Если над доской висит хорошее высказывание — это полезно. Особенно, если есть много поводов сослаться на авторитет автора. Эстетично, легко и быстро можно изготовить такое пособие из потолочных плиток (голь на выдумки хитра), наклеив на них аппликацию из напечатанных на принтере букв.

“Чудные счеты”. Много раз сослужили добрую службу на уроках по двоичной системе счисления рукодельные (из “киндер-сюрпризов”) двоичные счеты.

“Полезная полочка”. В обычном книжном шкафу нашлось место для всяких демонстрационных мелочей: от арифмометра до микропроцессора, от перфокарты до современных носителей информации (в том числе и в разобранном виде), от полужабытого программируемого (пока работоспособного) микрокалькулятора до “внутренностей” современного ПК... Главное, чтобы это не пылилось в шкафу, а использовалось на уроке, на кружке, даже просто в беседе по случаю.



“Информационный стол”. Стол для самоподготовки оказался удачным дополнением к обычным рабочим местам (особенно после уроков). Под стеклом на столешнице, в прозрачных карманчиках из канцелярских файлов, в папках-скоросшивателях — полезная дополнительная информация для школьников и учителей.

“Карандаши-ручки”. Можно избежать многих проблем и проблемок на уроках и других занятиях, если в кабинете в достатке запасные карандаши-ручки, листочки-скрепочки.

Конечно, хорошо, когда в кабинете тепло, уютно, светло.. Но, этого мало. В кабинете должна быть особая аура, генерирует которую учитель, а подпитывают ученики.

Наш кабинет — это наша гордость!

И.С. Исакова, Л.Н. Крылова,
п. Лимбяха, г. Новый Уренгой

Кабинет информатики в нашей школе один. В нем проходят уроки 9–11-х классов по информатике, 10–11-х классов по технологии, а также кружки, в том числе и для малышей. Помимо всего этого, на базе кабинета проводятся педагогические советы с использованием мультимедиа, всевозможные конференции и совещания. Порой бывает так,

что здесь мы проводим времени даже больше, чем дома.

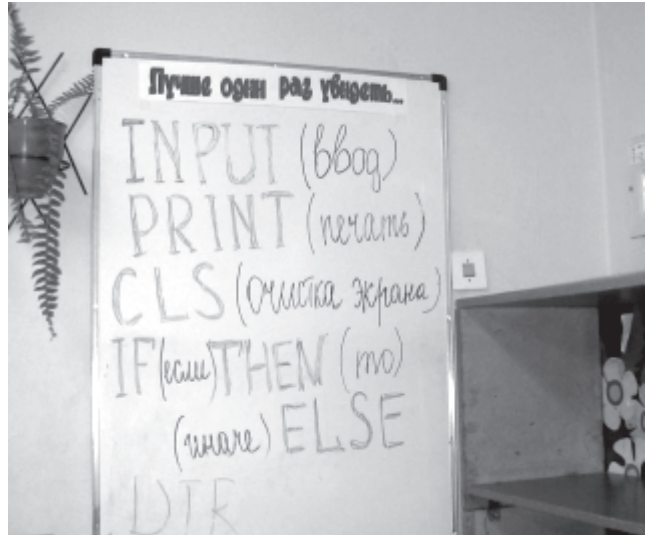
Кабинет информатики открылся в 1997 году, но и школа у нас относительно молодая: ей 18 лет.

Мы не будем здесь описывать, каким должен быть кабинет информатики. На эту тему достаточно много литературы. **Наш кабинет — это наша гордость.** Поэтому просто приглашаем на экскурсию.

При входе в кабинет располагается стенд, на котором размещается всевозможная информация: новости, конкурсы, объявления. Очень хочется, чтобы новую информацию получали не только те ребята, которые посещают данный кабинет, но и другие ученики школы.



Теперь зайдём и посмотрим, какой он у нас просторный, светлый и уютный.



При входе в кабинет (справа) располагается шкаф для сумок, рядом с ним маленькая доска под названием “Лучше один раз увидеть”. На ней мы записываем новые слова, чтобы они постоянно были у ребят перед глазами.

Остановимся подробнее на стендах.

1. “Выпускнику”. На этом стенде располагается вся информация по ЕГЭ относительно нашего предмета.

2. “Жизнь замечательных людей”. На этом стенде вывешиваются портреты и интересные факты из жизни тех людей, которые внесли значительный

вклад в науку при изучении темы, пройденной на уроке.

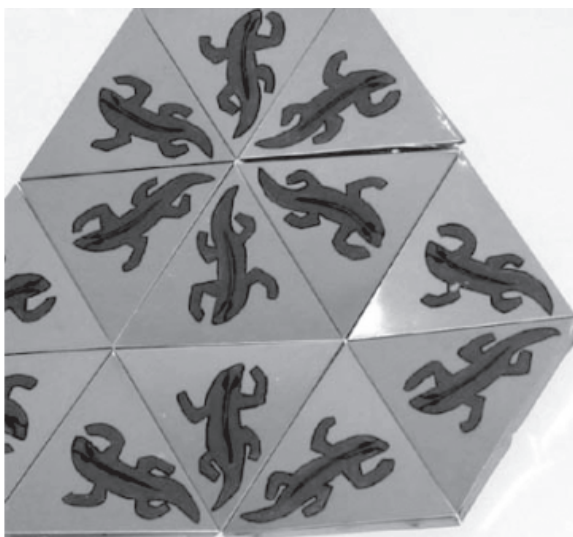
3. Стенд, на котором записаны правила правильной посадки за компьютером.

4. Далее следуют два стенда по технике безопасности, упражнения для глаз, рук и правила оказания медицинской помощи.

5. Еще у нас есть стена, которая называется “Награда наша героя”. Здесь собраны все грамоты (и наши, и учеников). Жалко только, что такая идея не пришла нам в голову сразу: копии многих грамот мы уже не можем собрать, так как ребята закончили школу.

Здесь же находятся две мои работы (И.С. Исаковой).

Первая — моя дипломная работа по математике на тему “Симметрия”. Вроде бы ничего общего с кабинетом, но очень интересно глядеть на этих милых ящерок. По крайней мере ребятам очень нравится.



Вторая работа — написанная маслом картина. Это первая и последняя (пока) такая большая работа. Даже не знаю, что на меня нашло, но очень захотелось нарисовать картину и повесить ее в кабинете.



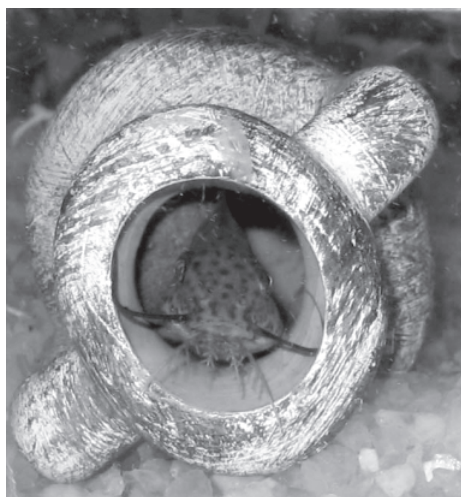
Доски, экраны, таблицы — это все типичное и про них не хочется рассказывать. Тем более что учителю информатики с его возможностями по силе создать

любой наглядный материал. И за многие годы работы его собралось много, он постоянно обновляется.

Еще у нас находится небольшая библиотека (конечно, в основном все из личного фонда), которой могут пользоваться все желающие. Помимо небольшой книжной библиотеки, мы создали библиотеку, состоящую из дисков различной тематики, но ими можно пользоваться только в классе. Для этого ребятам отведено специальное время работы в кабинете.

Также есть небольшой музей вычислительной техники.

Еще хотелось бы не забыть про главного жителя: для него наш кабинет является первым и единственным домом. Это сомик Фрося.



Вот и закончилась наша экскурсия по кабинету. Может, вам что-то и не понравилось, но это наша жизнь. Я люблю приходить рано утром: полюю цветы (их в кабинете очень много), приготовлю доску, раздаточный материал, посижу и полюбуюсь. Я никогда не устаю смотреть на свой любимый кабинет.

Стараюсь, чтобы кабинет не был скучным и “холодным”

Ж.В. Можарова,
г. Алапаевск, Свердловская обл.

В своем кабинете информатики я провожу очень много времени и могу сказать, что кабинет — мой второй дом.

Конечно, рядом со мной всегда ученики, которые любят предмет, который я преподаю. Поэтому я стараюсь, чтобы мой кабинет не был скучным и “холодным”. Хочется, чтобы он приносил ребятам не



только новые дополнительные знания, являлся почвой для их творчества и интеллектуального развития, но и был местом, где можно просто отдохнуть от уроков.

Наш кабинет оборудован в 2003 году новой современной компьютерной техникой: 11 новейших компьютеров, принтер, сканер, видеопроектор. Этот замечательный подарок нашей поселковой школе сделал бывший ее выпускник, ныне губернатор Саратовской области Ипатов Павел Леонидович. На отдельной стене нашего кабинета находятся фотографии поездки на Балаковскую атомную электростанцию летом 2003 года, куда нас приглашал Павел Леонидович, в то время директор БАЭС.

Следующая стена представляет собой информационно-методический стенд, на котором можно увидеть, прочитать какую-либо учебную, дополнительную или внеурочную информацию.

На стенде представлены следующие рубрики.

“*Наши достижения*”. Здесь мы вывешиваем грамоты спортивных и интеллектуальных состязаний, заметки о наших достижениях (из газет города Алапаевска). Здесь же вывешивается для проверки лист с домашним заданием по математике, так как я являюсь еще и учителем математики.



Техника безопасности и правила работы на компьютере. “Правила техники безопасности”, “Как следует сидеть за компьютером”, “Упражнения для развития гибкости пальцев рук”, “Основные правила набора текста”, “Зоны ответственности пальцев”, “Упражнения для снятия напряжения и усталости при работе на компьютере”, “Как правильно нажимать на клавиши”. Периодически материал обновляется и заменяется.

“*А вы как думали?*” В эту рубрику дети приносят вырезки из газет на компьютерную тему, журналы с интересными и полезными заметками, которые с удовольствием читают другие ребята. Здесь же ежемесячно вывешиваются материалы “Знаменательные даты месяца”.

“*Узнай побольше!*” В этой рубрике представляются материалы для углубленного изучения информатики и информационных технологий. Материал постоянно обновляется, дети используют его на дополнительных занятиях, на кружке или берут для изучения домой. Здесь же можно прочитать интересные факты из истории информатики, истории компьютера; о людях, которые внесли большой вклад в развитие компьютерной техники, и т.д.

Конкурс “*Инфознайка-2006*”. В данной рубрике ребятам предложены для решения “Гипервикторина” и “Гигатест”. Каждую неделю вывешивается по несколько вопросов. Дети приносят ответы на эти вопросы. Через неделю вывешиваются ответы и следующие вопросы. В конце года подводятся итоги и выбирается “инфознайка года”.

Результаты работы кружка “Школа компьютерных услуг (ШКУ)”. В школе работает кружок по выпуску бюллетеней, публикаций, поздравительных открыток, буклетов и т.д. А символ кружка “показывает” их остальным учащимся школы.

“*Ломаем голову*”. Ребята составляют кроссворды, ребусы.

“*Готовимся к экзамену*”. Вывешиваются материалы по подготовке к экзаменам по информатике и математике (пробные варианты, инструкции по выполнению заданий, рекомендации по подготовке и т.д.).

В верхней части информационно-методического стенда можно увидеть оформленный первыми моими



ми выпускниками стенд “Общая структура персонального компьютера”.

Могу сказать, что данный информационно-методический стенд необходим детям, они с интересом участвуют в конкурсах, читают дополнительную литературу, сами приносят интересные материалы.

Кроме этого, в кабинете имеются таблицы по разным темам курса “Информатика и ИКТ”.

По всем темам курса мною разработаны конспекты, карточки-проводники, дифференцированные задания, тесты и т.д. Данный материал находится в методическом шкафу. В основном при работе на компьютере учащиеся занимаются с индивидуальными папками по теме урока. Если ребенок не успел выполнить задание на уроке, он приходит после урока, берет нужную папку и выполняет работу.

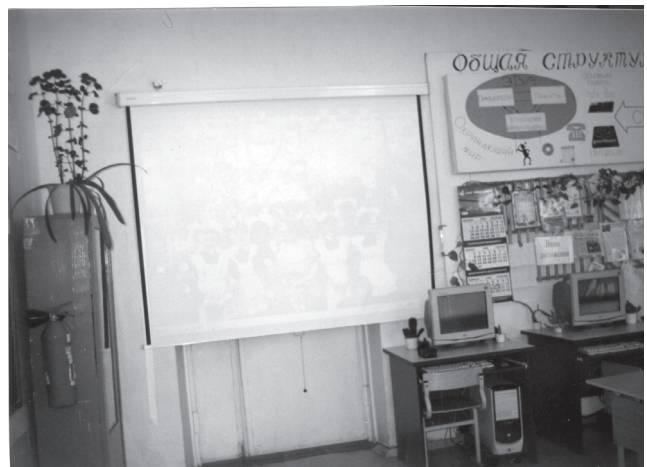
В другом шкафу находятся различные учебники и дополнительная литература.

На уроках и во внеурочной деятельности часто применяется презентационная техника.

В кабинете имеется коллекция различных программных продуктов (как по информатике и математике, так и по другим предметам), которую дети используют не только на уроках, но и в свободное от уроков время.

В кабинете огромное множество кактусов и цветов.

Могу сказать, что мой кабинет — это моя радость и гордость, потому что все здесь, начиная с ремонта и заканчивая дидактическими материалами и стендами, сделано своими руками, с помощью детей и их родителей. Огромное им за это спасибо!



ПОДПИСКА 2006. ВТОРОЕ ПОЛУГОДИЕ**Газета «ИНФОРМАТИКА»****Газета «Информатика» + серия «Информатика» Библиотечки «ПС»****32291** — подписной индекс для индивидуальных подписчиков**32591** — подписной индекс для организаций**При подписке на 6 месяцев — подарок:**

три выпуска Библиотечки «Первого сентября» серии «Информатика».

Подписка на любой почте России по каталогам «Газеты. Журналы» агентства «Роспечать» или Объединенному каталогу «Пресса России» (см. третью страницу обложки этого номера газеты).

Все газеты + все серии Библиотечки «ПС»**Газета «Информатика» вместе с Библиотечкой входит в «Первое сентября. Полный комплект изданий».**

Подписка на полный комплект изданий «Первого сентября» — существенная экономия для школы, которая приняла решение подписаться не на отдельные газеты нашего Издательского дома, а сразу на все. В этом случае также работает правило ШЕСТИ МЕСЯЦЕВ: при подписке на шесть месяцев второго полугодия ко всем газетам бесплатно прилагается Библиотечка «Первого сентября» соответствующей серии.

32745 — подписной индекс для организаций

Подписка на любой почте России по каталогам «Газеты. Журналы» агентства «Роспечать» или Объединенному каталогу «Пресса России» (см. третью страницу обложки этого номера газеты).

Серия «Информатика» Библиотечки «ПС»

Есть возможность подписаться только на Библиотечку «Первого сентября». Такая подписка — хороший подарок коллеге или недорогая возможность получить несколько брошюр, если в вашей школе выписывается одна газета «Информатика» для нескольких учителей.

19179 — подписной индекс по каталогу «Газеты. Журналы» агентства «Роспечать»

Аннотации и описания брошюр прошлых выпусков Библиотечки «Первое сентября» всех серий теперь можно найти на сайте интернет-магазина «Первое сентября».

**ИНТЕРНЕТ-МАГАЗИН**
«Первое сентября»<http://bookshop.1september.ru>
Тел. (495) 249-47-58

НАЧАЛКА

газета-клуб для всех,
кто учит информатике
маленьких детей



№ 8 (16–30 апреля)

Потрогать информатику руками. Часть 2

В.А. Козлова,
г. Пермь

В первой части статьи (см. “Началку” № 6/2006) мы представили наш подход к изучению информатики в начальной школе. Напомним, что в пермской компьютерной школе ЦИТО широко используются Лого-среды — ПервоЛого, ЛогоМиры и RoboLab. Принципиально важным нам представляется то, что на наших занятиях в “Школе компьютерных волшебников” дети не только экспериментируют с Лого в чисто компьютерном (экранном) варианте, но и могут “потрогать информатику руками”, управляя специализированными учебными роботами. В № 6/2006 были приведены фрагменты учебного блока “Управление исполнителями” из тетради учащегося. В этом номере мы хотим представить учебный блок “Использование датчиков”, на котором также ярко видны принципиально важные аналогии между черепашками и роботами.

Учебный блок “Использование датчиков” из тетради учащегося

1. Есть ли у черепашек органы чувств?

Да, у черепашек ПервоЛого есть своеобразные органы чувств: **датчики**. Они позволяют определять:

- цвет фона под черепашкой,
- касание с другой черепашкой,
- сигналы от других черепашек, передаваемые через специальный светофор.

Программирование черепашек на работу с датчиками

Возьми в инструментах ключ и, щелкнув им на черепашке, открой ее. На экране появится окно программирования черепашки.

- Вкладка с мышкой служит для записи основного алгоритма действий черепашки.



Рис. 1. Вкладка с мышкой

- Вкладка с палитрой — для записи действий, которые черепашка должна выполнить, обнаружив под собой определенный цвет фона.

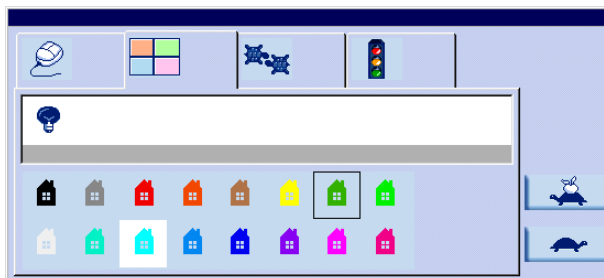


Рис. 2. Вкладка с палитрой

- Вкладка с черепашками — для определения действий при столкновении с другой черепашкой.

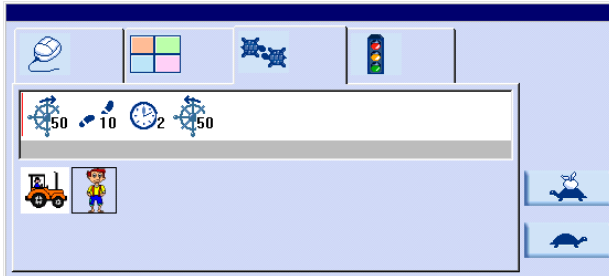


Рис. 3. Вкладка с черепашками

• Вкладка со светофором определяет действия черепашки, согласованные с другими черепашками. Эти действия определяются одним из 6 цветов светофора.

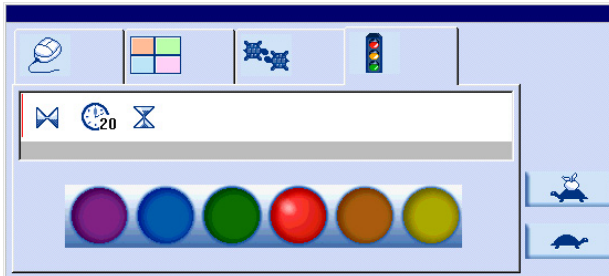


Рис. 4. Вкладка со светофорами

На любой вкладке программа действий черепашки записывается в поле ввода команд. Команды выбирать надо из меню команд.

Задание на компьютере

Наведи порядок на дороге по правилам дорожного движения

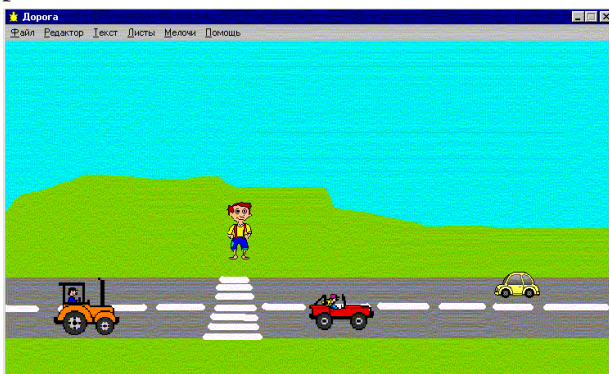


Рис. 5. Дорога

1. Открой, используя ярлык на Рабочем столе, программу под названием *Перволог*.
2. Открой из *Общей папки* альбом *Дорога.mw2*.
3. Запрограммируй каждую черепашку на движение вперед, но с разной скоростью.
4. Научи черепашку-мальчика распознавать цвета: на голубом небе — спрятаться, на зеленом лугу — показаться. Используй вкладку с палитрой.

5. Научи черепашки-машины разъезжаться на дороге, поворачивая в противоположные стороны при соприкосновении. Используй для этого вкладку с черепашками.

6. Научи черепашку — легковую машину при соприкосновении с черепашкой-мальчиком включать красный свет светофора, засыпать на 2 сек., просыпаться и включать зеленый сигнал светофора. Используй для этого вкладку с черепашками.

7. Научи черепашку-трактор реагировать на красный свет светофора, засыпая на 2 сек., чтобы мальчик успел перейти дорогу. Используй для этого вкладку со светофором.

8. Сохрани измененный проект в своей *Личной папке*.

2. Программирование работы Лего-робота с датчиком касания

Как ты помнишь, Лего-робот можно запрограммировать на выполнение каких-то действий в течение некоторого промежутка времени — для этого используют **таймер**. Кроме таймера, в качестве ограничителя может выступать информация от **датчика касания**. При этом робота можно научить реагировать или на **нажатие**, или на **отпускание датчика**.

Программируя использование датчика касания, необходимо указывать **номер порта** — разъема на RCX-блоке, к которому подключен датчик. Чтобы его указать правильно, надо сначала хорошо рассмотреть робота — к какому порту идет провод от датчика касания? А затем щелкнуть в программе по имеющемуся номеру порта и выбрать из появившегося меню нужный номер.

Включение в программу датчика касания в режимах Управление 2, Управление 3

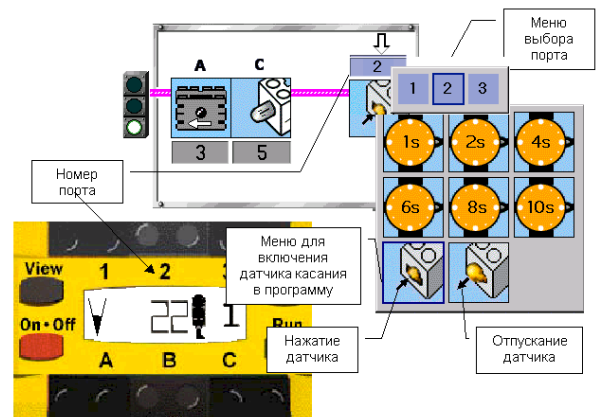


Рис. 6. Включение в программу датчика касания в режимах Управление 2 и 3

Включение в программу датчика касания в режиме Управление 4

Выбрав датчик касания, щелкни по стрелочке, чтобы указать, будет он реагировать на нажатие или на отпускание.

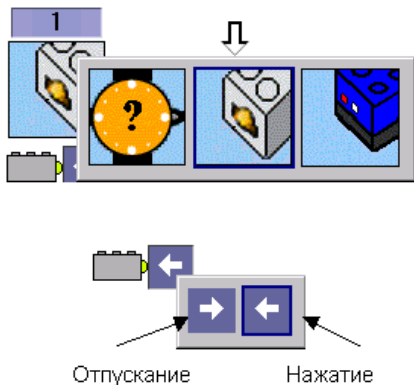


Рис. 7. Включение в программу датчика касания в режимах Управление 2, 3

Задание на компьютере

Составь программы для робота. Проверь работу каждой программы, передав ее роботу.

Задания для режима Управление 2

1. Робот едет вперед, пока не заденет препятствие.
2. Робот вертится, пока не заденет препятствие.

Задания для режима Управление 3

1. Проверь работу следующей программы:

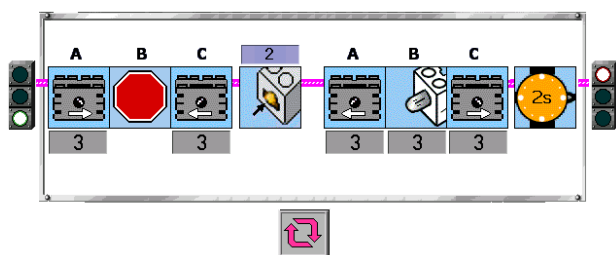


Рис. 8. Программа для робота

Измени программу так, чтобы робот сначала ехал вперед, пока не заденет препятствие. Затем ехал бы с включенной лампочкой назад 2 с. И повторял это, пока робота не выключат.

2. Следующая программа предполагает такое поведение робота:

Робот едет вперед с включенной лампочкой, пока не заденет препятствие. Затем гасит лампочку, разворачивается и едет в обратном направлении опять с включенной лампочкой.

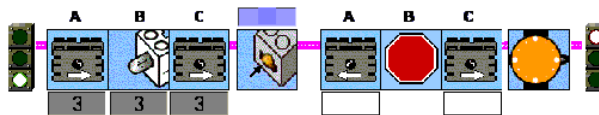


Рис. 9. Программа для робота

Укажи порт для датчика касания, подбери мощность моторов и время, которые должны быть установлены, чтобы робот разворачивался после встречи с препятствием на 180 градусов.

Задание для режима Управление 4 (повышенной сложности)

Придумай программу из нескольких шагов, по которой робот-разведчик сумеет объехать препятствие любого размера без помощи человека.

3. Робот проверяет освещенность

Время работы моторов Лего-робота можно задать не только таймером, но и информацией от датчиков касания и освещенности. При включении в программу датчиков необходимо указывать дополнительную информацию для Лего-робота:

- Для датчика касания — номер порта, к которому подключен датчик, и состояние датчика: нажатие или освобождение.
- Для датчика освещенности — номер порта, к которому подключен датчик, и уровень освещенности.

Появившуюся на экране метку-галочку надо повторными нажатиями кнопки View поставить напротив номера порта, к которому подключен датчик освещенности.

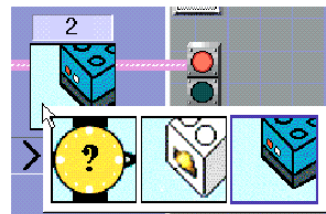


Рис. 10. Подключение датчика к выбранному порту

Условия, связанные с уровнем освещенности, задаются **проверками**, которые записываются с помощью знаков отношения “<”, “>”.

Прочитать условие, изображенное рисунком, можно так: “Жди, пока уровень освещенности станет больше 55”.

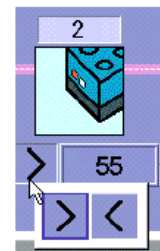


Рис. 11. Условие на уровень освещенности

Уровень освещенности отображается на дисплее блока RCX при нажатой черной кнопке **View**.

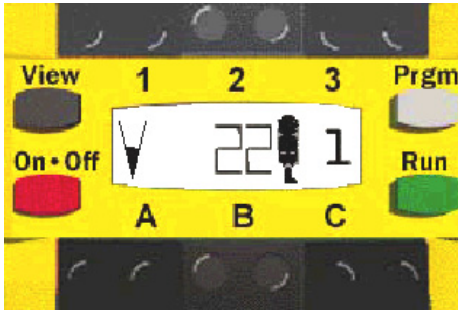


Рис. 12. Уровень освещенность на дисплее RCX

Упражнение

Что будет делать робот, выполняя следующую программу из двух шагов:

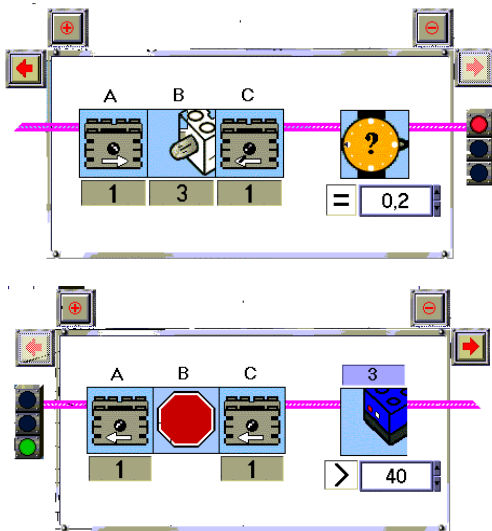


Рис. 13. Программа для робота

1. Сначала робот будет _____ в ожидании, пока _____ станет _____
2. Потом робот будет _____ в ожидании, пока _____ станет _____

Задания на компьютере

Запусти из *Главного меню* программу **RoboLab** и составь программы для робота. Проверь работу каждой программы, передав ее роботу.

Чтобы по твоим программам робот работал правильно, проверь и запиши уровни освещенности на разных цветных полях полигона:

Черный _____ Белый _____
Зеленый _____

Задания для режима Управление 4

1. Проверь работу программы, приведенной выше, запустив робота по черной полосе полигона. Если потребуется, измени номер порта, к которому подключен датчик освещенности, а также значение уровня освещенности.

2. Измени программу так, чтобы робот разворачивался против часовой стрелки в поисках черной полосы.

3. Напиши программу, выполняя которую робот будет на темных полях ехать по прямой с минимальной скоростью, а на белых полях — ускоряться.

4*. Напиши программу, выполняя которую робот будет по-разному вести себя на участках разного цвета:

- на черном — ехать назад,
- на белом — ехать вперед,
- на зеленом — кружиться.

5*. Запрограммируй “танец” робота из 4–6 шагов, выполняя которые он будет проверять состояние датчиков касания и освещенности.

Характеристики УМК “Школа компьютерных волшебников”

Сегодня “Школа компьютерных волшебников” представляет собой полноценный учебно-методический комплекс, включающий в себя:

- Рабочие тетради для учащихся (по одной на каждый семестр), содержащие конспект материала урока, упражнения, задания для выполнения на компьютере и домашние задания.
- Электронные презентации по теоретическим темам курса и презентации-тренажеры.
- Копилки файлов для выполнения упражнений на компьютере.
- CD со свободно распространяемыми программами, а также файлами для самостоятельной работы учащихся.
- Методические рекомендации по проведению занятий.
- Электронные тесты — тренировочные и контрольные.

Обучение по курсу “Школа компьютерных волшебников” проходит в трех полугодовых циклах, состоящих из:

- 12–14 уроков,
- самостоятельных работ, рассчитанных еще на 12 часов,

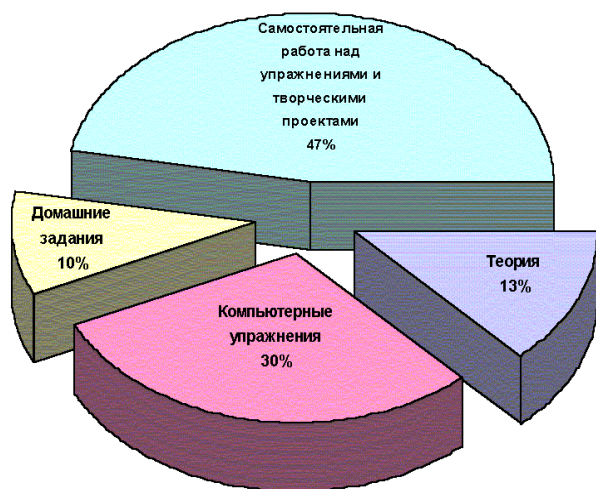


Рис. 14. Распределение учебного материала

- открытого урока в конце цикла, на котором дети представляют собственные проекты.

Краткое содержание курса

Цикл 1

Знакомство с устройством компьютера и Лего-робота, их возможностями по работе с информацией разных видов.

Освоение компьютерных технологий обработки информации разных видов на примере стандартных программ Windows.

Цикл 2

Использование программ-редакторов (текстового, графического, музыкального) для работы с информацией разных видов.

Понятия исполнителя, его системы команд, алгоритма и программы.

Составление линейных и циклических алгоритмов для компьютерных исполнителей и Лего-роботов.

Цикл 3

Использование разных компьютерных инструментов в комплексе для выполнения мультимедийных творческих работ в Лего-средах.

Планирование сложного поведения технических устройств, требующего анализа условий и выбора из нескольких разных вариантов действий.

Составление разветвляющихся алгоритмов для компьютерных исполнителей и Лего-роботов.

Основные результаты обучения

В области *компьютерных технологий* у детей формируются следующие умения:

- умение запускать, самостоятельно знакомиться и использовать в комплексе разные компьютерные программы и инструменты для выполнения творческих работ (компьютерных мультфильмов, электронных альбомов, сочинений, обучающих средств);
- умение сохранять результаты своей работы на дискете или жестком диске компьютера и демонстрировать их.

В области *алгоритмики и программирования* формируются следующие умения:

- умение планировать сложное поведение компьютерных исполнителей и Лего-роботов, требующее анализа условий и выбора из разных вариантов действий;
- умение составлять линейные, циклические, разветвляющиеся алгоритмы и программировать их для компьютерных исполнителей и Лего-роботов.

Кроме того, дети учатся грамотно использовать термины из области информатики и вычислительной техники для пояснения выполненной на компьютере работы.

Кроме того, дети учатся грамотно использовать термины из области информатики и вычислительной техники для пояснения выполненной на компьютере работы.

Координаты для связи с разработчиками

Адрес: г. Пермь, Комсомольский пр-т, 91. Компьютерная школа ЦИТО. Тел.: (342) 2-410-510, 244-17-83.

E-mail: slava@perm.raid.ru — Аспидов Вячеслав Владимирович, kurs14@yandex.ru — Козлова Валентина Алексеевна.

Использованные источники информации

1. ИНТ. Программные продукты Лого (<http://www.int-edu.ru/logo/products.html>).

2. Информатика: основы компьютерной грамоты. Начальный курс / Под ред. Н.В. Макаровой. СПб.: Питер, 2000.

3. Макарова Н.В. Информатика, 5–6-е классы. Начальный курс (2-е издание). СПб.: Питер, 2003.

4. ИНТ. Наборы LEGO DACTA для образовательной области “Технология” (<http://www.int-edu.ru/lego/catalog/techno.htm>).

5. Белиовская А.Г. Моделирование механо-микро-процессорных роботов с искусственным интеллектом на примере самоуправляемых роботов в среде “ROBOLAB” (<http://www.9151394.ru/projects/lego/legob/beliovskaya/>).

6. i-школа. ЛЕГО (<http://www.home-edu.ru/&r=class&p=robolab>).

7. Московская национальная еврейская школа № 1311, факультатив по робототехнике (<http://sch1311.msk.ort.ru/our/technology/robolab>).



роботландский университет

[начало информатика](#) [конструирование](#) [программирование](#) [интернет](#) [английский](#)

[учителя](#)

[школьники](#)

[учебники](#)

[программы](#)

[лаборатории](#)

[теория](#)

[практика](#)

[турниры](#)

[олимпиады](#)

[общение](#)

[конференции](#)

[творчество](#)

[помощь](#)

Роботландский университет — это сетевая школа при негосударственном образовательном учреждении дополнительного образования “Роботландия+” (г. Переславль-Залесский).

Руководитель университета — Дуванов Александр Александрович.
Сайт университета расположен по адресу: www.botik.ru/~robot/ru.

В университете обучаются, как правило, “коллективные ученики”.

“Коллективный ученик” — это группа детей, работающая под руководством одного или нескольких наставников (школьных учителей).

В конце годичного курса при успешном обучении школьный учитель получает удостоверение от негосударственного образовательного учреждения “Роботландия+” и удостоверение государственного образца от Российской академии повышения квалификации и переподготовки работников образования.

Школьники получают удостоверение Роботландского университета.

Популярные курсы университета

- “Зимние вечера” (куратор: Первин Юрий Абрамович).
Курс для младших школьников (3–4-е классы). Алгоритмизация, компьютерное письмо, рисование, электронная почта.
- “Конструирование” (куратор: Первин Юрий Абрамович).
Курс для детей 4–7-х классов. Создание компьютерных проектов на базе программ-конструкторов.
- “Азы информатики” (кураторы: Козлова Валентина Алексеевна, Кацай Ирина Ивановна).
Для школьников 5–8-х классов. Обучение построено на базе нового одноименного курса информатики.
- “Азы программирования”.
Для детей 6–8-х классов. Введение в программирование на базе исполнителей Кукарача и Корректор.
- “Web-конструирование” (куратор: Дуванов Александр Александрович).
Для старшеклассников. Создание классических HTML-документов. Основы web-дизайна и проектирования. Создание сайта, удобного для использования.

Зачем нужен Роботландский университет

Роботландский университет предлагает своим студентам то, что невозможно получить, замыкаясь в своей школе, работая даже по самым хорошим учебным пособиям, с самым продвинутым педагогическим программным обеспечением.

Дело в том, что программа Роботландского университета не сводится только к самому обучению, а предполагает создание реальных продуктов, сработанных по всем правилам изучаемых технологий. А это существенно повышает эффективность курсов, демонстрируя плоды обучения в реальных прикладных воплощениях.

Конечно, реальные воплощения возможны и в обычном учебном процессе под руководством школьного учителя.

Это так. Но продукт, сделанный своими руками, кажется прекрасным, и в нем трудно заметить погрешности! Внутри школы это трудно сделать даже учителю.

А ведь реальные продукты перед внедрением тестируются специалистами, которые сами не являются разработчиками проверяемого изделия.

Роботландский университет предлагает испытательный полигон в виде перекрестных проверок продукта силами других команд. Дирижером проверок является куратор курса, который, конечно, и сам проверяет, и компетентно комментирует все работы без исключения.

На курсах университета запланированы большие отчетные конкурсы и непрерывная цепочка маленьких, еженедельных.

Кроме того, постоянно работает общая педагогическая конференция. Учителя получают возможность обсудить проблемы, связанные с преподаванием, получить консультации от коллег, куратора и автора курса.

Приглашаем на курсы Роботландского университета!

Заявки на обучение принимаются в сентябре. Заявку можно подать при помощи формы, которая расположена на роботландском сайте в разделе www.botik.ru/~robot/ru/set.htm.

1996/1997

↑
десятый
учебный
год
в наших
сетях
↓

2005/2006



В мир информатики

73 (16—30 апреля)

Газета для любознательных учеников
и их талантливых учителей

О признаках делимости

Как известно, существуют простые признаки, позволяющие определить, что то или иное число делится, например, на 3, на 5, на 9 и т.п. Напомним эти признаки.

1. **Признак делимости на 3:** число делится на 3, если сумма его цифр делится на 3. Например, число 257802 (сумма цифр $2 + 5 + 7 + 8 + 0 + 2 = 24$) делится на три, а число 125831 (сумма цифр $1 + 2 + 5 + 8 + 3 + 1 = 20$) на три не делится.

2. **Признак делимости на 5:** число делится на 5, если его последняя цифра есть 5 или 0 (т.е. если на 5 делится число единиц его последнего разряда).

3. **Признак делимости на 2** аналогичен предыдущему: число делится на 2, если на 2 делится число единиц его последнего разряда.

4. **Признак делимости на 9** аналогичен признаку делимости на 3: число делится на 9, если сумма составляющих его цифр делится на 9.

Доказательство справедливости этих признаков не представляет труда. Рассмотрим, например, признак делимости на 3. Он основан на том, что единица каждого из разрядов десятичной системы (т.е. числа 1, 10, 100, 1000 и т.д.) при делении на 3 дает остаток 1. Поэтому всякое число

$$(a_n a_{n-1} a_{n-2} \dots a_1 a_0)_{10},$$

т.е. число

$$a_n \cdot 10^n + a_{n-1} \cdot 10^{n-1} + a_{n-2} \cdot 10^{n-2} + \dots + a_1 \cdot 10 + a_0$$

можно записать в виде

$$(a_n + a_{n-1} + a_{n-2} + \dots + a_1 + a_0) + B,$$

где число B делится на 3 без остатка. Отсюда следует, что

$$a_n \cdot 10^n + a_{n-1} \cdot 10^{n-1} + a_{n-2} \cdot 10^{n-2} + \dots + a_1 \cdot 10 + a_0$$

делится на 3 в том и только в том случае, если на 3 делится число $a_n + a_{n-1} + a_{n-2} + \dots + a_1 + a_0$, т.е. сумма цифр исходного числа.

Признак делимости на 5 вытекает из того, что число 10 — основание системы счисле-

ния — делится на 5, поэтому все разряды, кроме разряда единиц, при делении на 5 обязательно дают в остатке ноль. На том же самом основан и признак делимости на 2: число четное, если оно кончается четной цифрой.

Признак делимости на 9, как и признак делимости на 3, вытекает из того, что каждое число вида 10^k при делении на 9 дает в остатке 1.

Из сказанного ясно, что все эти признаки связаны с представлением чисел именно в десятичной системе и что они, вообще говоря, неприменимы, если пользоваться системой счисления с каким-либо другим основанием, отличным от 10. Например, число 86 в восьмеричной системе записывается в виде 126_8 , (так как $86 = 8^2 + 2 \cdot 8 + 6$). Сумма цифр равна 9, но 86 не делится ни на 9, ни на 3.

Однако для каждой позиционной системы счисления можно сформулировать свои признаки делимости на то или иное число.

Рассмотрим несколько примеров.

Будем писать числа в двенадцатеричной системе и сформулируем для такой записи признак делимости на 6. Так как число 12 — основание системы счисления — делится на 6, то число, записанное в двенадцатеричной системе, делится на 6 в том и только в том случае, если на 6 делится его последняя цифра (здесь то же самое положение, что и с делимостью на 5 или на 2 в десятичной системе).

Так как числа 2, 3 и 4 тоже служат делителями числа 12, то справедливы следующие признаки делимости: число, записанное в двенадцатеричной системе, делится на 2 (соответственно, на 3 и на 4), если его последняя цифра делится на 2 (соответственно, на 3 и на 4).

Задание для самостоятельной работы

Определить признаки делимости:

- 1) в двоичной системе — на 2 и на 4;
- 2) в троичной системе — на 2 и на 3;
- 3) в четверичной системе — на 2, на 3 и на 4;
- 4) в шестеричной системе — на 2, на 3, на 5 и на 6;
- 5) в восьмеричной системе — на 2, на 4, на 7 и на 8;
- 6) в двенадцатеричной — на 8, на 9 и на 11.

¹ Как известно, это есть так называемая “развернутая форма” заданного числа.

Основы программирования на Visual Basic

Продолжение. Начало см. "В мир информатики" № 69–72 ("Информатика" № 4–7/2006)

Н.М. Тимофеева,
г. Обнинск Калужской обл.

Пример программирования № 4 "Вычисление площади"

Постановка задачи

Разработайте программу для вычисления площади трапеции по известным значениям длин ее оснований и высоты.

План решения

На рис. 1 показан предлагаемый дизайн экрана:

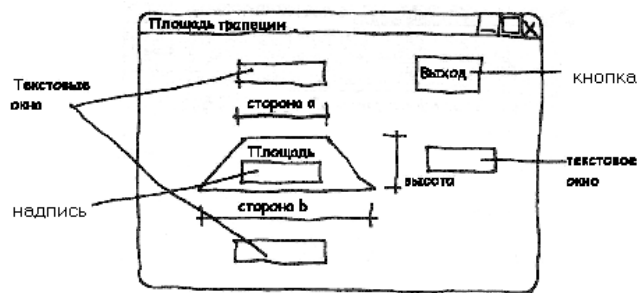
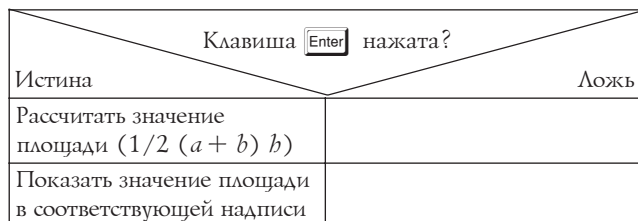


Рис. 1

Назначение каждого элемента управления:

- 1) текстовые окна — для ввода значений исходных параметров;
- 2) надписи — одна для показа вычисленной площади, остальные — для описания исходных параметров;
- 3) линии — для изображения трапеции и размерных линий;
- 4) кнопка — для завершения программы по щелчку.

Каждое текстовое окно будет иметь одинаковые подпрограммы, реагирующие на событие "Нажатие клавиши **Enter**". N–S-диаграмма для разработки подпрограмм:



Подпрограмма для кнопки будет аналогична разработанным во всех предыдущих проектах.

Создание интерфейса

- Используя элемент управления инструментом Line (Линия), начертите трапецию (вы вынуждены будете чертить каждую из четырех сторон отдельно). Добавьте линии для указания размеров. У линий есть свойство `BorderWidth` (ширина), которое позволяет чертить одни линии (например, размерные) тоньше, чем другие.

- Поместите на форму пять надписей, три текстовых окна и одну кнопку. Измените размеры и позицию надписей и текстовых окон, как показано на рис. 2 и 3.

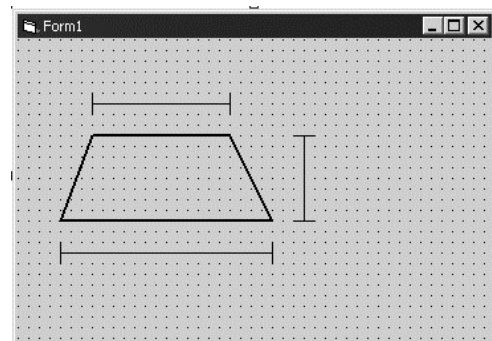


Рис. 2

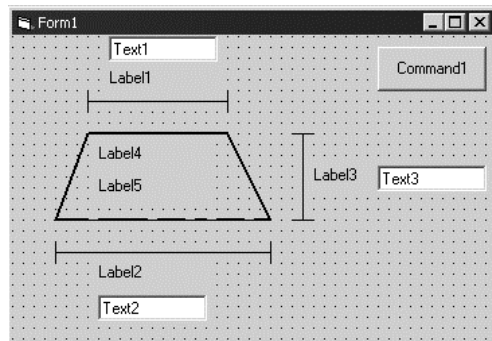


Рис. 3

Установка свойства

| | |
|--------------------------|------------------|
| Form1 | |
| <code>Caption</code> | Площадь трапеции |
| Label1 | |
| <code>Caption</code> | Сторона a |
| Label2 | |
| <code>Caption</code> | Сторона b |
| Label3 | |
| <code>Caption</code> | Высота |
| Label4 | |
| <code>Caption</code> | Площадь |
| Label5 | |
| <code>Name</code> | lblПлощадь |
| <code>BorderStyle</code> | 1 - Fixed Single |
| <code>Caption</code> | 0 |

Text1

| | |
|------|-------------|
| Name | txtСторонаА |
| Text | 0 |

Text2

| | |
|------|-------------|
| Name | txtСторонаВ |
| Text | 0 |

Text3

| | |
|------|-----------|
| Name | txtВысота |
| Text | 0 |

Command1

| | |
|---------|----------|
| Name | cmdВыход |
| Caption | Выход |

Написание кода

Событие по умолчанию для текстового окна — это Change (Изменение). А код, который будет написан для текстовых окон, будет реагировать на событие KeyPress (Нажатие на клавишу). Это событие выбирается из выпадающего списка, как это показано на рис. 4:

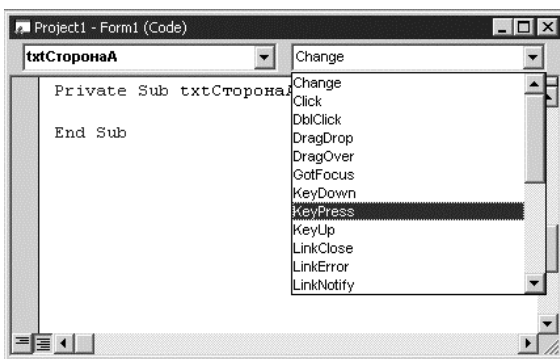


Рис. 4

Как указывалось выше, код, реагирующий на событие KeyPress (Нажатие на клавишу), будет одинаковым для всех трех текстовых окон. Когда нажата клавиша **Enter** в любом из текстовых окон, код должен обеспечивать вычисление площади и присваивание вычисленного значения свойству Caption надписи lblПлощадь (для этого случая была выбрана надпись, а не текстовое окно, так как указанное только что свойство надписи может быть изменено только с помощью кода).

Введите следующий код для текстового окна txtСторонаА:

```
Private Sub txtСторонаА_KeyPress(KeyAscii As Integer)
'Если нажата клавиша <Enter>,
If KeyAscii = 13 Then
'то вычисляется площадь
lblПлощадь.Caption = 1/2 *
(Val(txtСторонаА.Text) +
+ Val(txtСторонаВ.Text)) *
Val(txtВысота.Text)
End If
End Sub
```

Создайте аналогичный код для двух других текстовых окон (отличие будет состоять лишь в заголовках подпрограмм). При этом пользуйтесь операциями **Copy (Копировать)** и **Paste (Вставить)** из пункта меню **Edit (Правка)**, вместо того чтобы печатать весь текст заново.

Введите следующий код для кнопки:

```
Private Sub cmdВыход_Click()
End 'Завершить программу
End Sub
```

Сохранение проекта

Создайте папку ПРИМЕР4 и сохраните в ней форму и проект под именами *трапеция.frm* и *трапеция.vbp*.

Тестирование приложения

- Запустите программу.
- Обратите внимание, что площадь вычисляется

только тогда, когда вы нажимаете клавишу **Enter**.

• Решите задачу: *“Дорожным рабочим нужно сделать покрытие на участке дороги длиной 25 м, которая сужается с ширины 6 м до ширины 4 м. Какова общая площадь покрытия?”*

Взгляд назад

Проверьте, что программа выполняет все, что было задано при постановке задачи. Как вы думаете, пользователю понятно, как вводить начальные данные? Как бы вы могли сделать это более понятным? Продумайте еще раз каждый шаг.

Пояснения

1. Значение каждого свойства элемента управления имеет конкретный тип данных. В частности, свойство Text текстового окна — строковый (состоящий из символов) тип. Поэтому, когда это свойство используется в вычислениях, необходимо преобразовать значение в числовой тип данных. Вот почему в коде использовалась функция Val. Например: Val(txtСторонаА.Text).

2. При вводе данных в текстовое окно могла бы произойти ошибка (при вводе нечисловых значений). Функция Val частично обеспечивает защиту — любые нечисловые данные преобразуются этой функцией в ноль.

3. Как обычно, для расчета величин и присваивания их значению тех или иных свойств элементов управления используется оператор присваивания. Для значения свойства Caption надписи lblПлощадь расчет проводится по формуле:

$$A = 1/2 (a + b)h$$



В алгебре знак умножения можно опустить. В коде Visual Basic он обязательно должен присутствовать:

$$A = 1 / 2 * (a + b) * h,$$

или в нашем случае:

```
lblПлощадь.Caption = 1 / 2 *
(Val(txtСторонаА.Text) +
Val(txtСторонаВ.Text)) * Val(txtВысота.Text)
```

4. В заголовке процедур обработки события KeyPress (Нажатие на клавишу) можно увидеть параметр KeyAscii. Он объявляется как величина целого типа — Integer (то же самое, как вы понимаете целые в математике — целые положительные и отрицательные числа): (KeyAscii As Integer).

Каждое нажатие на клавишу клавиатуры ассоциируется с числовым кодом (так называемым “ASCII-кодом”). Клавиша  имеет код 13. Всякий раз, когда нажимается клавиша на клавиатуре, процедура обработки события KeyPress соответствующего текстового окна проверяет, не нажата ли клавиша . Как только это происходит — вычисляется площадь.

5. В этом примере программирования применяется так называемый “неполный условный оператор”. Его общий вид:

```
If <Условие> Then
<Действие (ия) >
End If
```

Он используется в программах в тех случаях, когда какое-то действие (какие-то действия) должно выполняться не всегда, а только при определенном условии. Обратите внимание, End If отмечает конец конструкции ветвления (неполного условного оператора) в коде.

Вопросы и задания для проверки знаний

1. Каждый символ на клавиатуре (и не только) имеет² ASCII-код (*American Standard Code for Information Interchange* — Американский стандартный код для обмена информацией). Буквы англий-

ского алфавита закодированы так: “А” — 65, “В” — 66, “С” — 67 и т.д., для строчных букв: “а” — 97, “b” — 98, “с” — 99 и т.д.

- назовите ASCII-код буквы “R”;
- назовите ASCII-код тринадцатой буквы английского алфавита (заглавной буквы);
- назовите ASCII-код буквы “y”.

2. Как вы думаете, уместно ли использовать тип данных Variant для свойства Caption (Надпись)?

3. Параметр KeyAscii (в рассмотренном примере программирования) имеет тип данных Integer. Что это значит?

4. Какой тип данных у свойства Text текстового окна?

5. Почему строки кода между If Then и End If записываются с отступами?

6. В каком случае к свойству Text должна быть применена функция Val?

7. Каждый пример программирования, рассмотренный до сих пор, содержал процедуру:

```
Private Sub cmdВыход_Click()
End ' Завершить программу
End Sub
```

Для чего она используется?

8. Запишите следующую формулу, пользуясь операторами Visual Basic:

$$S = 1/2 (a + l)n$$

9. Нарисуйте воздушного змея. Какие два размера нужны для вычисления его площади? Разработайте приложение для вычисления площади воздушного змея (змея нарисуйте с помощью элемента управления Line (Линия)).

От редакции. Пожалуйста, присылайте разработанные проекты и ответы на вопросы в редакцию. Ваша активность будет учтена при подведении итогов учебного года в нашей газете.

Задачник

тебе удалось прочитать криптограмму отправь письмо в редакцию (в обоих случаях прописными буквами и, естественно, без пробелов).

Матрица, забытая дома Человеком Рассеянным, имеет вид:

```
100000
000000
101010
010100
000001
```

Правильные ответы прислали:

— Агеева Марья, Винакур Валерий, Исаев Николай, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харла-

*Ответы,
решения,
разъяснения*

**к заданиям,
опубликованным в газете
“В мир информатики”
№ 63 (“Информатика”
№ 22/2005)**

1. Статья “Шифрование текста с помощью табличек”

В табличке, приведенной в начале статьи, зашифрована фраза *во вторник у вас будет контрольная работа по теме кодирование информации*, в задании 1 для самостоятельной работы — *поздравляю*

² Для представления в компьютере. — **Ред.**

панова Злата и Шелюх Василий, поселок Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;

— Бармина Наталья, Ноговицына Галина и Шабурова Елена, поселок Сосьва Серовского р-на Свердловской обл., школа № 1, учитель **Бедрина Н.Т.**;

— Бурцев Анатолий, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Галушкова Карина, поселок Надвоицы, Республика Карелия, в школу пока не ходит (ей 6 лет);

— Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.А.**;

— Иванова Татьяна, г. Сегежа, Республика Карелия, школа № 5, учитель **Меньшиков В.В.**;

— Исакова Олеся (ученица 2-го класса), поселок Лимбяха Новоуренгойского р-на Тюменской обл., школа № 1, учитель **Исакова И.С.**;

— Копылова Валерия, поселок Жатай, Республика Саха (Якутия), школа № 1, учитель **Копылова Л.Ю.**;

— Крадинов Александр, г. Балашов Саратовской обл., гуманитарно-педагогический лицей-интернат, учитель **Сухорукова Е.В.**;

— Силаев Валерий, Ардатовский аграрный техникум, поселок Ардатов Нижегородской обл., преподаватель **Касаткина С.Ю.**;

— Усманов Сергей, село Актаныш Актанышского р-на, Республика Татарстан, школа № 2, учитель **Гилязова Г.М.**

2. Статья “Задача Иосифа Флавия — частный случай”

Напомним, что предлагалось решить задачу, которая в общем виде формулируется так: “По кругу размещены n человек. Задан параметр расчета k , то есть каждый k -й человек будет выбывать из круга. Требуется определить P — порядковый номер человека, который останется в кругу последним”, для частного случая — когда $k = 2$.

Решение

А. Нетрудно убедиться, что когда число n есть степень двойки, то последним в кругу останется человек, имевший в исходной нумерации номер 1.

Б. Когда число n — любое (например, 20), то:

1) сначала выбудет какое-то число людей до момента, когда в кругу останется количество людей, равное степени двойки, — в рассматриваемом примере — 16. Это означает, что выбыло 4 человека, а среди них последним выбыл 8-й;

2) среди оставшихся 16 человек того, кто стоит первым после последнего выбывшего на этот момент, можно рассматривать как имеющего номер 1, и именно он, в соответствии с пунктом А, останется последним в кругу. Номер этого человека в исходном состоянии — 9.

В общем виде методика нахождения номера этого человека такая: из общего количества участников нужно исключить максимальную степень двойки, оставшееся число умножить на 2 и прибавить единицу. Например, если в круге будет стоять 271 человек, то номер последнего оставшегося будет равен $(271 - 256) * 2 + 1 = 31$.

Правильное решение прислали:

— Бурцев Анатолий, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.А.**

Кучеренко Марина (в прошлом учебном году — ученица 11-го класса средней школы села Шереметьево Вяземского р-на Хабаровского края, учитель **Дивакова А.И.**) разработала следующую методику нахождения искомого номера человека, использующую двоичную систему счисления:

- 1) число n перевести в двоичную систему;
- 2) в полученном двоичном числе первую слева единицу приписать в конце (возможные начальные нули при этом не учитывать);
- 3) перевести получившееся число в десятичную систему счисления.

Проиллюстрируем ее на примерах $n = 20$ и $n = 21$ (см. таблицу ниже). Обоснование методики приведено в последнем столбце таблицы.

| Значение | n | Двоичное представление | n | Двоичное представление | Комментарий к расчету |
|--|-----|---|-----|---|---|
| | 20 | 10100 | 21 | 10101 | |
| Соответствующая максимальная степень двойки (m) | 16 | 10000 | 16 | 10000 | |
| Сколько человек выбыло к моменту, когда осталось m человек (v) | 4 | $\begin{array}{r} 10100 \\ - 10000 \\ \hline 100 \end{array}$ | 5 | $\begin{array}{r} 10101 \\ - 10000 \\ \hline 101 \end{array}$ | Исходное число n без первой единицы и возможных начальных нулей |
| Номер последнего выбывшего человека | 8 | 1000 (в двоичной системе $100 * 2$) | 10 | 1010 (в двоичной системе $101 * 2$) | Чтобы умножить v на 2 и увеличить полученное число на 1, можно приписать к v справа единицу |
| Номер человека, который останется в кругу последним | 9 | 1001 (в двоичной системе $100 * 2 + 1$) | 11 | 1011 (в двоичной системе $101 * 2 + 1$) | |

Ответы, решения, разъяснения

к заданиям,
опубликованным
в газете "В мир информатики"
№ 64/"Информатика" № 23/
2005

1. Задача об остановившихся часах

"Изюминка" решения заключается в том, что, уходя из дома, человек догадался пустить в ход свои стенные часы (которые остановились, но не сломались) и заметить по ним, в котором часу он вышел, а затем — в котором часу вернулся. Так, по своим часам человек смог определить, сколько времени он отсутствовал. Придя к знакомому и уходя от него, человек заметил показания его часов. Это дало ему возможность определить продолжительность пребывания у знакомого.

Вычитая из продолжительности времени, которое он отсутствовал дома, продолжительность пребывания у знакомого, человек получил количество времени, затраченного на дорогу туда и обратно. Прибавив половину этого количества времени к показанию часов товарища, когда он от него ушел, человек в сумме получил то показание часов, на которое следовало поставить его настенные часы.

В соответствии с этими рассуждениями Рамиль Абдуллин, средняя школа деревни Старый Бабич Кармаскалинского р-на Республики Башкортостан (учитель **Абдуллин Р.Ф.**), разработал следующий алгоритм решения задачи, которая стоит перед человеком, часы которого остановились:

1. Пустить в ход свои стенные часы.
2. Отметить время ухода из дома по этим часам.
3. Пойти к знакомому.
4. Отметить время прихода к знакомому по его часам.
5. Определить время ухода от знакомого (также по его часам).
6. Рассчитать время пребывания у знакомого (разность значений времени по пунктам 5 и 4).
7. Вернуться домой.
8. Установить время возвращения домой по своим настенным часам.
9. Рассчитать время, которое он отсутствовал дома (разность значений времени по пунктам 8 и 2).
10. Вычесть из значения времени, рассчитанного в пункте 9, значение, вычисленное в пункте 6.
11. Разделить полученное в предыдущем пункте значение на 2.
12. Прибавить значение, рассчитанное в пункте 11, ко времени, установленному в пункте 5.
13. Поставить стрелки стальных часов на время, вычисленное в пункте 12.

Кроме Рамиля, правильные ответы прислали:

- Бережных Анастасия, средняя школа села Качикатцы Хангалусского улуса, Республика Саха (Якутия), учитель **Яковлева М.Д.**;
- Бурцев Анатолий, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;
- Гайсина Галия и Гайсин Рашит, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**;
- Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.Л.**;
- Иванова Татьяна и Колокольцева Ксения, г. Сегежа, Республика Карелия, школа № 5, учитель **Меньшиков В.В.**

2. Ребусы со звездочками

Ребус 1

$$\begin{array}{r} \begin{array}{r} 1089709 \\ 108 \\ \hline 97 \\ 96 \\ \hline 109 \\ 108 \\ \hline 1 \end{array} \quad \left| \begin{array}{r} 12 \\ 90809 \end{array} \right. \end{array}$$

Ребус 2

$$\begin{array}{r} \begin{array}{r} 110768 \\ 1008 \\ \hline 996 \\ 896 \\ \hline 1008 \\ 1008 \\ \hline 0 \end{array} \quad \left| \begin{array}{r} 112 \\ 989 \end{array} \right. \end{array}$$

Правильные ответы прислали:

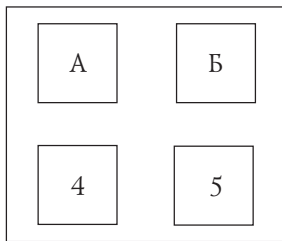
- Бережных Анастасия, средняя школа села Качикатцы Хангалусского улуса, Республика Саха (Якутия), учитель **Яковлева М.Д.**;
- Бурцев Анатолий, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;
- Гайсина Галия и Гайсин Рашит, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**;
- Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.Л.**;
- Горшкова Светлана, Мухарметова Эльвина, Орлов Александр и Шкенева Наталия, г. Стерлитамак, Республика Башкортостан, школа № 1, учитель **Орлова Е.В.**;
- Иванова Татьяна, г. Сегежа, Республика Карелия, школа № 5, учитель **Меньшиков В.В.**;
- Кирютин Алексей, Лось Олеся, Михаленко Анастасия и Николаев Никита, г. Струнино Владимирской обл., школа № 11, учитель **Волков Ю.П.**;

— Силаев Валерий, Ардатовский аграрный техникум, поселок Ардатов Нижегородской обл., преподаватель **Касаткина С.Ю.**;

— Шамсутдинова Альбина, г. Лениногорск, Республика Татарстан, школа № 8, учитель **Кашапова Р.Х.**

Четыре карточки

На столе лежат четыре карточки с надписями:



Известно, что на каждой карточке с одной стороны написана буква, с другой — цифра. Какие карточки достаточно перевернуть, чтобы выяснить, истинно или ложно следующее утверждение: “Если на одной стороне карточки — гласная буква, то на другой — четное число”? Ответ (с обоснованием) присылайте в редакцию.

Как получается молоко

Решите, пожалуйста, буквенный ребус:

$$\begin{array}{r}
 \text{К О Р О В А} \\
 + \quad \text{Т Р А В А} \\
 \hline
 \text{Д О Я Р К А} \\
 \hline
 \text{М О Л О К О}
 \end{array}$$

Известно, что КОРОВА больше, чем МОЛОКО. Как обычно, одинаковые цифры зашифрованы одинаковыми буквами, разные цифры — разными буквами.

Числа в две строки

Первые 8 натуральных чисел можно расставить в две строки так, что сумма чисел в верхней строке равна сумме чисел в нижней, а суммы чисел в столбцах также равны между собой:

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

Можно ли подобным образом расставить:

- 1) первые 10 натуральных чисел?
- 2) первые 12 натуральных чисел?

“Ломаем” голову

Итоги пятого тура конкурса № 41

**Внимание!
Конкурс**

Ответы за задания пятого тура прислали: Загольскайте Светлана, Казаков Евгений, Канзычакова Олеся, Сазанок Вассилий, Тиспирек Роман и Чернов Вячеслав, Арбатская средняя школа, Таштыпский р-н, Республика Хакасия, учитель **Медведева Т.А.**

Участником конкурса является также Иванов Максим, г. Шарыпово Красноярского края, школа № 8, учитель информатики **Балина И.А.**

Задача 1

Лучшие алгоритмы решения этой задачи содержат 7 команд. Приведем один из них:

| № пп | Команда | А | В | С | |
|------|---------|-------------------|---|---|---|
| Исх. | | 0 | 0 | 0 | |
| 1 | К2 | Наполнить В | 0 | 7 | 0 |
| 2 | К6 | Перелить из В в А | 3 | 4 | 0 |
| 3 | К7 | Перелить из В в С | 3 | 0 | 4 |
| 4 | К2 | Наполнить В | 3 | 7 | 4 |
| 5 | К10 | Вылить из А | 0 | 7 | 4 |
| 6 | К6 | Перелить из В в А | 3 | 4 | 4 |
| 7 | К7 | Перелить из В в С | 3 | 0 | 8 |

Игорь Клипов, приведший два принципиально разных оптимальных варианта, получает за эту задачу дополнительно 1 балл.

Задача 2

Многие участники конкурса привели алгоритмы решения задачи, при выполнении которых используется 12 л воды. Один из таких алгоритмов:

| № пп | Команда | А | В | С | |
|------|---------|-------------------|---|---|----|
| Исх. | | 0 | 0 | 0 | |
| 1 | К3 | Наполнить С | 0 | 0 | 12 |
| 2 | К9 | Перелить из С в В | 0 | 8 | 4 |
| 3 | К6 | Перелить из В в А | 3 | 5 | 4 |
| 4 | К5 | Перелить из А в С | 0 | 5 | 7 |
| 5 | К6 | Перелить из В в А | 3 | 2 | 7 |
| 6 | К5 | Перелить из А в С | 0 | 2 | 10 |

Все алгоритмы с использованием 12 л воды признаны правильными. А Николай Хе из г. Холмск Сахалинской обл. нашел еще лучшее решение:

| № пп | Команда | | А | В | С |
|------|---------|-------------------|---|---|----|
| Исх. | | | 0 | 0 | 0 |
| 1 | К2 | Наполнить В | 0 | 8 | 0 |
| 2 | К6 | Перелить из В в А | 3 | 5 | 0 |
| 3 | К5 | Перелить из А в С | 0 | 5 | 3 |
| 4 | К6 | Перелить из В в А | 3 | 2 | 3 |
| 5 | К5 | Перелить из А в С | 0 | 2 | 6 |
| 6 | К1 | Наполнить А | 3 | 2 | 6 |
| 7 | К5 | Перелить из А в С | 0 | 2 | 9 |
| 8 | К6 | Перелить из В в А | 2 | 0 | 9 |
| 9 | К9 | Перелить из С в В | 2 | 8 | 1 |
| 10 | К12 | Вылить из С | 2 | 8 | 0 |
| 11 | К7 | Перелить из В в С | 2 | 0 | 8 |
| 12 | К5 | Перелить из А в С | 0 | 0 | 10 |

При нем используется 11 л воды (соответствующие объемы выделены курсивом). За такое решение Николай получает дополнительно 1 балл.

Задача 3

Здесь оптимальное решение достаточно краткое:

| № пп | Команда | | А | В | С |
|------|---------|-------------------|---|---|----|
| Исх. | | | 0 | 0 | 0 |
| 1 | К2 | Наполнить В | 0 | 9 | 0 |
| 2 | К7 | Перелить из В в С | 0 | 0 | 9 |
| 3 | К2 | Наполнить В | 0 | 9 | 9 |
| 4 | К7 | Перелить из В в С | 0 | 7 | 11 |

Задача 4

Два алгоритма решения задачи, являющиеся лучшими:

1)

| № пп | Команда | | А | В | С |
|------|---------|-------------------|---|---|---|
| Исх. | | | 0 | 0 | 0 |
| 1 | К3 | Наполнить С | 0 | 0 | 9 |
| 2 | К9 | Перелить из С в В | 0 | 8 | 1 |
| 3 | К8 | Перелить из С в А | 1 | 8 | 0 |
| 4 | К3 | Наполнить С | 1 | 8 | 9 |
| 5 | К8 | Перелить из С в А | 4 | 8 | 6 |

2)

| № пп | Команда | | А | В | С |
|------|---------|-------------------|---|---|---|
| Исх. | | | 0 | 0 | 0 |
| 1 | К3 | Наполнить С | 0 | 0 | 9 |
| 2 | К8 | Перелить из С в А | 4 | 0 | 5 |
| 3 | К9 | Перелить из С в В | 4 | 5 | 0 |
| 4 | К3 | Наполнить С | 4 | 5 | 9 |
| 5 | К9 | Перелить из С в В | 4 | 8 | 6 |

Задача 5

Здесь лучшее решение одно:

| № пп | Команда | | А | В |
|------|---------|-------------------|---|----|
| Исх. | | | 0 | 0 |
| 1 | К1 | Наполнить А | 6 | 0 |
| 2 | К3 | Перелить из А в В | 0 | 6 |
| 3 | К1 | Наполнить А | 6 | 6 |
| 4 | К3 | Перелить из А в В | 2 | 10 |

Задача 6

Оптимальные алгоритмы решения задачи насчитывают 10 команд:

1)

| № пп | Команда | | А | В |
|------|---------|-------------------|---|----|
| Исх. | | | 0 | 0 |
| 1 | К1 | Наполнить А | 6 | 0 |
| 2 | К3 | Перелить из А в В | 0 | 6 |
| 3 | К1 | Наполнить А | 6 | 6 |
| 4 | К3 | Перелить из А в В | 0 | 12 |
| 5 | К1 | Наполнить А | 6 | 12 |
| 6 | К3 | Перелить из А в В | 2 | 16 |
| 7 | К6 | Вылить из В | 2 | 0 |
| 8 | К3 | Перелить из А в В | 0 | 2 |
| 9 | К1 | Наполнить А | 6 | 2 |
| 10 | К3 | Перелить из А в В | 0 | 8 |

2)

| № пп | Команда | | А | В |
|------|---------|-------------------|---|----|
| Исх. | | | 0 | 0 |
| 1 | К2 | Наполнить В | 0 | 16 |
| 2 | К4 | Перелить из В в А | 6 | 10 |
| 3 | К5 | Вылить из А | 0 | 10 |
| 4 | К4 | Перелить из В в А | 6 | 4 |
| 5 | К5 | Вылить из А | 0 | 4 |
| 6 | К4 | Перелить из В в А | 4 | 0 |
| 7 | К2 | Наполнить В | 4 | 16 |
| 8 | К4 | Перелить из В в А | 6 | 14 |
| 9 | К5 | Вылить из А | 0 | 14 |
| 10 | К4 | Перелить из В в А | 6 | 8 |

Участники конкурса, приведшие оба таких алгоритма, получают за эту задачу дополнительно 1 балл.

В задачах 7–10 лучшее решение одно.

Задача 7

| № пп | Команда | | А | В |
|------|---------|-------------------|---|----|
| Исх. | | | 0 | 0 |
| 1 | К2 | Наполнить В | 0 | 22 |
| 2 | К4 | Перелить из В в А | 6 | 16 |
| 3 | К5 | Вылить из А | 0 | 16 |
| 4 | К4 | Перелить из В в А | 6 | 10 |
| 5 | К5 | Вылить из А | 0 | 10 |
| 6 | К4 | Перелить из В в А | 6 | 4 |
| 7 | К5 | Вылить из А | 0 | 4 |
| 8 | К4 | Перелить из В в А | 4 | 0 |
| 9 | К2 | Наполнить В | 4 | 22 |
| 10 | К4 | Перелить из В в А | 6 | 20 |
| 11 | К5 | Вылить из А | 0 | 20 |
| 12 | К4 | Перелить из В в А | 6 | 14 |

Задача 8

| № пп | Команда | | А | В |
|------|---------|-------------------|---|---|
| Исх. | | | 0 | 0 |
| 1 | К1 | Наполнить А | 5 | 0 |
| 2 | К3 | Перелить из А в В | 0 | 5 |
| 3 | К1 | Наполнить А | 5 | 5 |
| 4 | К3 | Перелить из А в В | 2 | 8 |
| 5 | К6 | Вылить из В | 2 | 0 |
| 6 | К3 | Перелить из А в В | 0 | 2 |
| 7 | К1 | Наполнить А | 5 | 2 |
| 8 | К3 | Перелить из А в В | 0 | 7 |
| 9 | К1 | Наполнить А | 5 | 7 |
| 10 | К3 | Перелить из А в В | 4 | 8 |

Задача 9

| № пп | Команда | | А | В |
|------|---------|-------------------|---|----|
| Исх. | | | 0 | 0 |
| 1 | К2 | Наполнить В | 0 | 10 |
| 2 | К4 | Перелить из В в А | 7 | 3 |
| 3 | К5 | Вылить из А | 0 | 3 |
| 4 | К4 | Перелить из В в А | 3 | 0 |
| 5 | К2 | Наполнить В | 3 | 10 |
| 6 | К4 | Перелить из В в А | 7 | 6 |
| 7 | К5 | Вылить из А | 0 | 6 |
| 8 | К4 | Перелить из В в А | 6 | 0 |
| 9 | К2 | Наполнить В | 6 | 10 |
| 10 | К4 | Перелить из В в А | 7 | 9 |
| 11 | К5 | Вылить из А | 0 | 9 |
| 12 | К4 | Перелить из В в А | 7 | 2 |

Задача 10

| № пп | Команда | | А | В |
|------|---------|-------------------|---|----|
| Исх. | | | 0 | 0 |
| 1 | К1 | Наполнить А | 9 | 0 |
| 2 | К3 | Перелить из А в В | 0 | 9 |
| 3 | К1 | Наполнить А | 9 | 9 |
| 4 | К3 | Перелить из А в В | 7 | 11 |
| 5 | К6 | Вылить из В | 7 | 0 |
| 6 | К3 | Перелить из А в В | 0 | 7 |
| 7 | К1 | Наполнить А | 9 | 7 |
| 8 | К3 | Перелить из А в В | 5 | 11 |

Задача 11

Оптимальных алгоритма решения этой задачи — два:

1)

| № пп | Команда | | А | В |
|------|---------|-------------------|---|----|
| Исх. | | | 0 | 0 |
| 1 | К1 | Наполнить А | 5 | 0 |
| 2 | К3 | Перелить из А в В | 0 | 5 |
| 3 | К1 | Наполнить А | 5 | 5 |
| 4 | К3 | Перелить из А в В | 0 | 10 |
| 5 | К1 | Наполнить А | 5 | 10 |
| 6 | К3 | Перелить из А в В | 3 | 12 |
| 7 | К6 | Вылить из В | 3 | 0 |
| 8 | К3 | Перелить из А в В | 0 | 3 |
| 9 | К1 | Наполнить А | 5 | 3 |
| 10 | К3 | Перелить из А в В | 0 | 8 |
| 11 | К1 | Наполнить А | 5 | 8 |
| 12 | К3 | Перелить из А в В | 1 | 12 |
| 13 | К6 | Вылить из В | 1 | 0 |
| 14 | К3 | Перелить из А в В | 0 | 1 |
| 15 | К1 | Наполнить А | 5 | 1 |
| 16 | К3 | Перелить из А в В | 0 | 6 |

2)

| № пп | Команда | | А | В |
|------|---------|-------------------|---|----|
| Исх. | | | 0 | 0 |
| 1 | К2 | Наполнить В | 0 | 12 |
| 2 | К4 | Перелить из В в А | 5 | 7 |
| 3 | К5 | Вылить из А | 0 | 7 |
| 4 | К4 | Перелить из В в А | 5 | 2 |
| 5 | К5 | Вылить из А | 0 | 2 |
| 6 | К4 | Перелить из В в А | 2 | 0 |
| 7 | К2 | Наполнить В | 2 | 12 |
| 8 | К4 | Перелить из В в А | 5 | 9 |
| 9 | К5 | Вылить из А | 0 | 9 |
| 10 | К4 | Перелить из В в А | 5 | 4 |
| 11 | К5 | Вылить из А | 0 | 4 |
| 12 | К4 | Перелить из В в А | 4 | 0 |
| 13 | К2 | Наполнить В | 4 | 12 |
| 14 | К4 | Перелить из В в А | 5 | 11 |
| 15 | К5 | Вылить из А | 0 | 11 |
| 16 | К4 | Перелить из В в А | 5 | 6 |

Участники конкурса, приведшие оба таких алгоритма, получают за эту задачу дополнительно 1 балл.

Задача 12

Лучшее решение — одно:

| № пп | Команда | | А | В |
|------|---------|-------------------|---|---|
| Исх. | | | 0 | 0 |
| 1 | К1 | Наполнить А | 7 | 0 |
| 2 | К3 | Перелить из А в В | 0 | 7 |
| 3 | К1 | Наполнить А | 7 | 7 |
| 4 | К3 | Перелить из А в В | 5 | 9 |
| 5 | К6 | Вылить из В | 5 | 0 |
| 6 | К3 | Перелить из А в В | 0 | 5 |
| 7 | К1 | Наполнить А | 7 | 5 |
| 8 | К3 | Перелить из А в В | 3 | 9 |
| 9 | К6 | Вылить из В | 3 | 0 |
| 10 | К3 | Перелить из А в В | 0 | 3 |
| 11 | К1 | Наполнить А | 7 | 3 |
| 12 | К3 | Перелить из А в В | 1 | 9 |

По итогам всех пяти туров правильно решили 38 и более задач и являются лидерами “чемпионата” (в алфавитном порядке фамилий): Гайсин Рашит, Гайсина Галия, Глижинский Дмитрий, Дмитриев Юрий, Дубовицкий Николай, Зайцев Максим, За-

нозин Михаил, Кононов Владимир, Митин Илья, Уваровская Валентина, Хаустова Кристина, Хе Николай, Чапкевич Михаил, Шамсутдинова Альбина, а также команды ребят из городов Заозерный, Инта и Струнино. Список лидеров может быть дополнен после поступления новых ответов.

Почтовый ящик

Еще раз об исполнителе Водомере

Ученик школы № 33 г. Ярославля Анатолий Бурцев (учитель **Ярцева О.В.**) прислал в редакцию письмо. В нем он отмечает, что возможны задачи, которые исполнитель Водомер, описанный в условии конкурса № 41 (задания этого конкурса вызвали очень большой интерес у читателей газеты), решить не сможет. Например, если объем емкости А равен 6 л, емкости В — 8 л, то отмерить 3 л нельзя.

Предлагаем читателям исследовать проблему, обозначенную Анатолием, и определить, при каких значениях объема емкостей А и В (обозначим эти объемы a и b) с л воды не могут быть отмерены.

Свои мнения присылайте в редакцию.

Конкурс № 48 для учащихся

На компьютере “Macintosh” в текстовом редакторе был подготовлен документ. Его сохранили на дискете, а потом пытались открыть в текстовом редакторе Microsoft Word на компьютере типа IBM, однако из-за несовместимости форматов прочитать текст не удалось. *Что нужно сделать, чтобы можно было созданный документ прочитать и редактировать в редакторе Word?*

Ответ отправьте в редакцию до 15 мая по адресу: 121165, Москва, ул. Киевская, д. 24, “Первое сентября”, “Информатика” (или по электронной почте: inf@1september.ru). Пожалуйста, четко укажите в ответах свои фамилию и имя, населенный пункт, номер и адрес школы, фамилию, имя и отчество учителя информатики.

РЕДАКЦИОННАЯ ПОЛИТИКА

Общие вопросы

Вкладка “В мир информатики” выходит в каждом номере, начиная со с. 37 основной газеты. Она адресована учителям, но ее материалы могут использоваться и для самостоятельной работы учеников.

Интерактивность

Дорогие коллеги! Мы обязательно поощрим любую активность ваших учеников. Предложите им поучаствовать в конкурсах (которые специально не слишком сложны) — уверены, ребятам будет приятно увидеть свою фамилию в газете, в списке приславших правильные ответы. Может быть, они заметят какие-то погрешности, неточности в наших материалах, просто захотят поделиться своим мнением? Пусть обязательно напишут. Любую активность мы поощрим.

Можно ли опубликовать свои материалы?

Как и основная газета, вкладка является “открытой” — мы рассматриваем любые материалы, присланные нашими читателями и их учениками. За публикации выплачиваются гонорары в том же размере, что и в основной газете. При подготовке материалов для вкладки следует учитывать ее направленность и объем.

Технические вопросы

Если статья не посвящена конкретному языку программирования, для записи алгоритмов используется Школьный алгоритмический язык. Таким образом, мы стремимся сделать алгоритмы максимально понятными и переносимыми.

Учителю информатики: памятные даты и события апреля

Окончание. См. с. 1–2

Спрашивая с возмущением: “Кто станет работать даром?” — Гейтс, по всей видимости, полагал, что данный вопрос является риторическим, то есть уже содержащим в себе ответ. Но это было не так: уже существовал Time BASIC, распространенный практически бесплатно, а кроме того, находились люди, доказывавшие, что еще не все в этом мире решают деньги...

В апреле 1977 года был изготовлен первый модем для персонального компьютера, создателем которого стал уроженец городка Спартанберг (штат Южная Каролина) инженер Деннис С. Хайес.

В апреле 1984 года на рынке появилась первая так называемая “графическая рабочая станция” на чипе GE (Geometry Engine — геометрическая машина) для эффективной работы с объектами трехмерной графики (3D-graphics), которая была создана в компании Silicon Graphics [1].

В апреле 1992 года на рынке появилась операционная система Windows 3.1 — одна из первых версий Windows, получивших широкое распространение.

Операционная система DOS была выпущена в 1981 году одновременно с IBM PC и, несомненно, сыграла весьма значительную роль в успехе этих компьютеров. Однако создатели DOS оставили программистов, пишущих прикладные программы, практически наедине с “голым” компьютером. А вот для компьютеров Macintosh фирмы Apple (а также для машин фирмы Atari) ситуация была совсем иной. Компания Microsoft (разработчик MS-DOS) многократно предлагала фирме Apple продать лицензию на использование своей операционной системы на IBM PC и совместимых с ними машинах, но всякий раз получала отказ [10]. Тогда Microsoft начала разрабатывать собственную операционную систему Windows, чья первая версия появилась в 1985 году.

“Windows — серия популярных графических оболочек и операционных систем корпорации Microsoft. Представляет собой операционную среду — графический интерфейс, используемый для операций с файлами. Современную систему Windows можно представить в виде рабочего места, на котором расположено множество папок (Folders) и различных бумаг (Documents),

которые можно одновременно видеть и располагать в соответствии с заданными потребностями. Первая версия данной системы, **Windows 1.0,** была выпущена на рынок в 1985 году. Она обеспечивала лишь работу в поименованных окнах. Через два года появилась версия **Windows 2.0** (в конце 1987 г. — после выхода Windows/386 — ее переименовали в Windows/286). Вторая версия имела такие возможности, как перекрывающиеся окна (Windows) и пиктограммы (Icons). В 1990 году появилась полностью переработанная версия под названием **Windows 3.0.** Принципиальной новинкой ее была реализация возможности адресации ОЗУ (RAM) сверх 640 килобайт” [11].

Однако первую настоящую популярность обрела именно версия **Windows 3.1.**

В апреле 2003 года в Сан-Франциско была представлена 64-разрядная операционная система Windows DataServer 2003, на решение проблем безопасности которой затрачено 200 млн. долл. США [1].

Литература

1. *Полунов Ю.Л.* От абака до компьютера: судьбы людей и машин. Книга для чтения по истории вычислительной техники в двух томах. Т. II. М.: Издательско-торговый дом “Русская Редакция”, 2004.
2. *Шарле Д.Л.* По всему земному шару: Прошлое, настоящее и будущее кабелей связи. М.: Радио и связь, 1985.
3. *Частиков А.П.* Архитекторы компьютерного мира. СПб.: БХВ-Петербург, 2002.
4. *Носов В.А.* Клод Шеннон // Информатика № 33/1998.
5. *Поттосин И.В.* Андрей Петрович Ершов: жизнь и творчество // *Ершов А.П.* Избранные труды. Новосибирск: Наука, 1994.
6. *Ершов А.П.* Введение в теоретическое программирование (беседы о методе). М.: Наука, 1977.
7. *Ковальский И.* Дело жизни — информатика // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 10/1989.
8. *Ершов А.П., Кушниренко А.Г., Лебедев Г.В. и др.* Основы информатики и вычислительной техники. / Под ред. А.П. Ершова. М.: Просвещение, 1988.
9. *Гнеденко Б.В.* Колмогоров Андрей Николаевич // Большая советская энциклопедия. Изд. 3-е. М.: Советская энциклопедия, 1973. Т. 12.
10. *Фигурнов В.Э.* IBM PC для пользователя. Изд. 6-е, перераб. и доп. М.: ИНФРА-М, 1995.
11. *Мостицкий И.Л.* Новейший англо-русский толковый словарь по современной электронной технике. М.: ЛУЧШИЕ КНИГИ, 2000.

**ИЗДАТЕЛЬСКИЙ ДОМ
«ПЕРВОЕ СЕНТЯБРЯ»**
главный редактор —
А.С. Соловейчик

ГАЗЕТЫ
ИЗДАТЕЛЬСКОГО ДОМА
Первое сентября
гл. ред. — Е.В. Бирюкова,
индекс подписки — 32024;
Английский язык
гл. ред. — Е.В. Громушкина,
индекс подписки — 32025;
Библиотека в школе
гл. ред. — О.К. Громова,
индекс подписки — 33376;
Биология
гл. ред. — Н.Г. Иванова,
индекс подписки — 32026;
География
гл. ред. — О.Н. Коротова,
индекс подписки — 32027;
Дошкольное образование
гл. ред. — М.С. Аромштам,
индекс подписки — 33373;
Здоровье детей
гл. ред. — Н.В. Сёмкина,
индекс подписки — 32033;
Информатика
гл. ред. — С.Л. Островский,
индекс подписки — 32291;
Искусство
гл. ред. — М.Н. Сартан,
индекс подписки — 32584;
История
гл. ред. — А.Л. Савельев,
индекс подписки — 32028;
Литература
отв. сек. — С.Ф. Дмитренко,
индекс подписки — 32029;
Математика
и. о. гл. ред. — Л.О. Рослова,
индекс подписки — 32030;
Начальная школа
гл. ред. — М.В. Соловейчик,
индекс подписки — 32031;
Немецкий язык
гл. ред. — М.Д. Бузоева,
индекс подписки — 32292;
Русский язык
гл. ред. — Л.А. Гончар,
индекс подписки — 32383;
Спорт в школе
гл. ред. — О.М. Леонтьева,
индекс подписки — 32384;
Управление школой
гл. ред. — Я.А. Сартан,
индекс подписки — 32652;
Физика
гл. ред. — Н.Д. Козлова,
индекс подписки — 32032;
Французский язык
гл. ред. — Г.А. Чесновицкая,
индекс подписки — 33371;
Химия
гл. ред. — О.Г. Блохина,
индекс подписки — 32034;
Школьный психолог
гл. ред. — И.В. Вачков,
индекс подписки — 32898.

Гл. редактор
С.Л. Островский
Зам. гл. редактора
А.И. Сенокосов
Редакция
Е.В. Андреева
Д.М. Златопольский (редактор
вкладыш “В мир информатики”)
Л.Н. Картвелишвили
С.Б. Кишкина
Н.П. Медведева
Ю.А. Первин (редактор вкладыш
“Началка”)
Корректор
Е.Л. Володина
Дизайн и верстка
Н.И. Пронская

©ИНФОРМАТИКА 2006
Выходит два раза в месяц
При перепечатке ссылка
на ИНФОРМАТИКУ обязательна,
рукописи не возвращаются

Адрес редакции
и издателя:
Киевская, 24, Москва,
121165
тел. 249-48-96
Отдел рекламы: 249-98-70

Учредитель: ООО “Чистые пруды”
Зарегистрировано в Министерстве РФ по
делам печати. ПИ № 77-7230 от 12.04.2001.
Отпечатано в ОИД “Медиа-Пресса”,
ул. Правды, 24, Москва, ГСП-3, А-40, 125993
Тираж 6500 экз.
Срок подписания в печать по графику 22.03.2006.
Номер подписан 22.03.2006.
Заказ № 615508
Цена свободная

ИНДЕКС ПОДПИСКИ
для индивидуальных подписчиков **32291**
комплекта изданий **32744**

Тел.: (495) 249-31-38, 249-33-86. Факс (495) 249-31-84

Internet: inf@1september.ru
WWW: <http://www.1september.ru>

ИЗДАТЕЛЬСКАЯ ПОДПИСКА Тел.: (495) 249-47-58 E-mail: podpiska@1september.ru