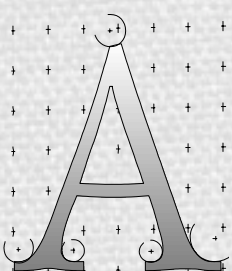


# ИНФОРМАТИК



Еженедельная газета Издательского дома «ПЕРВОЕ СЕНТЯБРЯ»  
ПОДПИСКА: (095) 249-47-58

## Программист

Программист должен обладать способностью превращать математику в абстракцию и логически выведенную в отечественной экономике платформу, что, что, что, из вида и бытия. Он должен сочетать акцентность бухгалтер и проницательность разведки, фантазия автора детективных романов с трезвой практичностью экономиста. А кроме того, программист должен иметь вкус к колоссальной работе, понимать интересы пользователя и много друзей.

«Андрей Петрович Еришов — известнейший ученый в области информатики, как в России, так и за рубежом, занялся вычислительными делами почти случайно» [1].



С Михайловым Шура Бура

Программирование было основано на операторных схемах, предложенных А.А. Лаврухиным. Под руководством Еришова разрабатывались одни из первых отечественных программирующих программ — для машин БЭСМ и Стрелка (программирующей программой называлась в то время комбинация языка программирования и транслятора). Позже под его руководством были созданы также известные системы, как АЛГОЛ (перевод в машинный язык), оптимизирующая система программирования для более сложных языков, чем Фортран, АЛГОЛ-6 и БЕТА. Они во многом определяли современную методологию оптимизирующей трансляции.



А.П. Еришов

Машин и работам по автоматизации программирования приносили и труды Еришова по теоретическому программированию, «чьи объектами являются математические абстракции программ» [5].

Андрей Петрович Еришов (1931–1988) поступил на работу в Институт математики и вычислительной техники Академии наук (ИТМ и ВТ) в 1953 году. В то время в ИТМ и ВТ формировались одни из первых в нашей стране структурных подразделений [2]. Спустя год Еришов заканчивает университет, а в 1957 году по приглашению академика от отдела автоматизации программирования в созданный незадолго до этого Вычислительный центр Академии наук. Вскоре, в связи с образованием Сибирского отделения Академии наук, по просьбе директоров Института математики и вычислительной техники академика С.П. Собольева он берет на себя обязанности организатора и фактического руководителя отдела программирования этого института и затем в 1960 году окончательно переезжает в Омск [2–4].

Позже отсюда Еришов уходит в состав созданного в 1964 году И. Маргуловым Сибирского отделения Академии наук, по просьбе директоров Института математики и вычислительной техники академика С.П. Собольева он берет на себя обязанности организатора и фактического руководителя отдела программирования этого института и затем в 1960 году окончательно переезжает в Омск [2–4].

Благодаря Еришову новосибирский Академгородок становится одним из ведущих центров в области информатики, а сам он оказывает большое влияние на развитие информатики в стране.

Помимо, как важна научно-организационная деятельность, при становлении новых научных направлений, Еришов уделяет ей много внимания не только в масштабах страны, но работает в редакциях международных изданий (Information Processing Letters, Theoretical Computer Science, Acta Informatica и другие [2]), а также участвует в международных, прежде всего Междумурской федерации по обработке информации (International Federation for Information Processing — IFIP), является членом нескольких ее комитетов и рабочих групп, активным участником ряда конгрессов. В 1980 году за творческую деятельность в ИТМ Еришов награжден «Серебряным сорденком» (Silver Core) — высшим знаком отличия, присвоенным в этой организации [2].

Для научной деятельности Андрея Петровича Еришова характерно сочетание теоретических исследований и практической работы. Теоретическое наследие Еришова обширно, оно включает около 200 работ (в том числе ряд монографий), не считая большого числа докладов, редакционных статей, отзывов и т.п.

Обобщая свой опыт руководства научными программами, Еришов выдвинул несколько тезисов, касающихся организации работ по информатике, сформулировав ряд общих принципов программирования как нового и своеобразного вида научной деятельности, затронув тот аспект, который позже будет назван «фундаментальными» в вычислительных системах [5].

Благодаря своему умению видеть новые перспективы в программировании, Еришов часто выступал как инициатор новых направлений исследований.

«Оптимизирующий транслятор выполняет жизненно важные функции программы для пользователя. Более быстрым и более компактным программам» [5].

«При этом сам русский язык «национально программируемая» («native language») был проблемой информатики» [5].

«Создатель формальной логики» / Информатика, № 41, 2000.

«Знакомьтесь, компьютер» / Пер. с англ. М. Мир, 1980.

«Знакомьтесь, компьютер» / Пер. с англ. М. Мир, 1980.

Еришов готовил специализированный курс по этой дисциплине для студентов старших курсов школы. В процессе программирования и школьного информатического обеспечения вообще, вел основную организационную работу [2]. Главным делом последних лет его жизни являлось создание школьной энциклопедии по информатике.

## Литература

1. Тарихин В.М. Андрей Петрович Еришов. Очерки истории информатики в России. Информатика // Информатика, № 20/98.
2. Глизицкий И.В. Андрей Петрович Еришов: жизнь и творчество // Еришов А.П. Избранные труды. Новосибирск: Наука, 1994.
3. Глизицкий И.В. Дядя жизни — информатика // Новые знания, наука, техника. Сер. «Вычислительная техника и ее применение» № 10/89.
4. Фундаментальное программирование // Информатика, № 16/2001.
5. Фундаментальное программирование // Информатика (Беседы о методе). М.: Наука, 1977.
6. Еришов А.П., Еришова Г.А., Глизицкий И.В. Основы информатики и вычислительной техники // Под ред. А.П. Еришова. М.: Просвещение, 1988.

«Создатель формальной логики» / Информатика, № 41, 2000.

«Знакомьтесь, компьютер» / Пер. с англ. М. Мир, 1980.

«Знакомьтесь, компьютер» / Пер. с англ. М. Мир, 1980.

# Л.Н. Картвеллишвили На стенде в кабинете информатики

## Аристотель, Лейбниц, Буль

Математическая логика тесно связана с логикой и объектами ее исследования. Основы логики, науки о законах и формах человеческого мышления (отсюда одно из ее названий — формальная логика), были заложены величайшим древнегреческим философом Аристотелем (384–322 гг. до н. э.), который в своих трактатах обстоятельно исследовал терминологию логики, подробно разобрал теорию умозаключений и доказательств, описал ряд логических операций, сформулировал основные законы мышления, в том числе законы противоречия и исключенного третьего [1–5]. Вклад Аристотеля в логику весьма велик, недаром другие ее называли — аристотелева логика.

Еще сам Аристотель заметил, что между созданной им наукой и математикой (тогда она именовалась арифметикой) много общего. Он пытался соединить две эти науки, а именно свести рассуждения, или, вернее, умозаключение, к вычислению на основании исходных положений. В одном из своих трактатов Аристотель вплотную приблизился к одному из разделов математической логики — теории доказательств.

В дальнейшем многие философы и математики развивали отдельные положения логики и иногда даже наметили контуры современного исчисления высказываний, но ближе всех к созданию математической логики подошел уже во второй половине XVII века выдающийся немецкий ученый Готфрид Вильгельм Лейбниц (1646–1716), указавший пути для перевода логики «из словесного царства, полного неопределенности, в царство математики, где отношения между объектами или высказываниями определяются совершенно точно» [6]. Лейбниц надевался даже, что в будущем философия, вместо того чтобы бесплодно спорить, станет брать бумажу и вычислять, кто из них прав [1]. При этом в своих работах Лейбниц затрагивал и джоннианскую систему исчисления.

Следует отметить, что идея использования двух символов для кодирования информации очень стара. Австралийские аборигены считали дождями, некоторые племена охотников-соборачков Новой Гвинеи и Южной Америки тоже пользовались двоичной системой счета. В некоторых африканских племенах передают сообщения с помощью барабана в виде комбинации звонких и глухих ударов. Знакомый всем пример двухсимвольного кодирования — азбука Морзе, где буквы алфавита представляются определенными сочетаниями точек и тире.

После Лейбница исследования в этой области вели многие выдающиеся ученые, однако настоящий успех пришел здесь к английскому математику-самоучке Джорджу Булю (1815–1864), целеустремленность которого не знала границ.

Материальное положение родителей Джорджа (отец которого был сапожником-мастером) позволило ему оканчивать лишь начальную школу для бедняков. Спустя какое-то время Буль, сменив несколько профессий, открыл маленькую школу, где сам преподавал [2]. Он много времени уделял самообразованию и вскоре ушелся идеями символической логики. В 1847 году Буль опубликовал статью «Математический анализ логики, или Опыт исчисления дедуктивных умозаключений», а в 1854 году появился главный его труд «Исследование законов мышления, на которых основаны математические теории логики и вероятностей».

Буль избрал своеобразную алгебру — систему обозначений и правил, применимую ко всем возможным объектам, от чисел и букв до предложений. Пользуясь этой системой, он мог закодировать высказывания (утверждения, истинность или ложность которых требовалось доказать) с помощью символов своего языка, а затем манипулировать ими, подобно тому, как в математике манипулируют числами. Основные операции булевой алгебры являлись комбинация (И), дизъюнкция (ИЛИ) и отрицание (НЕ).

Через некоторое время стало понятно, что система Буля хорошо подходит для описания электрических переключательных схем. Ток в цепи может либо протекать, либо отсутствовать, подобно тому, как утверждение может быть либо истинным, либо ложным. А еще несомненно действительный спутник, уже в XX столетии, ученые объединили созданный Джорджем Булем математический аппарат с двоичной системой исчисления, заложив тем самым основы для разработки цифрового электронного компьютера.

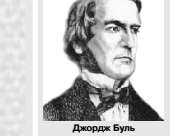
Отдельные положения работ Буля в той или иной мере затрагивались и до, и после него другими математиками и логиками [1, 2]. Однако сегодня в данной области именно труды Джорджа Буля причисляются к математической классике, а сам он по праву считается основателем математической логики и тем более важнейшим ее разделом — алгебры логики (булевой алгебры) и алгебры высказываний.



Аристотель



Готфрид Вильгельм Лейбниц



Джордж Буль

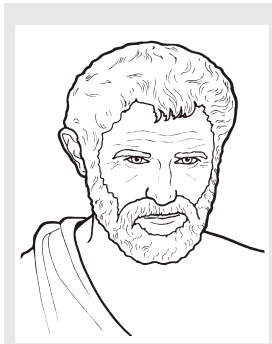
## Литература

1. Колосников А.Н., Драгалин А.Г. Введение в математическую логику. М.: Издательство Московского университета, 1982.
2. Соловьев Е.А. Интерпретация схемы логических операций // Новые знания, наука, техника. Сер. «Вычислительная техника и ее применение» № 5/88.
3. Дж. Уилкинс, Дж. Сигал. Дж. Буль. Введение в современную математику. Пер. с англ. М. Мир, 1983.
4. 1 + 1 = 7 // Информатика, № 3/2000.
5. Создатель формальной логики // Информатика, № 41, 2000.
6. Знакомьтесь, компьютер. Пер. с англ. М. Мир, 1980.

Еженедельная газета «Информатика» • Индекс подписки по каталогу «Роспечать» — 32291 • www.1september.ru

А р и с т о т е л ь ,

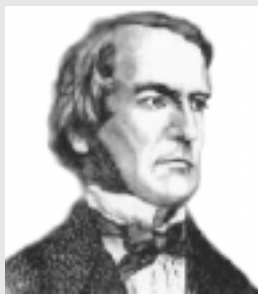
Л е й б н и ц ,      Б у л ь



Аристотель



Готфрид Вильгельм  
Лейбниц



Джордж Буль

#### Литература

1. Колмогоров А.Н., Драгалин А.Г. Введение в математическую логику. М.: Издательство Московского университета, 1982.

2. Соколов Е.А. Интегральные схемы логических операций // Новое в жизни, науке, технике. Сер. "Вычислительная техника и ее применение". Аппаратный состав ЭВМ, № 5/88.

3. Дж. Кемени, Дж. Снелл, Дж. Томпсон. Введение в конечную математику: Пер. с англ. М.: Мир, 1963.

4.  $1 + 1 = ?$  // Информатика, № 3/2000.

5. Создатель формальной логики // Информатика, № 41/2000.

6. Знакомьтесь: компьютер: Пер. с англ. М.: Мир, 1989.

Математическая логика тесно связана с логикой и обязана ей своим возникновением. Основы логики, науки о законах и формах человеческого мышления (отсюда одно из ее названий — формальная логика), были заложены величайшим древнегреческим философом **Аристотелем** (384—322 гг. до н. э.), который в своих трактатах обстоятельно исследовал терминологию логики, подробно разобрал теорию умозаключений и доказательств, описал ряд логических операций, сформулировал основные законы мышления, в том числе законы противоречия и исключения третьего [1—5]. Вклад Аристотеля в логику весьма велик, недаром другое ее название — аристотелева логика.

Еще сам Аристотель заметил, что между созданной им наукой и математикой (тогда она именовалась арифметикой) много общего. Он пытался соединить две эти науки, а именно свести размышление, или, вернее, умозаключение, к вычислению на основании исходных положений. В одном из своих трактатов Аристотель вплотную приблизился к одному из разделов математической логики — теории доказательств.

В дальнейшем многие философы и математики развивали отдельные положения логики и иногда даже намечали контуры современного исчисления высказываний, но ближе всех к созданию математической логики подошел уже во второй половине XVII века выдающийся немецкий ученый Готфрид Вильгельм **Лейбниц** (1646—1716), указавший пути для перевода логики "из словесного царства, полного неопределенностей, в царство математики, где отношения между объектами или высказываниями определяются совершенно точно" [6]. Лейбниц надеялся даже, что в будущем философы, вместо того чтобы бесплодно спорить, станут брать бумагу и вычислять, кто из них прав [1]. При этом в своих работах Лейбниц затрагивал и двоичную систему счисления.

Следует отметить, что идея использования двух символов для кодирования информации очень стара. Австралийские аборигены считали двойками, некоторые племена охотников-сборщиков Новой Гвинеи и Южной Америки тоже пользовались двоичной системой счета. В некоторых африканских племенах передают сообщения с помощью барабанов в виде комбинаций звонких и глухих ударов. Знакомый всем пример двухсимвольного кодирования — азбука Морзе, где буквы алфавита представлены определенными сочетаниями точек и тире.

После Лейбница исследования в этой области вели многие выдающиеся ученые, однако настоящий успех пришел здесь к английскому математику-самоучке Джорджу **Булю** (1815—1864), целеустремленность которого не знала границ.

Материальное положение родителей Джорджа (отец которого был сапожным мастером) позволило ему окончить лишь начальную школу для бедняков. Спустя какое-то время Буль, сменив несколько профессий, открыл маленькую школу, где сам преподавал [2]. Он много времени уделял самообразованию и вскоре увлекся идеями символической логики. В 1847 году Буль опубликовал статью "Математический анализ логики, или Опыт исчисления дедуктивных умозаключений", а в 1854 году появился главный его труд "Исследование законов мышления, на которых основаны математические теории логики и вероятностей".

Буль изобрел своеобразную алгебру — систему обозначений и правил, применимую ко всевозможным объектам, от чисел и букв до предложений. Пользуясь этой системой, он мог закодировать высказывания (утверждения, истинность или ложность которых требовалось доказать) с помощью символов своего языка, а затем манипулировать ими, подобно тому, как в математике манипулируют числами. Основными операциями булевой алгебры являются конъюнкция (И), дизъюнкция (ИЛИ) и отрицание (НЕ).

Через некоторое время стало понятно, что система Буля хорошо подходит для описания электрических переключательных схем. Ток в цепи может либо протекать, либо отсутствовать, подобно тому, как утверждение может быть либо истинным, либо ложным. А еще несколько десятилетий спустя, уже в XX столетии, ученые объединили созданный Джорджем Булем математический аппарат с двоичной системой счисления, заложив тем самым основы для разработки цифрового электронного компьютера.

Отдельные положения работ Буля в той или иной мере затрагивались и до, и после него другими математиками и логиками [1, 2]. Однако сегодня в данной области именно труды Джорджа Буля причисляются к математической классике, а сам он по праву считается основателем математической логики и тем более важнейших ее разделов — алгебры логики (булевой алгебры) и алгебры высказываний.

# И з о б р е т а т е л и

## Л и н е е к

Изобретателями первых логарифмических линеек являются англичане — математик и педагог Уильям Отред и учитель математики Ричард Деламайн.

Сын священника, Уильям Отред учился сначала в Итоне, а затем в Кембриджском королевском колледже, специализируясь в области математики. В 1595 году Отред получил первую ученую степень и вошел в совет колледжа. Ему было тогда чуть больше 20 лет. Позже Отред стал совмещать занятия математикой с изучением богословия и в 1603 году стал священником. Вскоре он получил приход в Олбьюри, близ Лондона, где и прожил большую часть жизни. Однако настоящим призванием этого человека являлось преподавание математики. «Все его мысли были сосредоточены на математике, — писал современник Отреда [1], — и он все время размышлял или чертил линии и фигуры на земле... Его дом был полон юных джентльменов, которые приезжали отовсюду, чтобы поучиться у него». Плату за обучение Отред не брал, хотя не был богатым. В 1631 году он написал для своих учеников учебник арифметики и алгебры «Ключ математики», пользовавшийся большой популярностью не только в XVII, но и в XVIII столетии.

Летом 1630 года у Отреда посетил его ученик и друг, лондонский учитель математики Уильям Форстер. Коллеги разговаривали о математике и, как бы сказали сегодня, о методике ее преподавания. В одной из бесед Отред критически отозвался о шкале Гюнтера, отметив, что манипулирование двумя циркулями отнимает много времени и дает низкую точность. [Валлиец Эдмунд Гюнтер построил логарифмическую шкалу, которая использовалась вместе с двумя циркулями-измерителями. Шкала Гюнтера представляла собой отрезок с делениями, соответствующими логарифмам чисел или тригонометрических величин. (Несколько таких шкал располагались на деревянной или медной пластинке параллельно друг другу.) С помощью цирку-



Джон Непер

лей-измерителей определяли сумму или разность длин отрезков шкалы, что в соответствии со свойствами логарифмов позволяло находить произведение или частное. Гюнтер ввел также общепринятое теперь обозначение *log* и термины *косинус* и *котангенс*. В 1620 году вышла книга Гюнтера, где дано описание его логарифмической шкалы, а также помещены таблицы логарифмов, синусов и котангенсов. Что же касается самого логарифма, то его изобрел, как известно, шотландец Джон Непер [1—9]. Видя недоумение Форстера, высоко ценившего данное изобретение, Отред показал своему ученику два изготовленных им вычислительных инструмента — две логарифмические линейки.

Первая линейка Отреда имела две логарифмические шкалы, одна из которых могла смещаться относительно другой, неподвижной. Второй инструмент представлял собой кольцо, внутри которого вращался на оси круг. На круге (снаружи) и внутри кольца были изображены «свернутые в окружность» логарифмические шкалы. Обе линейки позволяли обходиться без циркулей.

В 1632 году в Лондоне вышла книга Отреда и Форстера «Круги пропорций» с описанием круговой логарифмической линейки (уже иной конструкции), а описание прямоугольной логарифмической линейки Отреда дано в книге Форстера «Дополнение к использованию инструмента, называемого «Кругами пропорций», вышедшей в следу-

ющем году. Права на изготовление своих линеек Отред передал известному лондонскому механику Элиасу Аллену.

Линейка Ричарда Деламайна (который был в свое время ассистентом Отреда), описанная им в брошюре «Граммелогия, или Математическое кольцо», появившейся в 1630 году, тоже представляла собой кольцо, внутри которого вращался круг. Потом эта брошюра с изменениями и дополнениями издавалась еще несколько раз. Деламайн описал несколько вариантов таких линеек (содержащих до 13 шкал). В специальном углублении Деламайн поместил плоский указатель, способный двигаться вдоль радиуса, что облегчало использование линейки. Предлагались и другие конструкции. Деламайн не только представил описание линеек, но и дал методику градуировки, предложил способы проверки точности и привел примеры использования своих устройств.

По всей видимости, Уильям Отред и Ричард Деламайн изобрели логарифмическую линейку независимо друг от друга [1].

А в 1654 году англичанин Роберт Биссакер предложил конструкцию прямоугольной логарифмической линейки, общий вид которой сохранился до нашего времени...

### Литература

1. Гутер Р.С., Полунов Ю.Л. От абак до компьютера. Изд. 2-е, испр. и доп. М.: Знание, 1981.
2. Непер // Большая советская энциклопедия. Изд. 2-е. М.: Гл. науч. изд-во «Большая советская энциклопедия», 1954. Т. 29.
3. Стройк Д.Я. Краткий очерк истории математики: Пер. с нем. Изд. 4-е. М.: Наука, 1984.
4. Знакомьтесь: компьютер: Пер. с англ. М.: Мир, 1989.
5. Детская энциклопедия. М.: Изд-во Академии педагогических наук РСФСР, 1959. Т. 3.
6. Делман И.Я., Виленкин И.Я. За страницами учебника математики. М.: Просвещение, 1989.
7. Баврин И.И., Фрибус Е.А. Занимательные задачи по математике. М.: Гуманитарный издательский центр «ВЛАДОС», 1999.
8. Джон Непер // Информатика, № 15/2001.
9. Всего лишь линейка // Информатика, № 26/2001.

# Вильгельм Шиккард



Вильгельм Шиккард

Более 300 лет считалось, что автором первой счетной машины является Блез Паскаль [1–11]. И вот в 1957 году директор Кеплеровского научного центра Франц Гаммер сделал в Германии на семинаре по истории математики сенсационный доклад, из которого следовало, что проект первой счетной машины появился как минимум на два десятилетия раньше “паскалева колеса”, а сама машина была (по всей видимости) изготовлена в середине 1623 года [1].

Работая в городской библиотеке Штутгарта, Гаммер обнаружил фотокопию эскиза неизвестного ранее счетного устройства. (Оригинал находился в архиве выдающегося астронома и математика Иоганна Кеплера в Пулковской обсерватории близ Санкт-Петербурга.) Гаммеру удалось установить, что этот эскиз есть не что иное, как отсутствовавшее приложение к опубликованному ранее письму Иоганну Кеплеру профессора университета в Тюбингене Вильгельма Шиккарда (от 25 февраля 1624 года), где Шиккард, ссылаясь на чертеж, описывал изобретенную им счетную машину.

Машина содержала суммирующее и множительное устройства, а также механизм для записи промежуточных результатов [1–4]. Первый блок — шестиразрядная суммирующая машина — представлял собой соединение зубчатых передач. На каждой оси имелись шестерня с десятью зубцами и вспомогательное однозубое колесо — палец. Палец служил для того,

чтобы передавать единицу в следующий разряд (поворачивать шестеренку на десятую часть полного оборота, после того как шестеренка предыдущего разряда сделает такой оборот). При вычитании шестеренки следовало вращать в обратную сторону. Контроль хода вычислений можно было вести при помощи специальных окошек, где появлялись цифры. Для перемножения использовалось устройство, чью главную часть составляли шесть осей с “навернутыми” на них таблицами умножения.

После доклада Гаммера исследователи вспомнили и про другое письмо Шиккарда Кеплеру — от 20 сентября 1623 года, — на которое прежде не обращали должного внимания. В нем Шиккард сообщал, что осуществил механически то, что Кеплер делал алгебраически, а именно — сконструировал машину, автоматически выполняющую сложение, вычитание, умножение и деление. Шиккард писал, что Кеплер был бы приятно удивлен, если бы увидел, как машина сама накапливает и переносит влево десяток или сотню и как она отнимает то, что держит в “уме” при вычитании.

Гаммер обнаружил еще один набросок машины Шиккарда и письменные указания изготовлявшему ее механику Вильгельму Пфистеру.

Вильгельм Шиккард (1592–1636) появился в Тюбингене в 1617 году и вскоре стал профессором восточных языков местного университета. При этом он вел переписку с Кеплером и рядом немецких, французских, итальянских и голландских ученых по вопросам, касающимся астрономии. Обратив внимание на незаурядные математические способности молодого ученого, Кеплер порекомендовал ему заняться математикой. Шиккард прислушался к данному совету и достиг на новом поприще значительных успехов. В 1631 году он стал профессором математики и астрономии. А через пять лет Шиккард и члены его семьи умерли от холеры. Труды ученого были забыты.

Используя найденные Гаммером материалы, сотрудники Тюбингенского университета в начале 1960-х годов создали действующую модель машины Шиккарда.

Но была ли построена машина Шиккарда при его жизни? Здесь нет

однозначного ответа. Из уже упомянувшегося письма Шиккарда от 25 февраля 1624 года следует, что один, еще не совсем готовый, экземпляр машины, который находился у механика Пфистера, сгорел во время пожара.

На вопрос, использовал ли Паскаль при построении своей суммирующей машины идеи Шиккарда, ответ должен быть отрицательным [1]. Документы свидетельствуют о том, что никакие сведения о счетном устройстве 1623 года не дошли до научных кругов Парижа и, значит, Паскаль о нем знать не мог.

Машина Шиккарда была известна, по-видимому, лишь узкому кругу лиц и не повлияла на последующее развитие механизации счета, тем не менее имя Вильгельма Шиккарда находится в одном ряду с именами Блеза Паскаля, Готфрида Вильгельма Лейбница, Чарльза Бэббиджа, Пафнутия Львовича Чебышева, Германа Холлера и других выдающихся изобретателей счетных устройств XVII–XIX столетий [2].

Через десять лет после того, как Франц Гаммер сделал свое сенсационное сообщение, в Национальной библиотеке Мадрида были найдены два тома неопубликованных рукописей Леонардо да Винчи. Среди чертёжей обнаружили эскиз тринадцатиразрядного суммирующего устройства с десятизубыми колесами [1]. В целях рекламы оно было собрано фирмой IBM и оказалось вполне работоспособным.

Но являлся ли и данный проект первым?

## Литература

1. Гутер Р.С., Полунов Ю.Л. От абака до компьютера. Изд. 2-е, испр. и доп. М.: Знание, 1981.
2. Златопольский Д.М. Вычислительная машина Шиккарда // Информатика, № 33/2001.
3. Печерский Ю.Н. Этюды о компьютерах. Кишинев: Штиница, 1989.
4. Механические калькуляторы // Информатика, № 26/2001.
5. Вычислительные машины // Большая советская энциклопедия. Изд. 2-е. М.: Гл. науч. изд-во “Большая советская энциклопедия”, 1954. Т. 9.
6. Гиндикин С.Г. Рассказы о физиках и математиках. М.: Наука, 1985.
7. Знакомьтесь: компьютер: Пер. с англ. М.: Мир, 1989.
8. Стройк Д.Я. Краткий очерк истории математики: Пер. с нем. Изд. 4-е. М.: Наука, 1984.
9. Храмов Ю.А. Физики. Биографический справочник. Изд. 2-е. М.: Наука, Гл. редакция физико-математической литературы, 1983.
10. Леонов А.Г., Четвергова О.В. История компьютеров // Информатика, № 35/98.
11. Блез Паскаль // Информатика, № 6/2000.

# В е л и к и й ф р а н ц у з

*Когда я читаю Паскаля, мне кажется, что я читаю себя.*

Стендаль

**Б**лез Паскаль (1623—1662) — один из самых знаменитых людей в истории человечества. Ему посвящено огромное количество литературы. Каких только сторон жизни и наследия Паскаля не касалось “паскалеведение”! Конечно, особенно популярен он во Франции [1].

Паскаль умер, когда ему было 39 лет, но, несмотря на столь короткую жизнь, он вошел в историю как выдающийся математик, физик, философ и писатель [1—3]. Его именем названы единица давления (*паскаль*) и получивший чрезвычайно широкое распространение язык программирования.

Работы Паскаля охватывают самые разные области. Он является одним из создателей математического анализа, проективной геометрии, теории вероятностей, гидростатики (широко известен *закон Паскаля*, в соответствии с которым изменение давления в покоящейся жидкости передается в остальные ее точки без изменений), создателем механического счетного устройства — “паскалева колеса” — как говорили современники. Паскаль продемонстрировал, что воздух обладает упругостью, и доказал, что он имеет вес, открыл, что показания барометра зависят от влажности и температуры воздуха и потому его можно использовать для предсказания погоды [4]. Философские мысли Паскаля (после его смерти в разных вариантах, под разными названиями издавались материалы в виде книги, которую чаще всего называют “Мысли”) оказывали влияние на многих знаменитых людей и, в частности, на русских писателей — И.С. Тургенева, Ф.М. Достоевского, Л.Н. Толстого.

Некоторые из практических достижений Паскаля удостоились высшего отличия — сегодня мало кто знает имя их автора. Например, сейчас очень немногие скажут, что самая обыкновенная тачка — это изобретение Блеза Паскаля [1]. Ему же принадлежит идея омнибусов — многоместных конных экипажей с фиксированными маршрутами — первого вида регулярного общедоступного городского транспорта.

Уже в шестнадцатилетнем возрасте Паскаль сформулировал теорему о шестиугольнике, вписанном в коническое сечение (*теорема Паскаля*) [3]. (Известно, что позже он получил из своей теоремы около 400 следствий.) Через несколько лет Блез Паскаль создал механическое вычислительное устройство — суммирующую машину, которая позволяла складывать числа в десятичной системе счисления [2, 5—7]. В этой

машине цифры задавались путем соответствующих поворотов дисков (колесиков) с цифровыми делениями, а результат операции можно было прочитать в окошках — по одному на каждую цифру. Диски были механически связаны, при сложении учитывался перенос единицы в следующий разряд. Диск единиц был связан с диском десятков, диск десятков — с диском сотен и т.д. Если при повороте диск проходил через ноль, то следующий диск поворачивался на единицу “вперед”. Этот поворот, в свою очередь, мог вызвать поворот на единицу следующего диска (например, при прибавлении 1 к числу 99) и т.д. Главный недостаток суммирующей машины Паскаля состоял в неудобстве выполнения с ее помощью всех операций, кроме сложения. Однако изобретенный Паскалем принцип связанных колес стал основой, на которой строилось большинство вычислительных устройств на протяжении следующих трех столетий.

Блез Паскаль и другой великий француз, Пьер Ферма, стали основателями теории вероятностей, причем годом ее рождения часто называют 1654-й, когда Паскаль и Ферма независимо друг от друга дали правильное объяснение так называемого парадокса раздела ставки [8].

Два игрока играют в “безобидную” игру (т.е. шансы победить у обоих одинаковы), договорившись, что тот, кто первым выигрывает шесть партий, получит весь приз. Предположим, что игра остановилась до того, как один из них выиграл приз (например, первый игрок выиграл пять партий, а второй — три). Как справедливо разделить приз?

Хотя, вообще говоря, данная проблема не является парадоксом, безуспешные попытки некоторых видных ученых ее решить, а также неверные ответы создали легенду о парадоксе. Так, согласно одному решению следовало разделить приз в отношении 5 : 3, т.е. пропорционально выигранным партиям, согласно другому — в отношении 2 : 1 (здесь рассуждения велись, по всей видимости, следующим образом: поскольку первый игрок выиграл на две партии больше, что составляет треть часть от необходимых для победы шести партий, то он должен получить одну треть от приза, а оставшуюся часть нужно разделить пополам). А между тем делить надо в отношении 7 : 1.

И Паскаль, и Ферма рассматривали парадокс раздела ставки как задачу о вероятностях, установив, что справедливым является раздел, пропорциональный шансам первого игрока выиграть приз. Предположим, первому игроку осталось выиграть только одну партию, а второму для победы необходимо выиграть еще три партии, при-



чем игроки продолжают игру и играют все три партии, даже если некоторые из них окажутся лишними для определения победителя. Для такого продолжения все  $2^3 = 8$  возможных исходов будут равновероятными. Так как второй игрок получает приз только при одном исходе (если он выиграл все три партии), а в остальных случаях побеждает первый игрок, справедливым является отношение 7 : 1. (Паскаль и Ферма нашли также общее решение для случая, когда одному игроку для получения приза нужно выиграть еще  $n$  партий, а другому —  $m$  партий.)

Но, наверное, самой известной математической работой Блеза Паскаля является трактат об “арифметическом треугольнике”, образованном биномиальными коэффициентами (*треугольник Паскаля*), который имеет применение в теории вероятностей. Что же касается замечательной кривой 4-го порядка — *улитки Паскаля*, — то она названа так в честь отца Блеза Паскаля Этьенна, совмещавшего государственную службу с занятиями математикой [1, 3].

## Литература

1. Гиндикин С.Г. Рассказы о физиках и математиках. М.: Наука, 1985.
2. Знакомьтесь: компьютер: Пер. с англ. М.: Мир, 1989.
3. Стройк Д.Я. Краткий очерк истории математики: Пер. с нем. Изд. 4-е. М.: Наука, 1984.
4. Храмов Ю.А. Физики. Биографический справочник. Изд. 2-е. М.: Наука, Гл. редакция физико-математической литературы, 1983.
5. Леонов А.Г., Четвергова О.В. История компьютеров // Информатика, № 35/98.
6. Блез Паскаль // Информатика, № 6/2000.
7. Механические калькуляторы // Информатика, № 26/2001.
8. Секей Г. Парадоксы в теории вероятностей и математической статистике: Пер. с англ. М.: Мир, 1990.

# П а в е л   Л ь в о в и ч

## Ш и л л и н г

### И ЭЛЕКТРИЧЕСКИЙ ТЕЛЕГРАФ

**П**авел Львович Шиллинг (1786—1837) родился в Эстонии. Учился в кадетском корпусе и далее, после недолгой военной службы, с 1803-го по 1812 г. работал переводчиком, а затем секретарем в русском посольстве в Мюнхене [1]. В этом городе Шиллинг познакомился с немецким исследователем С.Т. Земмерингом и принимал участие в его электротехнических опытах. Первое изобретение Шиллинга относится к 1811 г., когда он предложил взрывать минные пороховые заряды на расстоянии посредством электрического запала, действующего от источника тока, связанного с ним двумя проводами. (Осенью 1812 г. Шиллинг продемонстрировал свое изобретение в Петербурге, взорвав мину под невиской водой.)



приема на ожидание передачи; и, наконец, приемник, или мультипликатор (т.е. устройство для усиления какого-либо действия).

Основной частью приемника являлась так называемая аstaticкая пара стрелок, которая малочувствительна к действию внешних магнитных полей. (Такую конструкцию предложил Ампер в 1821 году.)

Две магнитные стрелки укреплялись на общей медной оси и располагались параллельно од-

на другой, при этом их полюса были обращены в противоположные стороны. Стрелки могли вращаться в горизонтальной плоскости, причем одна стрелка располагалась внутри катушки из нескольких сотен витков изолированного провода, а другая — вне ее. К шелковой нити, на которой подвешивались стрелки, был прикреплен диск диаметром около четырех сантиметров. Одна из его сторон имела черный цвет, а другая — белый. В зависимости от направления тока в катушке магнитная стрелка поворачивалась в ту или иную сторону, и телеграфист, принимающий сообщение, видел либо черную, либо белую сторону диска. Если ток в катушку не поступал, то телеграфист видел ребро диска. Внизу располагался сосуд с ртутью для гашения колебаний аstaticческих стрелок и приведения их в первоначальное положение по окончании действия электрического тока.

Немного позже было предложено использовать шестимультимпликаторный телеграф. Здесь передача всех букв русского алфавита обеспечивалась отклонением стрелок одного или двух мультипликаторов из шести. (К примеру, для указания букв *А* и *Б* отклонялись стрелки только первого

\* Вольтов столб (впоследствии получивший название гальванического элемента) — источник непрерывного постоянного электрического тока, состоящий из чередующихся медных и цинковых кружков (пар), переложенных смоченными водой или кислотой суконными прокладками. Подобные элементы в виде сухих батареек, в которых применен электролит не в виде раствора, а в форме пасты, широко сейчас распространены. (Указанные устройства названы в честь итальянских исследователей Алессандро Вольта и Луиджи Гальвани.)

мультипликатора, причем при передаче буквы *А* его диск поворачивался к принимающему сообщению белой стороной, а при передаче буквы *Б* — черной.) Для передачи букв латинского алфавита хватало пяти мультипликаторов. Вскоре приемник стал содержать семь таких приборов, смонтированных на общей раме (седьмой мультипликатор использовался в устройстве вызова).

Первая публичная демонстрация нового телеграфа происходила в октябре 1832 года на квартире Шиллинга в Санкт-Петербурге [1, 3]. Первую телеграмму, состоящую из десяти слов, принял лично Павел Львович Шиллинг.

Через три года он с успехом продемонстрировал телеграф на съезде естествоиспытателей и врачей в Бонне. Вскоре русское правительство образовало “Комитет для рассмотрения электромагнетического 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 г. Комитет поручил Шиллингу устроить телеграфное сообщение между Петергофом и Кронштадтом и для этого составить проект и смету. Выполнить поручение ученый не успел: летом 1837 г. Павел Львович Шиллинг скончался.

Иногда пишут, что электромагнитный телеграф изобрел американец Сэмюэл Морзе (собравший в 1837 г. “аппарат Морзе”). Однако известно, что телеграф, основанный на принципе электромагнитного воздействия, был изобретен Шиллингом. Причина путаницы проста [1]: в одном случае термин *электромагнитный* употребляется для указания принципа действия, а в другом — для указания факта применения электромагнита: в конструкции приемника Морзе использовал электромагнит.

Ну а начиная с середины 1840-х годов телеграф стал быстро распространяться по всему миру...

#### Литература

1. Шарле Д.Л. По всему земному шару: Прошлое, настоящее и будущее кабелей связи. М.: Радио и связь, 1985.
2. Электрический телеграф: первые шаги // Информатика, № 48/2000.
3. Свириденко С.С. Современные информационные технологии. М.: Радио и связь, 1989.

Б О Р И С

С Е М Е Н О В И Ч

Я К О Б И

*“...я здесь чувствую себя очень хорошо. Но предупреждаю вас одновременно: остерегайтесь, как бы русские также и в научном отношении не превзошли вас, и ни в коем случае не думайте, что вы можете почитать на ваших немецких лаврах и что они не будут у вас отняты. Здесь имеет место разносторонняя деятельность”.*

Из письма Б.С. Якоби брату из Петербурга (1837 г.).

Немец по происхождению, Борис Семенович (Мориц Герман) Якоби (1801—1874) принял в 1837 году русское подданство и считал Россию “вторым отечеством, будучи связан с ней не только долгом подданства и тесными узами семьи, но и личными чувствами гражданина” [1]. Выдающийся физик и электротехник, член Петербургской академии наук Якоби всегда подчеркивал, что его изобретения принадлежат России.

Младший брат Бориса Семеновича, Карл Густав Якоби (1804—1851), был видным немецким математиком, одним из создателей теории эллиптических функций. Карл Якоби внес существенный вклад в развитие ряда областей математики. Например, он ввел в употребление функциональные определители (т.е. определители, составленные из функций) и указал на их роль при замене переменных в кратных интегралах [2]. В его честь получивший потом широкое применение в математике функциональный определитель специального вида назван *якобианом* [3, 4].

Что же касается Бориса Семеновича Якоби, то его научные интересы были связаны главным образом с физикой и особенно с электромагнетизмом [1, 5—7], причем ученый всегда стремился найти практическое применение своим открытиям.

В 1834 году Якоби изобрел электродвигатель с вращающимся рабочим валом, работа которого была основана на притягивании разноименных магнитных полюсов и отталкивании одноименных. (Это то же самое явление, которое заставляет магнитную стрелку компаса поворачиваться одним концом к северу, другим — к югу.) Немного позже появилась другая мо-

дель новой “магнитной машины”.

В 1839 году Якоби вместе с академиком Эмилием Христиановичем Ленцем (1804—1865) построил два усовершенствованных и более мощных электродвигателя. Один из них был установлен на большой лодке и вращал ее гребные колеса. При испытаниях лодка с экипажем из четырнадцати человек поднималась против течения Невы, борясь со встречным ветром. Данное сооружение представляло собой первое в мире электрическое судно. Другой электродвигатель Якоби — Ленца катил по рельсам тележку, в которой мог находиться человек. Эта скромная тележка придумана “бабушкой” трамвая, троллейбуса, электропоезда, электрокара. Правда, сидеть в ней было не очень удобно, поскольку свободного места там почти не оставалось из-за батареи. Других источников электрического тока тогда не знали [5]. (При этом элементы батарей быстро выходили из строя: цинковый электрод в них разрушался — “сгорал”, как сгорает уголь в топке паровой машины. Но уголь был дешев, а цинк в то время стоил очень дорого. Получалось, что работа электродвигателя с батареями почти в 12 раз дороже, чем работа паровой машины.)

А пятью годами раньше Якоби изобрел гальванопластику. (Гальванопластика — получение металлических копий с металлического и неметаллического оригинала путем электролиза, т.е. разложения веществ при прохождении через них постоянного электрического тока.) И вскоре данное открытие получило признание во всем мире. В Петербурге было создано предприятие, которое делало с помощью гальванопластики барельефы и статуи для украшения Исаакиевского собора, Эрмитажа, Зимнего дворца, золотило листы кровли для куполов, производило медные копии с форм для печатания денег, а также географических карт, почтовых марок, художественных гравюр.

Якоби предложил около десяти конструкций телеграфных аппаратов, в том числе буквопечатный аппарат (в 1850 г.), одним из первых в мире построил кабельные телеграфные линии: в Петербурге [Зимний дворец — Глав-



ный штаб (в 1841 г.) и Зимний дворец — Главное управление сообщений и публичных зданий (в 1842 г.)] и линию Петербург — Царское Село протяженностью около 25 км (в 1843 г.). Значительный интерес с технической точки зрения представляли его проекты кабельных линий Петербург — Москва и Петербург — Петергоф [1]. Большие заслуги имеет Якоби в создании подземных и подводных кабелей, в разработке технологии их производства, в подборе электроизоляционных материалов.

Много сделал этот ученый и для создания отечественного электротехнического оборудования. Он построил ряд электротехнических приборов, например, вольтметр, проводочный эталон сопротивления, несколько конструкций гальванометров, регулятор сопротивления.

“Я черпал из науки только то, что ведет или обещает привести к практическим результатам. Я поставил себе задачу примирить науку и технику, стереть неоправданное различие, которое установили между теорией и практикой”, — говорил Якоби [6].

Важное значение для России имели труды Якоби, касающиеся организации электротехнического образования [1, 7]. В начале 1840-х годов он составил и прочитал первые курсы прикладной электротехники, подготовил программу теоретических и практических занятий.

#### Литература

1. Якоби Борис Семенович // Большая советская энциклопедия. Изд. 2-е. М.: Гл. науч. изд-во “Большая советская энциклопедия”, 1957. Т. 49.
2. Стройк Д.Я. Краткий очерк истории математики: Пер. с нем. Изд. 4-е. М.: Наука, 1984.
3. Мышкис А.Д. Лекции по высшей математике. Изд. 4-е. М.: Наука, 1973.
4. Бронштейн И.Н., Семендяев К.А. Справочник по математике для инженеров и учащихся втузов. Изд. 13-е, исправленное. М.: Наука, 1986.
5. Детская энциклопедия. М.: Издательство Академии педагогических наук РСФСР, 1960. Т. 5.
6. От машин до роботов: очерки о знаменитых изобретателях, отрывки из документов, научных статей, воспоминаний, тексты патентов / Сост. М.Н. Ишков. (В 2 кн. Кн. 1.) М.: Современник, 1990.
7. Подданный России // Информатика, № 8/2001.



Во главе русской математики середины и второй половины XIX века стоял Пафнутий Львович Чебышев (1821—1894). Чебышев был воспитанником Московского университета, который он окончил в 1841 году. В этом учебном заведении Чебышев защитил и магистерскую диссертацию “Опыт элементарного анализа теории вероятностей”, и данная область стала одним из основных предметов его научных занятий [1].

Таким образом, в Москве начинал формироваться новый центр математических исследований. Позже, в 1860-е годы, на базе кружка математиков и механиков, собиравшихся у учителя Чебышева по университету профессора Н.Д. Брашмана, образовалось Московское математическое общество.

Чебышев поддерживал начинания подобного рода, однако участия в их организации не принимал, поскольку в 1847 году переехал в Петербург, где работал до своей кончины. Тридцать пять лет он читал лекции в Петербургском университете, являясь уже с 1853 года членом Академии наук. Преподавательская деятельность Чебышева была весьма плодотворной, причем он продолжал опекать своих учеников и по окончании ими университетского курса. “Одной из незабвенных заслуг Чебышева как учителя русских математиков было то, что он своими работами и указаниями в ученых беседах наводил своих учеников на темы для самостоятельных исследований и обращал их внимание на такие вопросы, занятия которыми всегда приводили к более или менее ценным результатам”, — отмечали его воспитанники.

Славу видного математика создали Чебышеву уже первые его работы по теории чисел — докторская диссертация “Теория сравнений” (1849 г.),

Произносится Чебышёв.

## М а т е м а т и к

представляющая собой оригинальный курс теории чисел, и приложение к ней.

В интегральном исчислении Чебышев первый доказал теорему об интегрируемости так называемых дифференциальных биномов только в трех случаях, которые были, по существу, известны еще Ньютону.

В теории вероятностей к заслугам Чебышева относятся, в частности, обобщение закона больших чисел, введение метода моментов, центральная предельная теорема для сумм независимых случайных величин. Он не раз читал курс теории вероятностей в Петербургском университете, причем заинтересовал ею самых выдающихся своих учеников А.А. Маркова (1856—1922) и А.М. Ляпунова (1857—1918), которые тоже внесли существенный вклад в эту науку. В значительной мере благодаря трудам школы Чебышева теория вероятностей, развиваясь в связи с запросами естествознания и прикладных наук, смогла достичь положения ведущей математической дисциплины.

Чебышев много занимался кинематикой механизмов — и теоретически, и практически. Ему принадлежит множество оригинальных конструкций. Например, он изобрел “стопходящую” машину, имитирующую движение животного при ходьбе [2], и арифмометр, в котором впервые была достигнута автоматизация выполнения всех арифметических действий.

*Арифмометр Чебышева* состоял из двух основных частей: суммирующей машины, сконструированной в 1878 году, и приставки для умножения, появившейся примерно пятью годами позже [3, 4]. После установки множителя и множителя надлежало только вращать рукоятку, обороты которой либо передавались на механизм переноса (как в обычном арифмометре), либо заставляли передвигаться на один разряд основной счетчик (суммирующую машину) относительно этого механизма. Для автоматизации всего процесса служил специальный управляющий регистр, на цифровых колесах которого устанавливался множитель. При умножении на цифру разряда единиц множителя установка колеса единиц управляющего регистра “уменьшалась” с каждым срабатыванием механизма переноса на единицу, пока не останавливалась на позиции 0. В этой позиции колесо, препятствовавшее ранее перемещению основного счетчика относительно механизма переноса, позволяло осуще-

ствить такое перемещение на один разряд, после чего обороты рукоятки передавались уже на колесо десятков счетчика управления и т.д.

Автоматические арифмометры представляли собой устройства, производившие любые арифметические действия над заданными числами после их установки и нажатия клавиши требуемой операции. И первым таким устройством был арифмометр Чебышева. Следует подчеркнуть, что в счетном приборе, сконструированном Чебышевым, перенос единицы в следующий разряд производился постепенно, непрерывно [5], и этот принцип нашел широкое применение с появлением в арифмометрах электропривода [поскольку увеличилась скорость их работы и при дискретном (“прерывистом”) способе передачи неизбежно появлялись толчки, снижающие надежность машин].

Исследование механизмов привело Чебышева к математическим задачам нового типа и созданию новой теории наилучшего приближения функции, которая сначала разрабатывалась в школе Чебышева, а в двадцатом столетии выросла в современную конструктивную теорию функций.

Занимался Чебышев и классическим способом приближенного представления функций — интерполяцией. Здесь он использовал метод наименьших квадратов, что привело его к общей теории ортогональных многочленов (*многочлены Чебышева*).

Чебышев являлся сторонником взаимного обогащения теории и практики. В одной из его работ говорится [6]: “Сближение теории с практикой дает самые благоприятные результаты, и не одна только практика от этого выигрывает; сами науки развиваются под влиянием ее, она открывает им новые предметы для исследования, или новые стороны в предметах давно известных”.

### Литература

1. *Стройк Д.Я.* Краткий очерк истории математики: Пер. с нем. Изд. 4-е. М.: Наука, 1984.
2. Чебышев // Большая советская энциклопедия. Изд. 2-е. М.: Гл. науч. изд-во “Большая советская энциклопедия”, 1954. Т. 47.
3. Вычислительные машины // Большая советская энциклопедия. Изд. 2-е. М.: Гл. науч. изд-во “Большая советская энциклопедия”, 1954. Т. 9.
4. Пафнутий Львович Чебышев // Информатика, № 18/2000.
5. *Печерский Ю.Н.* Этюды о компьютерах. Кишинев: Штиница, 1989.
6. *Гродзенский С.Я.* Андрей Андреевич Марков. М.: Наука, 1987.



# Р е в о л ю ц и о н е р

*“То, что Кантор своей теорией множеств произвел революцию в математике, общеизвестно” [1].*

“Понятие множества принадлежит к числу первоначальных математических понятий и может быть пояснено только при помощи примеров. Так, можно говорить о множестве людей, живущих на нашей планете в данный момент времени, о множестве точек данной геометрической фигуры, о множестве решений данного дифференциального уравнения. Люди, живущие на нашей планете в данный момент времени, точки данной геометрической фигуры, решения данного дифференциального уравнения являются элементами соответствующих множеств” [2].

“В современной математике понятие множества считается одним из основных, с него начинается изложение традиционных математических дисциплин и построение новых математических теорий” [3].

Кстати, раздел теории множеств, где исследуются операции над множествами, называется алгеброй множеств, а алгебра множеств, в свою очередь, является частным случаем теории булевых алгебр.

Теория множеств была создана в основном трудами математиков XIX века. Уже первые работы в этой области [например, Бернгарда Больцано (1781—1848), Рихарда Дедекинда (1831—1916)], где рассматривались числовые множества или множества функций, затрагивали вопрос о количественном сравнении бесконечных множеств. Является ли бесконечность множества чисто отрицательным свойством, не допускающим расчленения, или же существуют различные ступени математической бесконечности, бесконечные множества различной количественной силы, различной “мощности”? Ответ на данный вопрос дал немецкий математик, уроженец Петербурга Георг Фердинанд Людвиг Филипп Кантор (1845—1918) [2, 4]. “Даже если только перечислить все те понятия, которые Кантор ввел в математику, то и тогда будет сразу ясно, как много он сделал для развития проблемы бесконечности, так как все введенные им понятия относятся именно к бесконечности. Но Кантор не просто



ввел отдельные новые понятия, а связал их единой теорией множеств, которая и положила начало новой эпохе в развитии математики...

Большой заслугой Кантора является введение понятия мощности, аналогичного понятиям *число и количество* в конечных множествах...

Введение понятия мощности дало возможность сравнивать различные бесконечности” [1].

Возможность сравнительной количественной оценки множеств опирается на понятие взаимно-однозначного соответствия (или биекции) между двумя множествами [5, 6]. Пусть каждому элементу множества  $A$  соответствует в силу какого-либо правила или закона некоторый определенный элемент множества  $B$ . Если при этом каждый элемент множества  $B$  оказывается поставленным в соответствие одному и только одному элементу множества  $A$ , то говорят, что между множествами  $A$  и  $B$  установлено взаимно-однозначное соответствие (или биективное отображение, или биекция). Между двумя конечными множествами можно установить биекцию тогда и только тогда, когда оба множества состоят из одного и того же количества элементов. Обобщая данный факт, Кантор определил количественную эквивалентность, или равномощность, как возможность установить между множествами взаимно-однозначное соответствие. Ценность понятия мощности множества связана с существованием неравномощных бесконечных множеств. Например, мно-

жество всех действительных чисел и множество всех натуральных чисел имеют разные мощности. Первое имеет мощность так называемого континуума [2], а второе является счетным множеством, то есть таким бесконечным множеством, которое можно привести во взаимно-однозначное соответствие с натуральным рядом чисел [7].

Заслуга Кантора состоит не только в решении проблемы мощности множества: он сделал также решительный шаг, рассмотрев множества, состоящие из элементов произвольной природы.

Эти и некоторые другие идеи Кантора встретили сопротивление со стороны многих ученых [1, 4, 8, 9], но впоследствии оказали большое влияние на развитие математики. “Отношение к теории множеств Кантора было разноречивым с самого начала. Весьма влиятельной была идея о неприемлемости бесконечности из-за ее парадоксов, а вместе с ней — о неприемлемости самой теории множеств...”

Но многими математиками теория множеств была принята восторженно, и они стали излагать целые отрасли математики на ее основе” [1].

## Литература

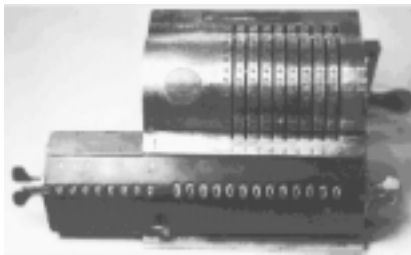
1. Бурова И.Н. Парадоксы теории множеств и диалектика. М.: Наука, 1976.
2. Ефимов Б.А. Множеств теория // Математическая энциклопедия. М.: Советская энциклопедия, 1982. Т. 3.
3. Пухначев Ю.В., Попов Ю.П. Математика без формул. М.: АО “Столетие”, 1995.
4. Стройк Д.Я. Краткий очерк истории математики: Пер. с нем. Изд. 4-е. М.: Наука, 1984.
5. Иванова О.А. Биекция // Математическая энциклопедия. М.: Советская энциклопедия, 1977. Т. 1.
6. Воднев В.Т., Наумович А.Ф., Наумович Н.Ф. Математический словарь высшей школы. Изд. 2-е. М.: Изд-во МПИ, 1987.
7. Коршунов Ю.М. Математические основы кибернетики. Изд. 3-е. М.: Энергоатомиздат, 1987.
8. Кантор // Большая советская энциклопедия. Изд. 2-е. М.: Гл. науч. изд-во “Большая советская энциклопедия”, 1953. Т. 20.
9. Плодотворные идеи немецкого математика // Информатика, № 47/2000.

**В И Л Ь Г О Д Т**

**О Д Н Е Р**

“Колесо Однера стало основой современных арифмометров. В дальнейшем на смену арифмометрам пришли настольные механические и электромеханические машины, а позднее — малые электронные цифровые машины” [1].

“У механических вычислительных машин счетный механизм состоит из счетчиков двух типов: 1) счетчик результатов, в котором образуется при сложении сумма, при вычитании — разность, при умножении — произведение, при делении — остаток; и 2) счетчик оборотов, который регистрирует коли-



Арифмометр системы В.Т. Однера. Первая серийная модель положила начало отечественному счетному машиностроению.

чество ходов машины (в нем при сложении подсчитывается количество слагаемых, при умножении — один из сомножителей, а при делении получается частное). Ввод цифровых данных осуществляется вручную с помощью специальных установочных механизмов, которые бывают рычажными (например, у арифмометра), ползунковыми, десятиклавишными и многоклавишными. При этом, в зависимости от типа механизма, время ввода одной цифры в машину колеблется от 1 до 0,2 — 0,25 с” [2].

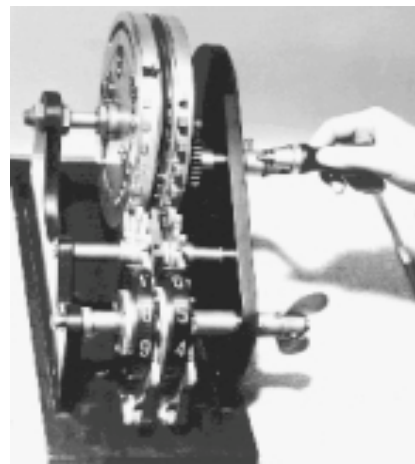
Одно из последних важных изобретений в области механической счетной техники сделал петербургский инженер Вильгодт Однер. В 1874 году он сконструировал колесо с выдвигающимися зубцами [3—5], а затем построил на его основе счетное устройство. Вскоре стали появляться различные разновидности арифмометра Однера, одной из которых был широко распространенный у нас в свое время арифмометр “Феликс”.

Арифмометр “Феликс” состоял из корпуса и каретки, которую, нажимая на специальный рычаг, можно было передвигать вправо и влево вдоль корпуса [3]. В крышке каретки имелись две группы окошек: справа — 13 окошек

счетчика результатов (в них при работе на арифмометре появлялись результаты сложения, вычитания и умножения), а слева — 8 окошек счетчика оборотов (в них при умножении появлялся множитель, а при делении — частное). Имелись также гасительные барашки, с чьей помощью можно было установить нули во всех окошках обоих счетчиков. Окошки счетчиков имели номера. При описании работы арифмометра говорили, что каретка находится в положении  $n$ , если стрелка на крышке корпуса указывала на окошко счетчика оборотов с номером  $n$ .

Внутри корпуса находился барабан Однера, приводимый во вращение рукояткой и состоящий из девяти основных и четырех дополнительных колес Однера. При каждом обороте барабана специальный “толкатель”, связанный с его осью, поворачивал цифровое колесо счетчика оборотов, находящееся против стрелки на крышке корпуса, на 1/10 полного оборота, причем при прямом вращении рукоятки в окошке появлялись последовательно белые цифры 1, 2, ..., 8, 9, а при обратном — красные цифры 1, 2, ..., 7, 8 и белая цифра 9.

При вычислении  $a \pm b$  ( $a$  и  $b$  — целые) надо было погасить счетчик результатов и поставить каретку в положение 1. Установив на барабане Однера число  $a$ , следовало прямым оборотом рукоятки перенести его в счетчик результатов. Установив затем на барабане число  $b$ , можно было его с



Двухразрядная модель арифмометра В.Т. Однера



помощью прямого оборота рукоятки прибавить к  $a$ , а с помощью обратного оборота — вычесть из  $a$ . Таким же образом определялась и алгебраическая сумма нескольких чисел.

Для умножения  $a$  на  $b$  следовало погасить оба счетчика и установить множитель  $a$  на барабане. Если, например,  $b = 437$ , то, переводя каретку в положение 3, нужно было сделать четыре прямых оборота (в счетчике результатов появлялось число  $a \cdot 400$ ), затем в положении 2 сделать три прямых оборота (в счетчике результатов к  $a \cdot 400$  добавлялось  $a \cdot 30$ ) и в положении 1 — семь прямых оборотов (в счетчик добавлялось еще  $a \cdot 7$ ). Таким образом, в счетчике оборотов оказывался множитель (437), а в счетчике результатов получалось произведение. В данном примере можно было обойтись и меньшим количеством оборотов, сделав в положениях 3 и 2 по четыре прямых оборота, а в положении 1 — три обратных оборота:

$$a \cdot 437 = a \cdot 400 + a \cdot 40 - a \cdot 3.$$

В 1890-х годах Однер и его компаньон наладили выпуск арифмометров — по 500 штук в год, причем скоро Однер стал единоличным хозяином предприятия.

К 1914 году в одной только России насчитывалось более 22 тысяч арифмометров Однера. Большими сериями они выпускались и за рубежом. В первой четверти XX века арифмометры Однера являлись единственными широко применявшимися математическими машинами [5, 6]. Вообще же колеса Однера использовались и в других счетных машинах [1—3, 7], имеющих существенные конструктивные отличия от арифмометра.

#### Литература

1. Борковский Б.А., Малиновский В.Н., Рабинович З.Л. Вычислительная техника // Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 1.
2. Корниенко Г.И. Вычислительная машина клавишная // Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 1.
3. Детская энциклопедия. М.: Изд-во Академии педагогических наук РСФСР, 1959. Т. 3.
4. Гутер Р.С., Полунов Ю.Л. От абака до компьютера. Изд. 2-е, испр. и доп. М.: Знание, 1981.
5. Печерский Ю.Н. Этюды о компьютерах. Кишинев: Штаница, 1989.
6. Механические калькуляторы // Информатика, № 26/2001.
7. Воюн В.П., Походило П.В. Вычислительная машина // Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 1.

# И н ж е н е р

Создателем первого действующего компьютера с программным управлением считают немецкого инженера Конрада Цузе (который с детства любил изобретать и, еще когда учился в школе, сконструировал модель машины для размена денег) [1].

О машине, способной выполнять вместо человека утомительные вычисления, он стал мечтать, когда был еще студентом. Не зная о работе Чарльза Бэббиджа, Цузе вскоре приступил к созданию устройства, во многом подобного Аналитической машине этого английского математика.

В 1936 году Цузе уволился из фирмы, где работал, чтобы отдавать больше времени постройке компьютера. Получив определенную сумму денег от друзей, он устроил “мастерскую” на маленьком столе в углу гостиной в доме родителей. Когда размеры машины стали расти, Цузе сначала придвинул к своему рабочему месту еще два стола, а затем переместился со своим прибором в середину комнаты. Примерно через два года компьютер, который занимал площадь около 4 м<sup>2</sup> и представлял собой хитросплетение реле и проводов, был готов.

Машина, названная им Z 1 (от *Zuse* — фамилии Цузе, написанной по-немецки), имела клавиатуру для ввода данных. Результат вычислений появлялся на панели — для этого использовались множество маленьких лампочек. В целом Цузе был доволен аппаратом, но считал ввод с клавиатуры неудобным и медленным. Он начал поиск других вариантов, и спустя какое-то время решение было найдено: команды для машины стали вводиться с помощью использованной 35-миллиметровой фотопленки, в которой пробивались отверстия. Машина, работавшая с перфолентой, получила название Z 2. А в 1941 году Конрад Цузе завершил постройку релейного компьютера Z 3, где применялась двоичная система счисления [1, 2].

Эти образцы машин были уничтожены при бомбардировках во время войны. Осталась лишь появившаяся в марте 1945 года машина Z 4 (которая использовалась для научных расчетов в Геттингенском университете), а позже Цузе изготовил еще модель Z 5. Основными элементами всех его компьютеров служили электромеханические реле, подобные тем, что применялись тогда, напри-

мер, в телефонных коммутаторах [1—5].

В 1942 году Цузе и австрийский инженер-электрик Хельмут Шрайер, время от времени сотрудничавший с Цузе, предложили создать устройство принципиально нового типа [1]. Они собрались перевести компьютер Z 3 с электромеханических реле на вакуумные электронные лампы, у которых нет движущихся деталей. Новая машина должна была действовать в сотни раз быстрее, чем любая из машин, имевшихся в то время в воюющей Германии. Однако данное предложение было отклонено: Гитлер наложил запрет на все “долговременные” научные разработки, поскольку был уверен в быстрой победе.

В тяжелые послевоенные годы Цузе, не имея возможности продолжать в полной мере работу непосредственно над компьютером, направил всю свою энергию на развитие теории. Он придумал эффективный способ программирования, причем не только для компьютера Z 4, но и для любой другой подобной машины.

Работая в одиночку, Цузе создал систему программирования, получившую название *Plankalkül* (Планкалькуль, “исчисление планов”) [2, 3]. Этот язык (превосходящий “в определенных моментах” по своим возможностям появившийся примерно на 12 лет позже Алгол) называют первым языком высокого уровня.

Цузе подготовил брошюру, где рассказал о своем творении и возможности его применения для решения разнообразных задач, включая сортировку чисел и выполнение арифметических действий в двоичной системе счисления (другие компьютеры того времени работали в десятичной системе), а также представил несколько десятков фрагментов программ на Планкалькуле для оценки шахматных позиций.

Не рассчитывая увидеть свой язык реализованным на компьютере, он отмечал:

“Планкалькуль родился исключительно как результат теоретической работы, без всякой связи с тем, появятся ли нет в обозримом будущем машины, подходящие к программам на Планкалькуле” [2].

Целиком работа Цузе была издана лишь в 1970-х годах. Эта публи-



кация заставила специалистов задуматься над тем, какое влияние мог бы оказать Планкалькуль, будь он широко известен раньше.

В США созданием релейных компьютеров занимались независимо от Цузе Джордж Штибитц (машины “Модель I”, ..., “Модель V”) и Говард Эйкен (“Марк I” и другие компьютеры) [1, 3, 4, 6—8]. А одной из наиболее совершенных “чисто релейных” машин была RVM-1, сконструированная и построенная под руководством специалиста по счетным устройствам Николая Ивановича Бессонова у нас в стране в середине 1950-х годов [4].

Релейные компьютеры имели невысокую скорость выполнения арифметических операций и невысокую надежность, что объяснялось прежде всего низким быстродействием и малой надежностью их основных счетных и запоминающих элементов — электромеханических реле. Кроме того, у этих машин был тот же недостаток, что и у Аналитической машины Бэббиджа: отсутствие хранимой в памяти программы. Однако они занимают весьма почетное место в истории компьютерной техники, так как являются первыми действовавшими автоматическими программно-управляемыми универсальными компьютерами.

## Литература

1. Знакомьтесь: компьютер: Пер. с англ. М.: Мир, 1989.
2. Язык компьютера: Пер. с англ. М.: Мир, 1989.
3. Частиков А.П. От калькулятора до суперЭВМ // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 1/88.
4. Гутер Р.С., Полунов Ю.Л. От абака до компьютера. Изд. 2-е, испр. и доп. М.: Знание, 1981.
5. Путь к транзистору // Информатика, № 6/2000.
6. Леонов А.Г., Четвергова О.В. История компьютеров // Информатика, № 35/98.
7. Семейство “Марк” // Информатика, № 26/2001.
8. Сначала — релейные // Информатика, № 26/2001.

# Д Ж О Р Д Ж Ш Т И Б И Т Ц И “М О Д Е Л И”

«Аналитическая машина Бэббиджа построена не была. Точно так же, как невозможно достигнуть Луны в деревянной ракете с двигателем внутреннего сгорания, так и невозможно было сделать Аналитическую машину из механических элементов. Ничего же другого техника XIX века предоставить не могла.

В конце концов был построен небольшой фрагмент машины... усилиями сына ученого, генерал-майора Бэббиджа. Он писал: “Я вполне уверен в том, что придет время, когда подобная машина будет построена и станет мощным средством распространения не только чистой математической науки, но и других областей знания. И я хочу ускорить приближение этого времени и помочь общей оценке работ моего отца, так мало... понятого массами даже образованных людей” [1].

Создателем первого действующего компьютера с программным управлением считается немецкий инженер Конрад Цузе [2—5]. В 1937 году в США, независимо от Цузе, постройкой машины, способной выполнять арифметические операции с комплексными числами, начал математик Джордж Штибитц [2, 3]. Его релейная машина “Модель I” появилась в 1939 году.

Электромеханическое реле состоит из двух основных частей: якоря и сердечника с обмоткой. Якорь, сделанный из мягкого железа, удерживается с помощью металлической пружины в оттянутом от сердечника положении. Если же по обмотке пропустить электрический ток, то сердечник становится магнитом и притягивает к себе якорь, преодолевая сопротивление пружины.

В 1985 году сотрудники Музея науки в Лондоне решили выяснить, возможно ли на самом деле построить вычислительную машину Чарльза Бэббиджа. После нескольких лет работы пришел успех: в ноябре 1991 года механический компьютер Бэббиджа впервые произвел серьезные вычисления (см.: *Свейд Д.* Механический компьютер Чарльза Бэббиджа: возвращение через полтора столетия. Пер. с англ. // Информатика, № 6/96).

К одному концу пружины, являющейся проводником электрического тока, присоединяется провод. По нему идет ток, который должен быть пропущен через весь элемент. На другом конце пружины этот ток с помощью контактов направляется по одному из двух проводников — в зависимости от положения якоря. “Таким образом, данный элемент является по существу переключателем, которым управляет наличие тока (цифра 1) или его отсутствие (цифра 0) в обмотке” сердечника [6].

В машинах Штибитца употреблялась двоично-пятеричная система представления чисел. Каждый десятичный разряд представлялся двумя цифрами — одна из них являлась цифрой в пятеричной системе и принимала значения от 0 до 4, другая — цифрой в двоичной системе. Получалось, что для указания любой десятичной цифры употреблялись семь реле, хотя включались каждый раз только два из них. Еще два реле использовались для указания знака (мантиссы и порядка). Двоично-пятеричная система позволяла осуществлять простой аппаратный контроль правильности работы машины [2].

Впервые “Модель I” демонстрировалась на заседании Американского математического общества в октябре 1940 года. Комплексные числа вводились в машину Штибитца, находившуюся в Нью-Йорке, с помощью расположенного в зале заседаний телетайпа; результаты передавались из Нью-Йорка по телеграфному каналу и выводились на печатающее устройство. (В частности, было получено произведение двух комплексных чисел.)

“Модель I” являлась специализированной вычислительной машиной, причем она не имела устройства для автоматического управления расчетами. Такое устройство появилось в “Модели II” — так называемом релейном интерполяторе, управляемом с помощью программы, “нанесенной” на перфоленту. В памяти машины могли поместиться пять пятиразрядных де-

сятичных чисел. После “Модели II”, появившейся в 1943 году, были построены еще две небольшие релейные машины: “Модель III” и “Модель IV”.

“Модель III”, известная как “баллистическая вычислительная машина”, имела несколько больший объем памяти, содержала 1300 реле и заменяла 25—40 человек, выполнявших с помощью настольных счетных машин вычисления, необходимые для составления баллистических таблиц.

“Модель IV” отличалась от своей предшественницы тем, что могла вычислять еще и значения тригонометрических функций.

Успех малых релейных машин способствовал созданию в 1944—1946 годах десятичной универсальной вычислительной машины “Модель V”, которая занимала площадь 90 м<sup>2</sup>, содержала около 9000 реле и имела в своем составе все блоки, предусмотренные в проектах Чарльза Бэббиджа. Ее запоминающее устройство состояло из 44 восьмиразрядных регистров, для ввода использовались перфоленты, а в качестве устройств вывода — перфораторы и телетайпы. Числа в машине представлялись в форме с плавающей запятой. Она выполняла сложение за 0,3 с, умножение — за 1 с, деление — за 2,2 с. Машина имела также специальные блоки, позволявшие вычислять функции  $\sin x$ ,  $\log x$ ,  $10^x$  и другие. По своим возможностям эта модель мало отличалась от предыдущей, но была значительно проще и имела более гибкие программные средства.

## Литература

1. *Печерский Ю.Н.* Этюды о компьютерах. Кишинев: Штеница, 1989.
2. *Гутер Р.С., Полунов Ю.Л.* От абака до компьютера. Изд. 2-е, испр. и доп. М.: Знание, 1981.
3. *Частиков А.П.* От калькулятора до суперЭВМ // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 1/88.
4. Знакомьтесь: компьютер: Пер. с англ. М.: Мир, 1989.
5. Язык компьютера: Пер. с англ. М.: Мир, 1989.
6. *Жоголев Е.А., Трифионов Н.П.* Курс программирования. М.: Наука, 1967.

Д Ж О Н

Б Э К У С

Создатели Фортрана, работавшие в фирме IBM, не подозревали, что этот язык получит такое признание. Когда в 1954 году они приступили к работе, информатика развивалась стихийно, и трудно было что-то планировать. Руководителем группы разработчиков был Джон Бэкус.



После службы в армии во время Второй мировой войны Бэкус попал в Нью-Йорк, где стал учиться в радиотехнической школе. «Тогда пределом моих стремлений, — признавался он позднее, — было сделать высококачественное звуковоспроизводящее устройство» [1]. Однако один из преподавателей убедил Бэкуса продолжить образование, и он поступил в Колумбийский университет. Так началась одна из самых плодотворных карьер в истории компьютерной техники.

В 1950 году Джон Бэкус, уже имея степень магистра математики, пришел в фирму IBM. Ему было тогда 25 лет. Вскоре он возглавил группу, разрабатывавшую интерпретатор «Быстрый кодировщик» для компьютера IBM 701, а позже принял участие в создании усовершенствованного варианта этой машины, модели IBM 704. В 1953 году Бэкус предложил создать для компьютера IBM 704 язык, позволяющий записывать команды почти в обычной алгебраической форме, и компилятор для него.

Данное предложение быстро нашло поддержку, так как в фирме искали пути, обеспечивающие увеличение сбита своих компьютеров, и потому стремились сделать их более привлекательными, «дружественными» пользователю. Как правило, инженеры, преподаватели, научные работники не имели ни времени, ни терпения изучать машинные коды или даже язык ассемблера, тем более что это требовало довольно глубокого знания того, как устроен и действует компьютер. Однако все они могли бы использовать язык высокого уровня, если бы его было достаточно легко освоить.

Первый отчет, связанный с созданием языка Фортран (FORTRAN, от *FORmula TRANslator* — транслятор, или переводчик формул), вышел в ноябре 1954 года [2–4], а первая версия языка была реализована на компьютере IBM 704 лишь через два с половиной года (получить эффективный

и надежный компилятор оказалось значительно более трудной задачей, чем предполагалось). В 1958 году появилась усовершенствованная версия — Фортран II, одно из достоинств которой состояло в том, что здесь можно было использовать и ассемблер, и затем другую версию — Фортран III. Большую популярность получила версия под названием Фортран IV, выпущенная в 1962 году. Однако стандарт языка еще отсутствовал, и это мешало переносить программы с машин одного типа на машины других типов. Ситуация изменилась в 1966 году — с введением стандарта, известного как Фортран 66.

Позже появились еще несколько версий, на основе которых была создана переработанная версия стандарта языка — Фортран 77, тоже ставшая весьма популярной. В 1984 году были рассмотрены предложения, связанные с выработкой новой стандартной версии Фортран 8x (позднее получившей название «Фортран 88»). В ней предусматривались некоторые изменения, касающиеся, в частности, работы с файлами.

Фортран до сих пор применяется при решении инженерных и научных задач, причем за прошедшие годы он значительно расширился и усовершенствовался [5], превратившись в язык, удобный для использования во многих областях (появился даже Visual Fortran). Многие идеи, заложенные в Фортране, нашли развитие в других языках программирования. Последним стандартом языка был Фортран 95. «Фортран будет всегда» — примерно таков смысл высказываний поклонников этого языка-долгожителя [3].

В конце 1950-х годов возникла идея создать еще один язык программирования — Алгол (ALGOL, от *ALGOrithmic Language* — алгоритмический язык). В Европе были обеспокоены диктатом компании IBM, давшей жизнь Фортрану, и других американских фирм на европейском рынке компьютеров [1]. В то же время американские программисты хотели получить универсальное средство программирования для научных задач, единый язык, устраивающий всех (Фортран, по понятным причинам, тут не подходил).

Первую версию языка, Алгол 58, приняли в ходе совещания, состоявшегося в Цюрихе весной 1958 года,

в котором участвовали ведущие специалисты из США и в их числе Джон Бэкус. Примерно через полтора года на конференции в Париже после восьмидневного обсуждения представители семи стран: США, ФРГ, Великобритании, Франции, Дании, Голландии и Швейцарии — утвердили улучшенную версию языка — Алгол 60, получившую потом наибольшую популярность [4, 6, 7]. Двумя годами позже на конференции в Риме были сформулированы поправки и дополнения к описанию языка, но своего названия (Алгол 60) он не изменил. Алгол 60 имел несколько слабых мест, и главное из них — бедные средства ввода-вывода. Этот недостаток был исправлен в языке Алгол 68 (появившемся в 1968 году и переработанном в 1975 году).

В Алголе нашли отражение новые идеи, касающиеся создания языков программирования, причем наиболее важные из этих идей относятся к блокам и процедурам. Он стал первым языком программирования с блочной структурой (позволяющей делить программы на блоки, разрабатываемые независимо друг от друга). Алгол оказал существенное влияние на формирование языков высокого уровня, а языки Паскаль, Си и Ада даже называют «алголоподобными».

Джон Бэкус активно помогал развитию Алгола (который многое унаследовал от Фортрана) и попутно разработал специальную систему определений для языков программирования. Например, определяя элемент «цифра», он указывал: «<цифра> := 0|1|2|3|4|5|6|7|8|9|». Вскоре такой способ записи назвали нормальной формой Бэкуса, или БНФ. (Позже датский астроном Петер Наур внес уточнения в методику Бэкуса, и эту форму стали называть формой Бэкуса — Наура, при этом сокращение осталось прежним — БНФ.) Потом БНФ долго употреблялась при описании языков высокого уровня [1].

#### Литература

1. Язык компьютера: Пер. с англ. М.: Мир, 1989.
2. Малыгина М.П., Частиков А.П. Языки программирования: Фортран // Новое в жизни, науке, технике. Сер. «Вычислительная техника и ее применение», № 8/88.
3. Долгожитель // Информатика, № 41/99.
4. Ставшие классическими // Информатика, № 26/2001.
5. Пройдаков Э.М., Теплицкий Л.А. Англо-русский толковый словарь по вычислительной технике, Интернету и программированию. Изд. 2-е, испр. и доп. М.: Издательско-торговый дом «Русская редакция», 2000.
6. Малыгина М.П., Частиков А.П. Языки программирования: Алгол // Новое в жизни, науке, технике. Сер. «Вычислительная техника и ее применение», № 9/88.
7. Универсальный, международный, классический // Информатика, № 8/2000.

# Ученый и организатор науки

*“Задача накопления, обработки и распространения (обмена) информации стояла перед человечеством на всех этапах его развития. В течение долгого времени основными инструментами для ее решения были мозг, язык и слух человека. Первое кардинальное изменение произошло в связи с приходом письменности, а затем изобретения книгопечатания...”*

*Положение в корне изменилось с появлением электронных вычислительных машин (ЭВМ)...”*

*В.М. Глушков*



щей машинки, а характерной чертой тоже являлась “простота взаимодействия с человеком”. В каком-то смысле это была первая персональная ЭВМ [10, 11]. Для нее употреблялся язык программирования Алмир-65 — “русифицированное развитие” языка Алгол-60. В модели МИР-1 (1968 г.) предусматривались ввод с перфоленты и вывод на нее. В этом компьютере использовались узлы

Виктор Михайлович Глушков (1923—1982) окончил Ростовский университет в 1948 году. В начале своей научной карьеры Глушков вел исследования в области абстрактной и топологической алгебры, а затем стал заниматься теоретическими и прикладными вопросами кибернетики. Получил важные результаты в теории цифровых автоматов, а также в области автоматизации проектирования ЭВМ, приложения компьютерной техники к управлению производственными процессами и экономикой, разработки новых принципов построения структур малых ЭВМ для инженерных расчетов [1]. С 1962 года являлся директором организованного им Института кибернетики Академии наук Украины (который носит теперь его имя). Под руководством Глушкова был создан целый ряд компьютеров и компьютерных систем [2]: “Киев”, “Днепр”, “Проминь”, МИР, “Проект”, “Львов”, “Дисплан”. Наиболее известными из них являются ЭВМ “Киев” и машины семейств “Днепр”, “Проминь”, МИР [3—11].

Машина “Киев” была выпущена в 1958 году. Эта ламповая ЭВМ, имевшая производительность 6—10 тыс. оп./с, впервые в нашей стране использовалась для дистанционного управления технологическими процессами [3—5].

“Днепр” — “первая отечественная цифровая управляющая вычислительная машина широкого назначения на полупроводниковых элементах. Создана... в 1961 году. Состоит из центральной вычислительной части и устройства связи с объектом” [6]. Вычислительная часть этого компьютера представляла собой самостоятельную универсальную цифровую вычислительную машину “средней производительности” (время выполнения операции сложения — от 29,5 до 57,5 мкс), имеющую оперативное запоминающее устройство “переменного” объема (употреблялись блоки емкостью по 512 слов; всего могло быть использовано до восьми блоков).

“Днепр-2” — «управляющая вычислительная система, ориентированная на применение в качестве центрального звена в информационно-управляющих системах на промышленных предприятиях. Состоит из двух основных частей — вычислительного комплекса “Днепр-21” и управляющего комплекса “Днепр-22”» [7]. Вычислительный комплекс предназначался для обработки информации, поступающей от внешних устройств, а также от управляющего комплекса, и мог использоваться “как самостоятельная вычислительная машина для обработки экономических данных и решения инженерно-технических задач”. Управляющий комплекс применялся “для приема информации от управляемого объекта, выдачи управляющих воздействий на объект, а также первичной обработки информации”. Кроме того, управляющий комплекс обеспечивал обмен информацией между оператором, следящим за технологическим процессом, и вычислительным комплексом.

“Проминь” — «семейство малых цифровых электронных вычислительных машин, предназначенных для автоматизации инженерных расчетов средней сложности. Для машин семейства характерны простота общения с человеком, малые размеры и потребление небольшого количества энергии» [8]. В семейство входили транзисторные ЭВМ: “Проминь” (выпущенная в 1962 г.); “Проминь-М” (1965 г.), отличающаяся от машины “Проминь” наличием вывода на “цифропечатающую” машинку, и “Проминь-2” (1967 г.).

МИР, “машина для инженерных расчетов — семейство малых электронных цифровых вычислительных машин, предназначенных для решения широкого круга инженерно-конструкторских математических задач” [9]. Машина МИР (как и “Проминь”) могла быть установлена в небольшой комнате. Ввод и вывод информации тут осуществлялись с помощью электрической пишу-

повышенной надежности. В 1969 году появилась ЭВМ “МИР-2”, имевшая дисплей со световым пером (он помогал быстро выводить, проверять и редактировать информацию и обеспечивал отображение на экране результатов решения задач), а также внешнюю память на магнитных картах [10]. Здесь применялся язык Аналитик — расширение языка Алмир.

Машины двух последних семейств (“Проминь” и МИР) широко использовались в научно-исследовательских организациях и вузах.

## Литература

1. Глушков // Большая советская энциклопедия. 3-е изд. М.: Изд-во “Советская энциклопедия”, 1971. Т. 6.
2. Глушков В.М. Основы безбумажной информатики. Изд. 2-е, испр. М.: Наука, 1987.
3. Частиков А.П. От калькулятора до суперЭВМ // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 1/88.
4. “Киев”, “Днепр” и другие // Информатика, № 26/2001.
5. Хоменко Л.Г. “Киев” // Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 1.
6. Хоменко Л.Г. “Днепр” // Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 1.
7. Кухарчук А.Г., Никитин А.И., Стогий А.А. “Днепр-2” // Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 1.
8. Хоменко Л.Г. “Проминь” // Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 2.
9. Хоменко Л.Г. МИР // Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 1.
10. Осипов Л.А. Языки программирования: Алмир-Аналитик // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 10/89.
11. Первый персональный компьютер // Информатика, № 37/99.

Г е н е р а т о р

ф а н т а с т и ч е с к и х

и д е й

В 1950-х годах специалисты по искусственному интеллекту начали искать язык, пригодный для манипулирования понятиями, выраженными словами и фразами на естественном языке. (Искусственный интеллект — раздел информатики, ставящий своей целью разработку методов решения задач, для которых отсутствуют формальные алгоритмы: понимание естественного языка, обучение, доказательство теорем, распознавание образов.) Первый результат тут был получен в виде семейства языков под названием ИПЛ (IPL, от *Information Processing Languages* — языки обработки информации), созданного одним из пионеров в области искусственного интеллекта Алленом Ньюэллом и его сотрудниками. Центральным для ИПЛ является понятие списка. Представляя данные в виде списка слов и символов, программист мог связать понятия в памяти компьютера приблизительно таким же образом, как, по мнению специалистов по искусственному интеллекту, они связываются в памяти человека [1].

Понятием списка заинтересовался и Джон Маккарти, разносторонний математик, один из ведущих исследователей в области искусственного интеллекта (причем сам термин *искусственный интеллект* был предложен именно им в 1956 году).

Маккарти родился в 1927 году в семье профсоюзного деятеля. Среди его многочисленных работ наиболее важные относятся к концу 1950-х годов, когда он работал в Массачусетском технологическом институте (МТИ). Маккарти был генератором самых фантастических идей, а также “человеком поразительной внешности с... шокирующими манерами, например, привычкой поворачиваться и уходить, оборвав разговор на полуслове” [2].

В 1957—1958 годах по инициативе Маккарти при МТИ сформировалась лаборатория искусственного интеллекта, и он вместе с несколькими студентами стал создавать язык высокого уровня, в котором понятие списка сочеталось с системой обозначений, заимствованной из специального раздела математики — так называемого лямбда-исчисления ( $\lambda$ -исчисления). Введенное в 1931 году математиком Алонзо Черчем, лямбда-исчисление



Джон  
Маккарти

оперирует всего тремя типами элементов [3]: символами, представляющими переменные и константы; скобками для группировки символов; обозначениями функций с использованием греческой буквы *лямбда*.

Появившийся вскоре язык получил название Лисп (LISP, от *LISt Processing* — обработка списков). Спустя какое-то время вышла статья Маккарти “Рекурсивные функции и символьные вычисления” [4] с математическим обоснованием применения Лиспа. “Лисп являлся для нас не просто языком, который используют для определенных целей, — говорил один из студентов, помогавших Маккарти, — им можно было любоваться, как прелестной вещью” [1].

Программы и данные Лиспа существуют в форме символьных выражений, хранящихся в виде “списковых” структур. Употребляются два вида объектов: атомы и списки. Атомы — это символы, применяемые для идентификации объектов, которые могут быть числовыми или символьными. Список — это последовательность (возможно, пустая) элементов, заключенных в круглые скобки, каждый из которых является либо атомом, либо списком. Для манипулирования списками употребляются так называемые списочные примитивы (примитивные функции), из которых получаются все другие функции обработки списков [1, 3, 5—7]. Первая реализация Лиспа была осуществлена на ламповом компьютере IBM 704. [У нас первый транслятор (интерпретатор) Лиспа был создан для БЭСМ-6 в конце 1960-х годов.]

Даже небольшой фрагмент программы на Лиспе способен содержать десятки пар скобок, куда заключены списки. Часто пары скобок вкладываются друг в друга, образуя список внутри списка, который, в свою очередь, также находится внутри списка (иногда бывает 8—10 уровней вложенности). Характерная особенность языка Лисп состоит в том, что и программы, и данные, обрабатываемые с их помощью, имеют одинаковую структуру.

В 1960-е годы предпринимались попытки аппаратной реализации Лиспа, однако реальная возможность решения этой задачи появилась позже, когда начало интенсивно развиваться производство интегральных схем. А с 1982 года Лисп-компьютеры (с реализацией раз-

личных версий языка) стали выпускать серийно целым рядом фирм.

Скорость и легкость, с которой можно писать, выполнять и изменять программы на Лиспе, способствовали появлению многочисленных поклонников этого языка и вне лабораторий искусственного интеллекта [1]. Подобно большинству популярных языков, Лисп имеет достаточно большое “потомство”. Так, в Великобритании употреблялся появившийся в 1960-х годах язык ПОП-2 (POP 2), названный в честь одного из его создателей Робина Дж. Поплстона из Эдинбургского университета.

Лиспу уже более сорока лет, и, несмотря на то, что этот язык имеет определенные недостатки (например, связанные с уже отмечавшейся необходимостью использования большого количества скобок), он весьма широко применяется на всех этапах создания систем искусственного интеллекта и вообще решения сложных логических задач. “LISP... в течение длительного времени рассматривался как кандидат на стандарт языка в области искусственного интеллекта. Позднее в эти притязания вмешался Prolog” [8].

В 1959 году Маккарти предложил концепцию разделения времени (режим работы компьютера, позволяющий нескольким программистам, работая за разными терминалами, одновременно использовать его память и процессор) [1, 2]. Тем самым специалисты избавлялись от необходимости ждать несколько часов или даже дней, пока их задания будут обработаны машиной. (По другим данным [9], идею разделения времени впервые высказал англичанин К.Стрейчи.)

#### Литература

1. Язык компьютера: Пер. с англ. М.: Мир, 1989.
2. Компьютер обретает разум: Пер. с англ. М.: Мир, 1990.
3. *Малыхина М.П., Частиков А.П.* Языки программирования: Лисп // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 3/89.
4. *McCarthy J.* Recursive Functions of Symbolic Expressions and Their Computation by Machine. Part 1. *Communications of the Associations for Computing Machinery*, 1960, vol. 3, no. 4.
5. *Хелмс Г.Л.* Языки программирования: Пер. с англ. М.: Радио и связь, 1985.
6. Лисп // Информатика, № 17/2000.
7. (Lots (of (Idiotic (Silly (Parentheses)))))) // Информатика, № 40/2001.
8. Толковый словарь по вычислительной технике (Microsoft Corporation): Пер. с англ. М.: Издательский отдел “Русская редакция” ТОО “Channel Trading Ltd.”, 1995.
9. *Частиков А.П.* От калькулятора до супер-ЭВМ // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 1/88.

...Программист должен обладать способностью первоклассного математика к абстракции и логическому мышлению в сочетании с эдисоновским талантом соорудить все, что угодно, из нуля и единицы. Он должен сочетать аккуратность бухгалтера с проницательностью разведчика, фантазию автора детективных романов с трезвой практичностью экономиста. А кроме того, программист должен иметь вкус к коллективной работе, понимать интересы пользователя и многое другое.

А.П. Ершов  
 “Андрей Петрович Ершов — известнейший ученый в области информатики, как и многие из его поколения, занялся вычислительным делом почти случайно” [1].



С Михаилом Шура-Бурой

Андрей Петрович Ершов (1931—1988) поступил на работу в Институт точной механики и вычислительной техники Академии наук (ИТМ и ВТ) в 1953 году, когда был еще студентом Московского государственного университета им. М.В. Ломоносова. В то время в ИТМ и ВТ формировался один из первых в нашей стране коллективов программистов [2]. Случайно год Ершов заканчивает университет, а в 1957 году его назначают заведующим отделом автоматизации программирования в созданном незадолго до этого Вычислительном центре Академии наук. Вскоре, в связи с образованием Сибирского отделения Академии наук, по просьбе директора Института математики данного отделения академика С.Л. Соболева он берет на себя обязанности организатора и фактического руководителя отдела программирования этого института и затем в 1960 году окончательно переезжает в Сибирь [2—4]. Позже отдел Ершова вошел в состав созданного в 1964 году Г.И. Марчуком Вычислительного центра Сибирского отделения Академии наук.

Благодаря Ершову новосибирский Академгородок становится одним из ведущих центров в области программирования, а сам он оказывает большое влияние на развитие информатики в стране. Понимая, как важна научно-организационная деятельность при становлении новых научных направлений, Ершов уделяет ей много внимания не только в масштабе страны. Он работает в редколлегиях международных изданий (Information Processing Letters, Theoretical Computer Science, Acta Informatica и других [2]), а также участвует в мероприятиях, проводимых Международной федерацией по обработке информации (International Federation for Information Processing — IFIP); является членом нескольких ее комитетов и рабочих групп, активным участником ряда конгрессов. [В 1980 году за плодотворную деятельность в IFIP Ершов награждают “Серебряным сердечником” (Silver Core) — высшим знаком отличия, принятым в этой организации.]

Для научной деятельности Андрея Петровича Ершова характерно сочетание теоретических исследований и практических работ. 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 начал с создания алгоритмов численных методов и стандартных программ. Первая работа Ершова была посвящена предложенному им методу обращения матриц.

В середине 1950-х годов шло 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. Под руководством Ершова разрабатывались одни из первых отечественных программирующих программ — для машин БЭСМ и “Стрела” (программирующей программой называлась в то время комбинация языка программирования и транслятора).

Позже под его руководством были созданы такие известные системы, как АЛЬФА (первая в мировой практике оптимизирующая система программирования для более сложных языков, чем Фортран), АЛЬФА-6 и БЕТА. Они во многом определили современную методологию оптимизирующей трансляции<sup>1</sup>.

Идейно к работам по автоматизации программирования примыкают и труды Ершова по теоретическому программированию, “Чьим объектом изучения являются математические абстракции программ” [5].

Ершов умел оценить текущее состояние науки и практики и наметить направления, которые будут определять развитие программирования. Он одним из первых в стране увидел те новые возможности взаимодействия с ЭВМ, которые способны дать системы разделения времени. В 1966 году Андрей Петрович организует работы, связанные с автоматическими информационными станциями (проект АИСТ). Проект АИСТ предусматривал проведение широкого круга исследований, касающихся архитектуры вычислительных комплексов, их программ-



Тандем с Э.Дейкстрой, сентябрь 1978 г.

ного обеспечения и моделирования вычислительных систем. В рамках данного проекта подготовлена первая в стране развитая система разделения времени АИСТ-0 (которая была реализована на многоаппаратном комплексе из отечественных ЭВМ).

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 позже будет назван *дружелюбностью к пользователю*, одним из первых в стране поставил задачу создания технологии программирования<sup>2</sup>.

Благодаря своему умению видеть новые перспективы в программировании, Ершов часто выступал как инициатор новых направлений исследований<sup>1</sup>.

<sup>1</sup> Оптимизирующий транслятор выполняет эквивалентные преобразования программы для получения более быстрой или более компактной программы.

<sup>2</sup> При этом сам русский термин “технология программирования” (*software engineering*) был предложен Ершовым.



ваний. Сюда относятся, например, методы использования естественных языков для взаимодействия с ЭВМ, автоматизация редакционно-издательской деятельности [2].

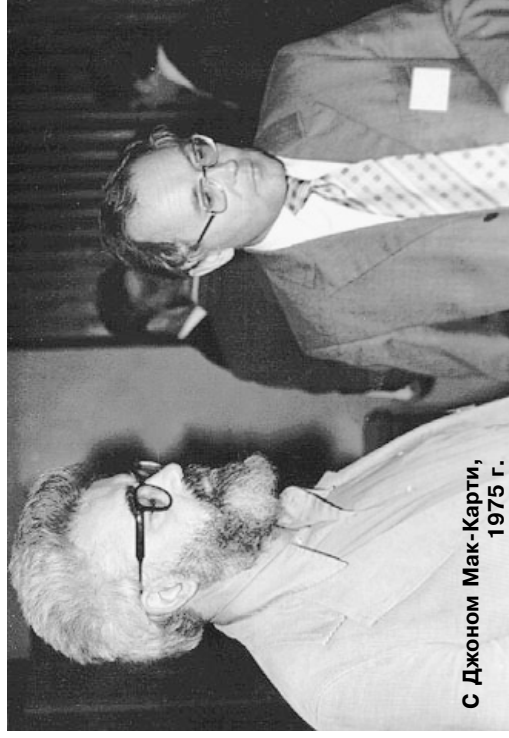
Большое внимание Ершов уделял подготовке кадров. В Новосибирском университете он организовал и затем бесценно руководил коллективом, занимающимся подготовкой студентов и аспирантов в области системного и теоретического программирования.

В последний период жизни Андрей Петрович Ершова привлекло обучение программированию и вообще информатике в школе. Он стал одним из создателей и признанным лидером так называемой “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. Концепции, состояние, перспективы” определены перспективы развития данного направления на годы вперед. Андрей Петрович являлся одним из авторов и редакторов первого школьного курса информатики, а также методического пособия к этому курсу. Незадолго до его кончины появился новый учебник информатики [6], одним из авторов и редактором которого он был.

Ершов готовил телевизионный курс по этой дисциплине, руководил созданием школьных систем программирования и школьного программного обеспечения вообще, вел большую организационную работу [2]. Главным делом последних лет его жизни являлось создание школьной энциклопедии по информатике.

## Литература

1. *Турский В.М.* Андрей Петрович Ершов. Очерки истории информатики в России. Избранные статьи // Информатика, № 29/99.
2. *Полтосин И.В.* Андрей Петрович Ершов: жизнь и творчество // Ершов А.П. Избранные труды. Новосибирск: Наука, 1994.
3. *Ковалевский И.* Дело жизни — информатика // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 10/89.
4. Лидер отечественного программирования // Информатика, № 16/2001.
5. *Ершов А.П.* Введение в теоретическое программирование (беседы о методе). М.: Наука, 1977.
6. *Ершов А.П., Кушинченко А.Г., Лебедев Г.В. и др.* Основы информатики и вычислительной техники // Под ред. А.П. Ершова. М.: Просвещение, 1988.



С Джоном Мак-Карти, 1975 г.







# П р и д у м а в ш и й Ч е р е п а ш к у

*“Лого — инструмент для познания и развития собственного мышления, и в этом отличие этой среды от систем программирования, ориентированных в первую очередь на обеспечение более эффективного использования аппаратуры” [1].*

Сеймур Пейперт пытался изучать работу мозга и однажды заметил, что “всегда рассматривал процесс обучения как хобби”. Следуя этому “хобби”, он осваивал такие разные виды деятельности, как пилотирование самолетов, чтение на китайском языке, “ремесло” иллюзиониста. Думая о процессах мышления, Пейперт, подобно классическим рассеянными профессорами, иногда как бы терял связь с окружающим миром: однажды, пролетев полпути над Атлантикой, он вдруг вспомнил, что оставил свою жену в аэропорту Нью-Йорка [2].

Пейперт родился в Южной Африке. Спустя какое-то время после окончания высшего учебного заведения, он получил ученую степень доктора математики, а затем провел пять лет в Женеве, работая с выдающимся швейцарским физиологом Жаном Пиаже. Теория Пиаже, согласно которой ребенок учится в процессе игры с окружающими его предметами, оказала на Пейперта сильное влияние. Вспоминая о своем детском увлечении автомобилями, Пейперт, например, сделал вывод, что именно интерес к ним открыл ему связь “объект — мысль” и облегчил в дальнейшем постижение математических абстракций.

Решив, что объектом такого рода способен стать компьютер, Сеймур Пейперт перешел в 1964 году в Лабораторию искусственного интеллекта Массачусетского технологического института (МТИ), где занялся этим новым направлением исследований [2, 3]. Пейперт был убежден, что учащиеся начальной школы и даже дошкольники могут на-

учиться программировать. “Освоение языков — одно из увлекательнейших занятий для ребенка, — говорил он. — Каждый нормальный ребенок учится разговаривать. Почему бы ему не научиться “разговаривать” с компьютером?” При этом Пейперт считал, что даже Бейсик, простейший компьютерный язык того времени, слишком абстрактен для ребенка.

Вскоре Пейперт и его коллеги создали новый язык на основе Лиспа, назвав его Лого (что по-гречески означает *слово*). По инициативе Пейперта в языке стала использоваться так называемая черепашка, обеспечивающая связь “объект — мысль” (сначала употреблялась механическая черепашка, ползающая по полу, а затем — ее условное изображение на экране). Как отмечал Пейперт, “компьютер обычно шаг за шагом ведет ребенка за собой”, а Лого, наоборот, “убеждает ребенка в том, что он способен управлять машиной, позволяет ребенку сказать: “Здесь я хозяин”. Сначала Лого предназначался для больших компьютеров и мини-компьютеров, имеющихся в МТИ, но потом проник в тысячи классов начальных школ.

Таким образом, Лого — язык со встроенной черепашкой [2, 4], представляющей собой треугольный значок на экране компьютера, который возникает и движется в соответствии с командами, вводимыми с клавиатуры. Сперва новичок дает черепашке простые команды, наподобие **ВПЕРЕД 100**, что означает “переместиться вперед на 100 шагов”, или **ВЛЕВО 60**, т.е. “сделать поворот влево на 60 градусов”. Такие команды можно использовать для составления небольших программ или процедур, формирующих геометрические фигуры — окружности, квадраты, треугольники, и эти процедуры, в свою очередь, можно объединять в более сложные процедуры и соответственно фигуры.

Подобно своему “родителю” Лиспу, Лого относится к расширяемому языку. Помимо встроенных операций (**ВПЕРЕД**, **НАЗАД**, **ВПРАВО**, **ВЛЕВО**), Лого позволяет программисту определять

новые процедуры. Слово для указывает, что вы собираетесь определить процедуру: следующее слово — это имя процедуры, а слово **КОНЕЦ** указывает на завершение определения. После того как процедура определена, к ней можно обращаться. Надо набрать имя процедуры, и компьютер выполнит включенные в нее команды, например:

```
ДЛЯ КВАДРАТ
ПОВТОР 4 [ВПЕРЕД 100 ВПРАВО 90]
КОНЕЦ
?КВАДРАТ
```

Черепашка пройдет вперед 100 шагов и сделает поворот вправо на 90 градусов (так повторится четыре раза, пока не будет построен квадрат).

Другая особенность Лого, делающая его простым для обучения, — отсутствие необходимости указывать заранее типы переменных, как это нужно делать, например, в Паскале. В Лого для обозначения переменных просто используется двоеточие.

```
ДЛЯ ТРЕУГОЛЬНИК :РАЗМЕР
ПОВТОР 3 [ВПЕРЕД :РАЗМЕР ВПРАВО 120]
КОНЕЦ
?ТРЕУГОЛЬНИК 80
```

Эта процедура формирует равнобедренный треугольник, трижды повторяя команду, указанную в квадратных скобках. При ее выполнении переменной **:РАЗМЕР**, задающей длину стороны треугольника, присваивается значение 80 шагов.

Во многом благодаря Сеймуру Пейперту сегодня практически любой дошкольник в процессе игры способен освоить программирование, которое несколькими десятилетиями раньше было доступно лишь небольшому количеству специалистов. И это лишь одна сторона дела...

#### Литература

1. Юдина А.Г. Лого — среда для развития мышления // Информатика, № 33/95.
2. Язык компьютера: Пер. с англ. М.: Мир, 1989.
3. Компьютер обретает разум: Пер. с англ. М.: Мир, 1990.
4. Зайцев А.А. Повесть о настоящей черепашке // Информатика, № 3/2001.

# Н и к л а у с      В и р т

## И    я    з    ы    к    и

В середине 1960-х годов Международная федерация по обработке информации — IFIP (*International Federation for Information Processing*) предложила нескольким специалистам в области информатики принять участие в создании нового языка программирования — преемника Алгола-60. Среди них был швейцарский ученый Никлаус Вирт, который преподавал в то время в Станфордском университете. Вирт участвовал в создании проекта языка, названного Алгол-W (по его фамилии — Wirth). И хотя IFIP данный проект отклонила (был принят проект языка Алгол-68), Вирт, возвратившись в Швейцарию, продолжил работу. Он хотел получить инструмент для обучения студентов программированию: в этом смысле Вирта не устраивал не только новый Алгол, но и все “ныне используемые основные языки программирования, свойства и конструкции которых зачастую нельзя объяснить логически и убедительно и которые нередко просто оскорбляют здравый смысл” [1]. Вскоре Никлаус Вирт и его сотрудники из Федерального технического университета в Цюрихе подготовили первую версию языка, получившего название Паскаль [в честь знаменитого француза — математика, физика, философа и писателя Блеза Паскаля (1623—1662), построившего одну из первых механических счетных машин], затем — первый вариант компилятора, а в 1971 году вышло описание языка. В 1974 году появилась новая версия Паскаля, и спустя еще какое-то время было опубликовано руководство для пользователей, которое, по словам Вирта, вполне можно рассматривать как “стандартное определение” языка Паскаль [2].

Еще подростком Вирт увлекался конструированием радиоуправляемых моделей самолетов, причем позже он получил в Калифорнийском университете степень бакалавра в области электротехники. К разработке языка Вирт подошел так, как подошел бы инженер к конструированию машины. А “искусство инженера, — говорил он, — состоит в том, чтобы делать сложные задачи простыми” [1].

Язык Паскаль требует от программиста описания всех переменных в отдельном разделе, помещаемом в начале программы. Наличие такого раздела ограничивает свободу программиста. Но одновременно Паскаль требует



строгости программирования, который способствует уменьшению количества ошибок. Как и всякий преемник, Паскаль имеет многие черты своего предшественника; например, Паскаль, как и Алгол-60, предусматривает блочную структуру программ (программы состояются из блоков, которые начинаются и оканчиваются словами BEGIN и END соответственно). Структура программы на Паскале позволяет даже специалисту, не являющемуся ее автором, обнаруживать и исправлять допущенные ошибки и вносить изменения. Это делает Паскаль подходящим языком для создания больших программ. В то же время часто говорят, что Паскаль особенно удобен для изучения теории и практики программирования, но не для реальных применений (и ничего удивительного здесь нет: основная цель, которую преследовал Вирт, создавая Паскаль, — разработать язык всего лишь для обучения программированию).

Успех Паскаля превзошел все ожидания [1, 3, 4]. Возможно, главной причиной популярности этого языка было то, что он способствовал развитию зарождающегося тогда движения за так называемое структурное программирование, которое потом очень быстро обрело силу.

Паскаль стал прародителем нескольких языков программирования [1, 2, 5]. Примерно через 10 лет после него Вирт создал язык Модула-2 (Modula 2, от *MODular Language* — модульный язык), который основан на многих концепциях, используемых в Паскале, причем наиболее примечательным тут стало введение средств построения больших и сложных программ, которые можно было теперь создавать в виде отдельных самостоятельных модулей, а затем собирать их вместе [1, 6].

“Можно считать, что история языка Модула-2 начинается в 1980 году, когда Никлаус Вирт, известный боль-

шинству специалистов как создатель языка Паскаль, опубликовал описание нового языка программирования, названного им Модула.

В отличие от Паскаля, который был по замыслу языком для обучения программированию, Модула с самого начала представляла собой язык для профессиональных системных программистов, продолжая лучшие традиции своего предшественника и обогащая их новыми идеями, соответствующими таким современным требованиям к языкам программирования, как структурность, модульность и способность к расширению.

Как и множество других языков программирования, Модула подверглась эволюции, во время которой ее первоначальное название было переделано в имя Модула-2. Базовым описанием языка считается описание, представленное в третьем издании книги Вирта “Программирование на языке Модула-2”, то есть в [7]. (Его развитием является язык Модула-3, разработанный совместно фирмами DEC и Olivetti, а Вирт создал еще язык Oberon.)

На Паскаль опирались и все четыре языка, отобранные военным ведомством США в конце 1970-х годов после того, как был объявлен конкурс, в результате которого появился универсальный язык высокого уровня, получивший в дальнейшем название “Ада” (в честь первого программиста — Августы Ады Лавлейс). Следует также отметить, что среди расширенных и диалектов Паскаля особую популярность получило семейство систем программирования фирмы Borland для персональных компьютеров, получившее название “Турбо Паскаль”.

### Литература

1. Язык компьютера: Пер. с англ. М.: Мир, 1989.
2. Малыгина М.П., Частиков А.П. Языки программирования: Паскаль // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 6/90.
3. Язык Паскаль // Информатика, № 34/2000.
4. Преемник Алгола // Информатика, № 41/2001.
5. Пройдаков Э.М., Теплицкий Л.А. Англо-русский толковый словарь по вычислительной технике, Интернету и программированию. Изд. 2-е, испр. и доп. М.: Издательско-торговый дом “Русская редакция”, 2000.
6. Белецкий Я. ТопСпид: Расширенная версия языка Модула-2 для персональных компьютеров IBM: Пер. с польского. М.: Машиностроение, 1993.
7. Wirth, N. Programming in Modula 2. Third edition. Springer Verlag, Berlin, 1985.

# Н а ч а л о

В начале 1970-х годов появились первые микрокомпьютеры, а в январском номере американского журнала Popular Electronics за 1975 год была помещена первая часть статьи, в которой описывался набор для сборки "первого в мире мини-компьютера, способного конкурировать с коммерческими моделями" [1]. Это был компьютер Альтаир-8800 на основе микропроцессора Intel-8080. В течение нескольких следующих недель компания-производитель MITS (*Micro Instrumentation and Telemetry Systems* — микроприборы и телеметрические системы), располагавшаяся в штате Нью-Мексико, получила за свой набор сотни чеков на сумму 397 долларов каждый.

[Новая машина (которую владелец набора должен был сначала еще собрать) обладала весьма ограниченными возможностями. Оперативная память Альтаира имела объем всего 256 байт, экран отсутствовал, и покупателям приходилось снабжать машину блоками расширения памяти, монитором и т.д.]

Кроме того, компьютер выпускался без программного обеспечения, и если его владелец хотел, чтобы он делал что-то полезное, нужно было сначала подготовить соответствующую программу (причем в двоичном коде).]

В то время в компьютерной фирме Honeywell, располагавшейся недалеко от Бостона, работал молодой программист Пол Аллен. Узнав о новинке, он отправился к своему другу, первокурснику Уильяму (Биллу) Гейтсу. Обсудив ситуацию, молодые



Альтаир-8800

люди решили создать для новой машины программное обеспечение. О следующем их шаге, наверное, всегда будут упоминать в рассказах, связанных с историей компьютерной техники. Гейтс и Аллен позвонили в компанию MITS и сообщили ее владельцу Эду Робертсу, что подготовили программу, способную переводить операторы языка Бейсик в машинный код для Альтаира, причем готовы продать ее Робертсу. Тот, не задумываясь, ответил, что купит интерпретатор Бейсика, способный работать на Альтаире.

Гейтс и Аллен быстро взялись за дело. Надо сказать, что будучи еще учениками средней школы в Сиэтле они вступили в сообщество хакеров. Их школьный клуб арендовал компьютер (покупал машинное время), и Гейтс самостоятельно научился программировать, когда ему было всего 13 лет. Друзья даже основали компанию Traf-o-Data и пытались продавать результаты своих компьютерных исследований транспортных потоков в небольших городах, расположенных близ Сиэтла. Особого успеха эти попытки не имели, однако слух о способностях двух молодых людей широко распространился, и им предложили написать программу для компаний, обслуживающих электросети северо-восточной части США и Канады. "Никто не знал, что тогда мы были еще школьниками", — вспоминал позднее Гейтс.

Приобретенный опыт весьма пригодился при разработке программного обеспечения для компьютера Альтаир. Причем нужно отметить, что Гейтс и Аллен его даже не видели. Все, чем они располагали, — это руководство, касающееся системы команд микропроцессора Intel-8080, на базе которого был

построен Альтаир, и сведения о характеристиках машины. Для отладки же использовался большой компьютер.

Когда первый вариант интерпретатора был получен, друзья подготовили соответствующим образом перфоленту и Аллен полетел на самолете в город, где находилась компания MITS, чтобы показать программу Робертсу. К тому времени Робертс оборудовал одну из своих машин дополнительной платой памяти, телетайпом и устройством для чтения перфолент. Во время полета Аллен сообразил, что они с Гейтсом забыли написать программу загрузки, без которой их интерпретатор просто не смог бы "загрузиться" в память. Пришлось составлять ее прямо в самолете.

На следующий день Аллен внес последние изменения, ввел интерпретатор Бейсика в машину и запустил программу на выполнение. Он понимал, что если в программе есть хоть одна ошибка, то работать она не будет. Однако телетайп ожил, и на бумаге появился текст: "РАЗМЕР ПАМЯТИ?"

Программа работала. В ответ Аллен напечатал "7K". Затем он задал компьютеру простейшую задачу: "PRINT 2 + 2". Машина сразу ответила: "4". Потом разработчики интерпретатора отмечали, что Робертс и его коллеги из фирмы MITS "были крайне взволнованы, поскольку раньше никто из них не видел, чтобы эта машина что-нибудь делала". Робертс купил программу (на условиях отчисления авторского гонорара с каждого проданного экземпляра), а Аллен и Гейтс вскоре во второй раз в жизни организовали компанию, назвав ее теперь Microsoft [1—3]. Они продолжали совершенствовать интерпретатор для Альтаира, а кроме того, написали ряд других программ для малых машин. "Вообще-то это была ерунда, — позднее говорил Гейтс, — но пользователи были довольны" [1].

В течение долгого времени фирму Microsoft возглавлял Гейтс. Затем, с 1998 года, некоторое время ее главой был Аллен. Сегодня председатель правления (совета директоров) и главный архитектор программного обеспечения корпорации Microsoft Билл Гейтс является одним из самых богатых людей в мире (и самым богатым среди тех, кто получил свое состояние не по наследству).

#### Литература

1. Язык компьютера: Пер. с англ. М.: Мир, 1989.
2. Знакомьтесь: компьютер: Пер. с англ. М.: Мир, 1989.
3. Малыгина М.П., Частиков А.П. Языки программирования: Бейсик // Новое в жизни, науке, технике. Сер. "Вычислительная техника и ее применение", № 1/90.



Молодые программисты Пол Аллен и Билл Гейтс (в нижнем ряду крайние справа и слева соответственно) написали интерпретатор Бейсика для первого в мире ПК Altair компании MITS.

# Д Е Н Н И С Р И Т Ч И

*Язык С — это инструмент, острый как бритва: с его помощью можно создать и элегантную программу, и кровавое месиво.*

Брайан Керниган

Одним из самых популярных языков программирования является язык С, который был создан как инструмент написания операционной системы UNIX в научно-исследовательской фирме Bell Telephone Laboratories американской корпорации AT&T (*American Telephone and Telegraph*). В 1969 году Кен Томпсон, считавшийся среди коллег лучшим программистом в мире, начал разрабатывать эту систему для мини-компьютера PDP-7 фирмы DEC. А в 1972 году Деннис Ритчи, специалист по системному программированию, который получил степень бакалавра по прикладной математике в Гарвардском университете, активно помогавший разрабатывать UNIX, представил язык С, где сочетались лучшие свойства ассемблера и языков высокого уровня. От ассемблера были взяты гибкие и эффективные средства работы с памятью, а от языков высокого уровня — широкий набор управляющих конструкций, возможность использования сложных структур данных, средств ввода и вывода [1]. Ритчи считал, что новый язык пригодится для работы. И он не ошибся.

Для повышения скорости функционирования операционных систем при их создании обычно применялся язык низкого уровня — ассемблер, а язык С настолько хорошо себя зарекомендовал, что на нем была написана почти вся операционная система UNIX [2—4].

Вообще это один из самых универсальных языков программирования. Его можно применять также для создания драйверов внешних устройств, трансляторов языков программирования, текстовых процессоров, программ для решения математических, экономических и многих других прикладных задач [5].

Название языка соответствует третьей букве английского алфавита. Один из языков, появившийся в 1960-х годах, назывался APL (*A Programming Language*). Тем самым как бы была занята первая буква ал-



фавита — А (Эй). Во время работы над операционной системой UNIX Томпсоном был создан язык программирования В (Би), “для внутреннего пользования”, который оказал сильное влияние на следующий язык, разработанный Ритчи для тех же целей. [Язык В частично основывался на созданном в Кембриджском университете (шт. Массачусетс) языке BCPL, от *Basic Combined Programming Language* — базовый комбинированный язык программирования.] Подчеркивая эту преемственность, языку дали название С (Си).

Первое описание языка было приведено в книге [2]. “Керниган подготовил почти весь пояснительный материал, я же отвечал за приложение, куда было включено справочное руководство по языку, и за главу о взаимодействии с UNIX”, — сообщает Ритчи. Долгое время данное описание считалось стандартом, однако некоторые его места допускали неоднозначное толкование, что породило множество трактовок языка С. Для исправления ситуации при Американском национальном институте стандартов (*American National Standards Institute* — ANSI) был образован комитет по стандартизации языка, и в 1983 году был утвержден стандарт, получивший название ANSI C [5—7]. “Прямыми потомками” языка С являются языки [8]: С++ (Б.Страуструп, 1984 г.); Concurrent C (Н.Джихани, 1986 г.); Objective C (Б.Кокс, 1986 г.); С\* (Thinking Machines, 1987 г.); С+@, ранее Calico (Bell Labs, 1991). Самым известным здесь стал, безусловно, С++.

После реорганизации AT&T, уже в рамках Lucent Labs, Ритчи возглавлял группу, разрабатывавшую операционную систему Plan 9 (1995 г.). Она

явилась основой следующего проекта его группы — операционной системы Inferno, о создании которой было объявлено в апреле 1996 года. Для этой системы Ритчи подготовил язык Limbo [8, 9].

Сейчас Ритчи (которому в сентябре 2001 года исполнилось 60 лет), имеющий степень бакалавра по физике и магистра по прикладной математике, возглавляет отдел исследований в области системного программного обеспечения в фирме Bell Laboratories.

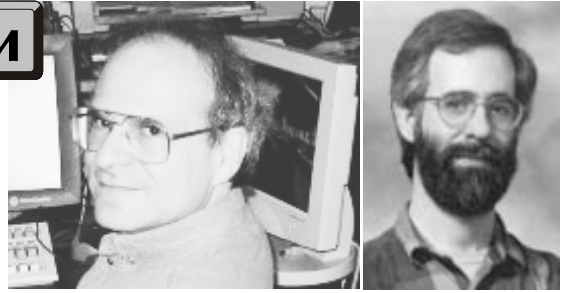
В 1983 году Деннису Ритчи и Кену Томпсону за разработку и реализацию языка программирования С и операционной системы UNIX была вручена премия Тьюринга. В 1988 году Ритчи был избран в Американскую национальную инженерную академию (National Academy of Engineering).

Почему же язык С покорило сердца миллионов программистов? “Главные достоинства С состоят в том, что он предоставляет программисту возможность полного управления реализацией и что программы, написанные на С, работают весьма быстро”, — считает Брайан Керниган. А вот ответ самого Денниса Ритчи [8]: “Язык С — это невероятный, оглушительный и огромный успех. Хотя повороты судьбы ему помогали, он, безусловно, удовлетворил потребности в таком языке реализации систем, который был достаточно эффективным, чтобы заменить ассемблер, и в то же время достаточно абстрактным и гибким, чтобы описывать алгоритмы и взаимодействия в широком спектре программных сред”. При этом “существует масса прекрасных языков (прекраснее, чем С), которые не привились, — отмечает Ритчи, — но кто-то все же выигрывает...”

## Литература

1. Дукаревич Г.Б. Введение в программирование на языке Си // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 4/91.
2. Kernighan B.W. and Ritchie D.M. The C Programming Language. Prentice-Hall, 1978. [Керниган Б., Ритчи Д., Фьюер А. Язык программирования Си. Задачи по языку Си: Пер. с англ. М.: Финансы и статистика, 1985.]
3. Язык компьютера: Пер. с англ. М.: Мир, 1989.
4. Болски М.И. Язык программирования Си. Справочник: Пер. с англ. М.: Радио и связь, 1988.
5. Стариков Ю.А. Мобильность программ и особенности реализации языка Си. Изд. 2-е, доп. М.: МП “Память”, 1992.
6. Эй, Би и, наконец, Си // Информатика, № 5/2000.
7. Юбилей // Информатика, № 34/2001.
8. Богатырев Р. Летопись языков. Си // Мир ПК, № 8/2001.
9. Мосени П. Inferno: ОС для сетевых компьютеров // Мир ПК, № 10/1997.

# С О З Д А Т Е Л И В Е Л И К И Х П Р О Г Р А М М



В 1978 г. Дэн Бриклин (слева) и Боб Фрэнкстон создали первую электронную таблицу — VisiCalc.

“Моменты рождения великих программ, как и литературных шедевров, никому не известны” [1].

В середине 1970-х годов инженер-программист Дэниел Бриклин посещал курсы в Гарвардской школе бизнеса. Он работал в фирме DEC (*Digital Equipment Corporation*), создавшей популярную серию компьютеров PDP. Здесь Бриклин участвовал в подготовке системы обработки текстов, а потом перешел в другую компанию. Опыт работы в фирмах показал ему, что, разбираясь в технологии, он недостаточно хорошо разбирается в бизнесе. И Бриклин отправился на курсы в Гарвард.

Занимаясь утомительными вычислениями, связанными с определением изменения суммы налога, Бриклин пришел к выводу, что есть другой, более эффективный способ выполнения подобных расчетов. Главная проблема здесь заключалась в том, что при изменении какой-то одной цифры приходилось пересчитывать все связанные с ней величины. Электронный калькулятор, разумеется, облегчал работу, но все равно расчеты подобного рода отнимали слишком много времени. При этом малейшая ошибка могла все испортить.

Бриклин решил поручить такие вычисления компьютеру. Свою идею он обсудил с другим программистом, Робертом Фрэнкстоном, который проявил к ней интерес. В конце 1978 года Фрэнкстон начал готовить программу, и уже через несколько месяцев работа была завершена. Фрэнкстон и Бриклин дали разработанной системе название VisiCalc (*Visible Calculator*) [1–6]. Она формировала на экране персонального компьютера столбцы чисел, которые при изменении каких-либо данных в этой электронной таблице мгновенно изменялись соответствующим образом.

В течение первого года после начала продаж системы VisiCalc сбыт персональных компьютеров — это были машины Apple II — резко возрос, причем было продано около 100 тыс. экземпляров программы (экземпляр которой стоил 200 долларов).

Спустя три года после нее появилась созданная Митчелом Кэпором программа Lotus 1-2-3, которая сразу же продемонстрировала свои неоспоримые преимущества. Новый табличный процессор соединял в себе лучшие качества системы VisiCalc с графическими возможностями и обладал удобными средствами поиска информации.

Митчел Кэпор, как и многие другие молодые люди, сделавшие свои состояния в первые годы широкого распространения персональных компьютеров, не был похож на потенциального миллионера. Кэпор имел явные способности к математике, но в Йельском университете, куда он поступил после окончания школы (и где специализировался в области математики и лингвистики), его оценки были достаточно скромны. Возможно, это объяснялось тем, что Кэпор слишком много времени проводил на университетской радиостанции, где возглавил спустя какое-то время музыкальный отдел. К тому же он увлекся трансцендентальной медитацией\* и даже получил право на ее преподавание [1].

Закончив в 1971 году университет, Кэпор два года работал диск-жокеем, после чего переехал в Кембридж (шт. Массачусетс), где стал пробовать свои силы уже в программировании. Однако данное занятие казалось Кэпору скучным, и он переключился на преподавание трансцендентальной медитации.

В 1978 году, увлекшись персональными компьютерами, Кэпор стал продавать усовершенствованную им стереосистему для машины Apple II. А вскоре он вновь занялся программированием и создал две хорошие программы для деловых операций. Покупателей для своих программ он находил с помощью фирмы, сыгравшей существенную роль в успехе системы VisiCalc. Прошло немного времени, и доходы Кэпора стали выражаться шестизначным числом. Однако Кэпор на этом не успокоился: продав свои авторские права на програм-

\* Трансцендентальная медитация — особый вид отрешенности от окружающего мира, практикуемый в ряде восточных религиозно-философских систем.

мы больше чем за миллион долларов, он основал собственную компанию Lotus Development Corporation с целью разработки и продажи новой системы — Lotus 1-2-3.

Во второй половине 1981 года фирма IBM приступила к выпуску персональных компьютеров, а при разработке табличного процессора Lotus 1-2-3 Кэпор ориентировался именно на IBM PC. Он был убежден, что это должно дать его программе преимущество в конкурентной борьбе.

Расчет Кэпора полностью оправдался. О появлении системы Lotus 1-2-3 было объявлено в конце 1982 года, причем на ее рекламу затратили около миллиона долларов. Расходы окупились очень скоро. Система заняла первую строку в списке самых популярных программ, и через полгода было куплено 60 тыс. ее экземпляров, каждый из которых стоил 495 долларов.

Так Дэниел Бриклин, Роберт Фрэнкстон, а затем Митчел Кэпор дополнили ряды тех, кто нажил состояния на программном обеспечении.

Программа Lotus 1-2-3 во многом превзошла система Microsoft MultiPlan для машин Apple, IBM PC и других. Однако у программы Lotus 1-2-3 имелось одно очень важное преимущество: она действовала в несколько раз быстрее. При этом уже, конечно, существовали и другие табличные процессоры [7, 8].

А в 1985 году появилась система Excel...

## Литература

1. Язык компьютера: Пер. с англ. М.: Мир, 1989.
2. Кох О. Excel 5.0: Пер. с нем. СПб.: BHV — Санкт-Петербург, 1994.
3. Сенокосов А.И. К XX годовщине переворота // Информатика, № 16/99.
4. Экран в клеточку // Информатика, № 46/2000.
5. Предшественники // Информатика, № 19/2001.
6. Excel: путь к успеху // Информатика, № 43/2001.
7. Фигурнов В.Э. IBM PC для пользователя. Изд. 6-е, перераб. и доп. М.: ИНФРА-М, 1995.
8. Кренкель Т.Э., Коган А.Г., Тараторин А.М. Персональные ЭВМ в инженерной практике. М.: Радио и связь, 1989.

# Р Э Т Л И Ф Ф И Д Р У Г И Е

**В** 1978 году появился текстовый редактор WordStar, примерно через год начала использоваться первая в мире электронная таблица VisiCalc, а затем наступила очередь системы управления базами данных dBase II. Вот “три кита, на спинах которых персональный компьютер стремительно выплыл на пик мирового прогресса и стал его символом... Именно эти программные средства убедили простых тружеников умственного труда не просто в пользе персонального компьютера, но в его необходимости” [1].

“Система управления базами данных должна выполнять три основные функции” [2], которые, впрочем, “типичны и для других форм хранения данных.

1. *Ввод данных.* В системе должна существовать структура, где могут накапливаться данные. Кроме того, в системе необходимо предусмотреть возможность просмотра этих данных и внесения в них изменений...

2. *Запросы по данным...*

3. *Составление отчетов.* Отчет отличается от запроса двумя вещами. Во-первых, отчет обычно охватывает не какую-либо часть базы данных, а всю ее целиком. Во-вторых, при получении отчета информация, как правило, предварительно обрабатывается”.

Представление о том, что такое база данных, дает, например, обычная записная книжка с фамилиями, именами, адресами и телефонами. К такой “базе данных” обращаются, чтобы получить те или иные сведения, чаще всего нужный номер телефона или адрес. В отличие от записной книжки, компьютер способен хранить огромное количество записей, а главное преимущество, которое дает система управления базами данных, заключается в возможности быстрого поиска информации и представления ее в удобной форме. При этом поиск данных в компьютерной “картотеке” может осуществляться по любой совокупности признаков, характеризующих представленные в ней объекты [2, 3]. К примеру, система для учета студентов высшего учебного заведения должна позволять быстро получить список студентов, которым меньше восемнадцати лет, или тех, кто имеет академическую задолженность.

Создателем системы dBase II был инженер из НАСА (Национальный

комитет по авионавигации и исследованию космического пространства, находящийся в США) Уэйн Рэтлифф [4, 5]. Первый вариант программы Рэтлифф подготовил в свободное от основной работы время и, дав ему название “Вулкан”, стал пытаться продавать, однако особого успеха эти попытки не имели. Вскоре программой заинтересовался Джордж Тейт, тогда уже занимавшийся программным бизнесом.

Джордж Тейт (позже — один из основателей фирмы Ashton Tate) служил в армии, а затем работал мастером по ремонту радиоаппаратуры. В середине 1970-х годов в числе других энтузиастов он приобрел набор для сборки первого “коммерчески распространяемого” компьютера Альтаир-8800, построенного на основе микропроцессора Intel 8080. Сборка компьютера, несмотря на его очень ограниченные возможности [оперативная память Альтаира имела емкость 256 байт (несколько строк текста), экран отсутствовал, и владельцам приходилось обеспечивать машину блоками расширения памяти, монитором и т.д.], была непростым делом. Однако и произведя сборку, купивший набор человек по-прежнему мог рассчитывать лишь на свои собственные силы, так как компьютер выпускался без программного обеспечения, а приобрести его было негде. Если владелец компьютера хотел, чтобы машина делала что-то полезное, ему нужно было сначала создать соответствующую программу (причем используя двоичный код).

Собирая машину, Джордж Тейт понял, что его знания о ней весьма скудны, и решил “изучать и постигать компьютер”. Вскоре Тейт стал хорошим специалистом в этой области и мог подрабатывать ремонтом компьютеров. Через некоторое время он стал работать менеджером по сбыту в компании, производившей компьютерные терминалы, а в 1980 году вместе со своим приятелем Хэлом Лашли начал заниматься продажей программного обеспечения.

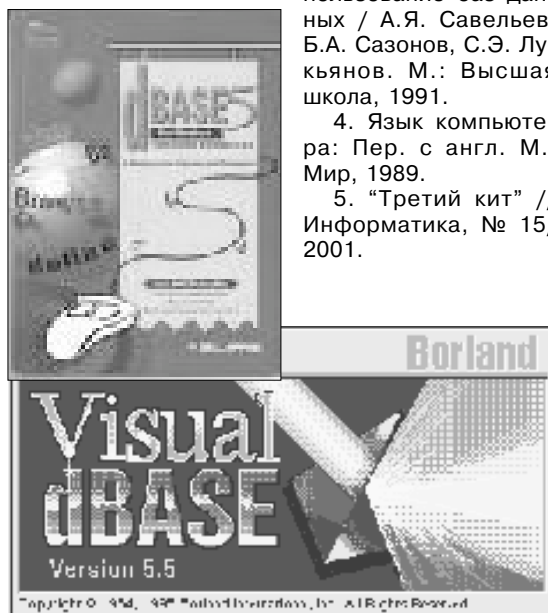
Узнав о существовании малоизвестной сис-

темы “Вулкан”, Тейт и Лашли заключили с Рэтлиффом контракт, предусматривающий их монополию на распространение этой программы. К тому времени на название “Вулкан” заявила свои права одна из фирм, а потому партнеры решили дать программе другое “имя”. Их рекламный агент предложил название dBase II, которое, как он считал, звучало весьма неплохо с технической точки зрения. Кроме того, оно как бы указывало, что это новая, улучшенная версия своего предшественника — системы dBase.

На самом деле никакого предшественника не существовало, однако система dBase II была улучшенной в другом смысле: по ряду показателей она значительно превосходила другие программы, ориентированные на решение соответствующего класса задач [2]. Скоро эта система стала “бестселлером”, а Рэтлифф, Лашли и Тейт почти так же быстро пополнили ряды миллионеров, наживших состояния на программном обеспечении [4].

## Литература

1. *Сенокосов А.И.* К XX годовщине переворота // Информатика, № 16/99.
2. *Крамм Р.* Системы управления базами данных dBase II и dBase III для персональных компьютеров: Пер. с англ. М.: Финансы и статистика, 1988.
3. Персональный компьютер для всех: В 4 кн. Кн. 3. Создание и использование баз данных / А.Я. Савельев, Б.А. Сазонов, С.Э. Лукьянов. М.: Высшая школа, 1991.
4. Язык компьютера: Пер. с англ. М.: Мир, 1989.
5. “Третий кит” // Информатика, № 15/2001.



# С а к с о ф о н и с т

“Среди расширений и диалектов Паскаля особую популярность получило семейство систем программирования фирмы Borland для персональных компьютеров, получившее название Турбо Паскаль” [1].

“Система программирования Турбо Паскаль представляет собой единство двух в известной степени самостоятельных начал — компилятора с языка программирования Паскаль и некоторой инструментальной программной оболочки, способствующей повышению эффективности создания программ...”

Разумеется, не только и даже не столько среда (т.е. средства, предоставляемые программной оболочкой. — Прим. ред.), пусть даже весьма совершенная, послужила причиной широкой популярности Турбо Паскаля. Не следует забывать, что в его основе лежит мощный язык программирования, представляющий собой значительно расширенную версию языка Паскаль. Задуманный первоначально автором Н.Виртом как средство обучения программированию, этот язык в интерпретации фирмы Borland приобрел множество дополнительных свойств, позволяющих говорить о нем как о вполне современном универсальном языке программирования” [2].

Создатель Турбо Паскаля Филип Кан родился в Париже (его отец, уроженец Германии, был инженером, а мать-француженка — кинопродюсером) [3]. Кан некоторое время учился в Цюрихе, где прослушал вводный курс Никлауса Вирта, посвященный Паскалю [1, 3—7]. Вернувшись во Францию, он защитил диссертацию по математике. В этот период он иногда подрабатывал, играя на саксофоне в ночных клубах.

Кан занимался математикой, а программированием интересовался только потому, что упражнения в данной области в какой-то степени помогали ему решать задачи и доказывать теоремы. Затем он приобрел свой первый микрокомпьютер Apple II, и стал вместе с двумя приятелями писать программы на Паскале и продавать их. Кану не нравились существующие компиляторы для Паскаля, и он решил создать Турбо Паскаль. В 1982 году Филип Кан, взяв эту уже почти готовую про-

грамму и 2000 долларов, отправился в Калифорнию [3].

Кан получил туристскую визу и не имел права на так называемую “зеленую карточку” — правительственное разрешение на получение работы в США. Однако он решил сделать ставку на Турбо Паскаль и стал организовывать фирму, которая должна была разрабатывать программное обеспечение. Кан дал ей название Borland International [2—4], подсказанное телевизионной передачей о космонавте с похожим именем — Франке Бормане (ставшем председателем правления Восточных авиалиний). Кан решил, что имя, имеющее “всеамериканское звучание”, послужит подходящим прикрытием для чужака в мире бизнеса.

Однако, несмотря на громкое название компании, ему не удалось заинтересовать ни одну корпорацию из тех, что вкладывали деньги в различные проекты. Тогда Кан собрал небольшую сумму денег у родственников и весной 1984 года организовал офис в двух комнатах, располагавшихся над гаражом. Затем он уговорил сотрудников популярного компьютерного журнала дать рекламу Турбо Паскаля в кредит.

Кан понимал, что в его ситуации от рекламы зависит очень многое, если не все. Прочитав книгу о продаже товаров по почте, где рекомендовалось употреблять яркие цвета для привлечения внимания потенциальных покупателей, Кан при оформлении рекламного объявления использовал резкие зеленые, синие и розовые тона. В книге также говорилось, что для быстрого успеха стоимость товара должна быть меньше 50 долларов. Поэтому он оценил свой сложный компилятор и редактор программ в 49 долларов 95 центов. Другие системы стоили примерно в 10 раз больше.

Реклама сыграла решающую роль в завоевании рынка. Только за первый месяц Кан получил заказов на 150 тыс. долларов. Потом стало помогать и то, что в компьютерных изданиях помещались обзоры, где Турбо Паскаль по удобству написания коротких программ сравнивался с Бейсиком. Специалисты отмечали также скорость работы созданной Каном системы, которая действовала во много раз быстрее конкурирующих с ней программ.



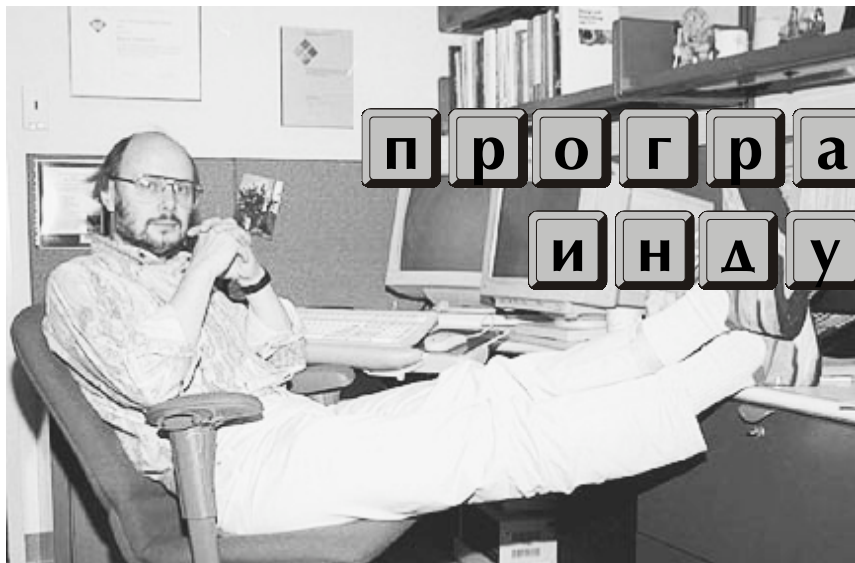
За первые два года было продано не менее 300 тыс. копий системы Турбо Паскаль, что превысило объем продаж всех прочих компиляторов для микрокомпьютеров. Подобно тому, как десятью годами раньше новая версия Бейсика “вывела на орбиту” фирму Microsoft, так Турбо Паскаль перевел Borland International в разряд основных производителей программного обеспечения [3, 4]. В 1985 году Кан с легкостью истратил почти 50 тыс. долларов на организацию приема, пригласив на него 600 клиентов фирмы Borland и других гостей. По случаю своего успеха (достигнутого в 33 года) он исполнил несколько музыкальных произведений на саксофоне [3].

[Во второй половине 1990-х годов, после приобретения двух компаний, фирма сменила название на INPRISE (от *IN*tegrated *enterPRI*SE — “объединенное предприятие”), однако вскоре было решено вернуться к прежнему названию, но с небольшими изменениями [8]: если раньше компания именовалась Borland International, то теперь ее полное название Borland Software Corporation.]

## Литература

1. Язык Паскаль // Информатика, № 34/2000.
2. Фаронов В.В. Программирование на персональных ЭВМ в среде Турбо Паскаль. М.: Изд-во МГТУ, 1991.
3. Язык компьютера: Пер. с англ. М.: Мир, 1989.
4. Толковый словарь по вычислительной технике (Microsoft Corporation): Пер. с англ. М.: Издательский отдел “Русская редакция” ТОО “Channel Trading Ltd.”, 1995.
5. Малыгина М.П., Частиков А.П. Языки программирования: Паскаль // Новое в жизни, науке, технике. Сер. “Вычислительная техника и ее применение”, № 6/90.
6. Пройдаков Э.М., Теплицкий Л.А. Англо-русский толковый словарь по вычислительной технике, Интернету и программированию. Изд. 2-е, испр. и доп. М.: Издательско-торговый дом “Русская редакция”, 2000.
7. Приемник Алгола // Информатика, № 41/2001.
8. Черняк Л. “Борланд” снова Borland // Computerworld Россия, 03.04.2001.





# Я З Ы К П Р О Г Р А М М Н О Й И Н Д У С Т Р И И

“Появление благодаря работам Бьерна Страуструпа языка Си++, ставшего фактически главным языком программной индустрии, а также повальное увлечение объектно-ориентированным проектированием и программированием заставило уйти в тень его именитого предшественника — язык Си” [1].

“Сообщество сторонников Си++ страдало от чрезмерной популярности этого языка: он постоянно становился объектом нападок, ведь в современном мире, живущем по законам коммерции, честная борьба — большая редкость”.

Бьерн Страуструп

В начале 1980-х годов в научно-исследовательской фирме Bell Telephone Laboratories американской корпорации AT&T (*American Telephone and Telegraph*) Бьерном Страуструпом в результате дополнения и расширения языка Си был создан язык, получивший название “Си с классами” [2, 3]. В 1983 году это название было заменено на Си++. Здесь как бы сделано указание на то, что Си++ — язык следующего поколения по отношению к Си [4]: как известно, операция инкремента ++ увеличивает значение переменной на единицу. (При создании Си++ Страуструп искал базовый язык для включения в него концепции классов, взятой из языка Симула. Как альтернатива Си, по словам Страуструпа, рассматривались языки Модула-2, Ада, Смолток, Меса и СЛУ. Но предпочтение было отдано Си [1].)

Язык Си++ является языком объектно-ориентированного программирования.

Идея использования программных объектов развивалась разными исследователями в течение целого ряда лет. Одним из первых языков такого типа является Симула-67 [5, 6]. Утвердило же свой статус объектно-ориентированное программирование в созданном Аланом Кеем языке Смолток (*Smalltalk*), который появился в 1972 году [7, 8].

“Объект есть логическая единица, которая содержит данные и правила

(методы) обработки этих данных. В языке Си++ в качестве таких правил обработки выступают функции, т.е. объект здесь объединяет в себе данные и функции, обрабатывающие эти данные” [2].

Объектно-ориентированный язык создает так называемое программное окружение в виде множества независимых объектов, каждый из которых отличается своими свойствами и способами взаимодействия с другими объектами. Каждый объект тут в каком-то смысле подобен компьютеру: он способен получать сообщения от других объектов и, в свою очередь, посылать сообщения, запоминая информацию и обрабатывать ее определенным образом. Описывая структуру обмена сообщениями между объектами, программист задает совокупность операций, которые и образуют программу. Можно использовать объекты для решения задач, не понимая механизма работы объектов, т.е. рассматривать их как своего рода “черный ящик”. Объектно-ориентированный язык скрывает многие детали функционирования системы внутри самих объектов. Но при необходимости программист может изменять детали внутреннего устройства объектов, формируя тем самым другие объекты, нужные для решения новых задач [9].

Каждый объект в объектно-ориентированном окружении обладает своими собственными характеристиками, однако объекты, имеющие одинаковые свойства и ведущие себя сходным образом, объединяются в группы, или классы.

“Понятие класса является центральным понятием языка Си++. Вообще класс — это определенный программистом тип, к которому относятся любые структуры данных, определенные посредством ключевых слов *class*, *struct* и *union*. При определении класса расширяются возможности языка Си++, так как в дальнейшем этот класс можно использовать как любой другой тип Си++.

Как только класс определен, можно создавать объекты этого класса, каждый из которых имеет члены, определенные в классе” [10] (а именно, данные-члены и функции — члены класса). Пример класса — графическая точка на экране. Данными-членами тут являются ее координаты, а также цвет. Функции-члены могут обеспечивать инициализацию точки, определение ее цвета, задание нового цвета.

Вообще говоря, “в принципе возможен объектно-ориентированный подход к проектированию даже в традиционных языках программирования, например, таких, как Си; однако использование объектно-ориентированных языков типа Си++ позволяет существенно облегчить работу...”.

Бьерн Страуструп является автором книг *The C++ Programming Language* (“Язык программирования Си++”), *The Design and Evolution of C++* (“Архитектура и эволюция Си++”), *Why C++ isn't just an Object-Oriented Programming Language* (“Почему Си++ — не просто язык объектно-ориентированного программирования”). Сейчас Страуструп возглавляет отдел программных исследований в лаборатории AT&T Labs (корпорация AT&T была в свое время реорганизована). Область его научных интересов охватывает технологии распределенных систем и операционных систем, сферу имитационного моделирования, проектирования и программирования [11].

## Литература

1. Богатырев Р. Летопись языков. Си // Мир ПК, № 8/2001.
2. Березин Б.И., Березин С.Б. Начальный курс Си и Си++. М.: Диалог-МИФИ, 1999.
3. Дьюхарст С., Старк К. Программирование на Си++. Пер. с англ. Киев: ДиаСофт, 1993.
4. А, В, С, ..., Си++ // Информатика, № 47/2000.
5. Киндлер Е. Языки моделирования: Пер. с чешского. М.: Энергоатомиздат, 1985.
6. Окольников В.В., Покровский С.Б. Симула // Математическая энциклопедия. М.: Советская энциклопедия, 1984. Т. 4.
7. Малыгина М.П., Частиков А.П. Языки программирования: Симула // Вычислительная техника и ее применение, № 5/90.
8. Знакомьтесь: компьютер: Пер. с англ. М.: Мир, 1989.
9. Язык компьютера: Пер. с англ. М.: Мир, 1989.
10. Карнов Б., Баранова Т. Си++: Специальный справочник. СПб.: Питер, 2000.
11. Страуструп Б. Мне бы хотелось объединения Си и Си++ // Computerworld Россия, 03.04. 2001.

# Д О Н А Л Ъ Д К Н У Т

*“На мировом рынке компьютерной литературы существует ряд книг, предназначенных для обучения основным алгоритмам и используемых при программировании. Их довольно много, и они в значительной степени конкурируют между собой. Однако среди них есть особая книга. Это трехтомник Искусство программирования Д.Э. Кнута, который стоит вне всякой конкуренции и входит в золотой фонд мировой литературы по информатике...”*

Издатели русского перевода книги Дональда Кнута  
“Искусство программирования”

**“У** каждой книги своя судьба. Одни появляются незаметно и так же незаметно исчезают в потоке времени, покрываясь пылью на полках библиотек. Другие в определенный период пользуются спросом у узкого круга специалистов — до тех пор, пока им на смену не приходят новые справочники. Третьи, поднимаясь над временем, оказывают мощное влияние на технологическое развитие общества. Книг, относящихся к последней категории, не так уж много. Их выход в свет — всегда праздник. Проходят годы, изменяются технологии, но новые поколения с неослабевающим интересом перечитывают их страницы. Именно к таким книгам относится многотомный труд известного американского ученого Д.Э. Кнута *Искусство программирования*”.

Профессор Анатолий Анисимов (из предисловия к [1])

“Если вы считаете себя действительно хорошим программистом, прочитайте *Искусство программирования* [Кнута]... Если вы сможете прочесть весь этот труд, то вам определенно следует отправить мне резюме”.

Билл Гейтс (Microsoft Corporation)

Дональд Эрвин Кнут (р. в 1938 г.) окончил в 1960 году отделение математики Кейсовского технологического института и через три года получил докторскую степень в Калифорнийском технологическом институте. С 1968 г. Кнут является профессором, а в настоящее время — почетным профессором информатики Станфордского университета. Является также почетным доктором многих университетов мира, в том числе Санкт-Петербургского университета. Основные области его науч-

ных интересов: теоретическое программирование, математический анализ алгоритмов, история и методология информатики [2]. Перу Дональда Кнута принадлежат 19 монографий и более 160 статей. О высоком международном авторитете этого ученого свидетельствуют переводы его трудов на многие языки мира.

Наибольшую известность Кнуту принесла монументальная серия монографий *The Art of Computer Programming (Искусство программирования)*, посвященная основным алгоритмам и методам вычислительной математики [1—3], из запланированных семи томов которой изданы в полном объеме первые три. Сейчас он полностью занят подготовкой новых книг данной серии, работу над первым томом которой он начал еще в 1962 году. Интересно, что одно из правил, которым руководствуется Кнут при написании своих книг, гласит: “Автору, заинтересованному в доходчивости излагаемого им материала, следует всячески избегать сносок, поскольку они отвлекают внимание читателя” [4].

Дональд Кнут является также создателем всемирно известной “компьютерной типографии”, которая состоит из систем  $T_E X$  и METAFONT, предназначенной для издания математических книг [1, 2, 4]. Вот что писали в свое время о первой из этих систем [5]: “Для документов с интенсивным использованием формул гораздо эффективнее специализированная система  $T_E X$ , разработанная математиком Д.Кнута, автором широко известного трехтомника *Искусство программирования*”.

...некоторые вещи, в частности включение рисунков, в  $T_E X$  делать не



так удобно, как в издательской системе. Но в итоге для документов с большим количеством формул общее время их подготовки к изданию сокращается в несколько раз. Поэтому многие известные западные издательства используют для набора книг и журналов по математике, физике, химии, технике и т.д. систему  $T_E X$ , а Американское математическое общество принимает для публикации только рукописи, подготовленные в  $T_E X$ ”.

Профессор Кнут удостоен многочисленных премий и наград, среди которых [1, 2]: премия Ассоциации по вычислительной технике (ACM Grace Murray Hopper Award), премия имени Тьюринга (ACM Alan M. Turing Award), медаль за научные заслуги (National Medal of Science), премия Института (профессионального общества) инженеров по электротехнике и радиоэлектронике (IEEE Computer Pioneer Award), премия Математической ассоциации США (MAA Lester R. Ford Award). В 1996 году он был удостоен престижной награды Kyoto Prize за достижения в области передовых технологий.

Живет в городе Станфордского университета.

### Литература

1. *Кнут Д.* Искусство программирования: Пер. с англ. Том 1: Основные алгоритмы. Изд. 3-е, Том 2: Получисленные алгоритмы. Изд. 3-е, Том 3: Сортировка и поиск. Изд. 2-е. М.: Издательский дом “Вильямс”, 2000.
2. *Грин Д., Кнут Д.* Математические методы анализа алгоритмов: Пер. со 2-го англ. изд. М.: Мир, 1987.
3. *Грэхем Р., Кнут Д., Паташник О.* Конкретная математика. Основание информатики: Пер. с англ. М.: Мир, 1998.
4. *Кнут Д.* Все про  $T_E X$ : Пер. с англ. Протвино: АО RDTEX, 1993.
5. *Фигурнов В.Э.* IBM PC для пользователя. Изд. 6-е, перераб. и доп. М.: ИНФРА-М, 1995.

# Е В Г Е Н И Й В Е С Е Л О В

## И Л Е К С И К О Н

Автор появившегося в 1985 году текстового процессора Лексикон (греч. *lexicon* — словарь) Евгений Веселов входил в свое время в знаменитую группу молодых программистов, работавших под руководством Виктора Брябрина в Вычислительном центре Академии наук СССР и готовивших оригинальные программы для только начинающих тогда у нас появляться IBM PC и совместимых с ними компьютеров. Веселов писал, что работа над текстовым процессором велась без какого-либо серьезного плана. В то время в группе Брябрина имелся уже свой неплохой, но не лишенный недостатков редактор, так сказать, для внутреннего пользования. “Захотелось создать что-нибудь получше только для того, чтобы легче было вводить и редактировать собственные программы и статьи” [1].

Популярность Лексикона, который называют “русским народным редактором”, основывалась прежде всего на том, что он дает возможность готовить тексты на русском языке, не требуя от пользователя никаких специальных навыков [1, 2]. С той или иной версией этой программы в то время имел дело практически каждый отечественный пользователь персонального компьютера. Лексикон был разработан как независимая часть интегрированной системы “Мастер”, содержащей, помимо него, табличный процессор, систему управления базами данных, графический процессор [3–8]. “Ранние версии Лексикона распространялись бесплатно, последующие — уже на коммерческой основе (с защитой от копирования). Некоммерческие версии Лексикона обладают сравнительно скромными возможностями: они не позволяют использовать различные гарниры символов, не обеспечивают работу с пропорциональными шрифтами, набор текста в несколько столбцов, употребление сносок и т.д. Но для многих пользователей этого и не требуется. Коммерческие версии Лексикона обеспечивают больше возможностей и удобнее в использовании” [3].

А истинный успех на рынке программ ожидал Лексикон после того, как усилиями компании “Микроинформ” его удалось представить как

настоящий программный продукт со всеми необходимыми атрибутами [1].

Компания “Микроинформ” была зарегистрирована летом 1988 года как шестидесятое негосударственное предприятие в России и всего лишь третья фирма в сфере компьютерного бизнеса (после компаний “Интерквадро” и “Диалог”). Первые годы основную прибыль фирме приносила торговля компьютерным оборудованием. Вырученные деньги позволяли развивать другие направления деятельности, в частности, заниматься прикладными разработками, обучением.

Однако “Микроинформу” не хватало сильных программистов, и Евгению Николаевичу Веселову предложили продолжить работу над Лексиконом в этой компании. Веселов не согласился сразу на переход, а вот некоторые его коллеги перешли в “Микроинформ” и начали трудиться над новыми версиями знаменитой программы. Веселов выступал в роли научного консультанта, но позже пришел в фирму на должность технического директора. Компания “Микроинформ” обеспечила всю необходимую инфраструктуру, которая делает программу рыночным продуктом. Был освоен промышленный выпуск тысячами тиражами, работали специальные службы продаж, было налажено обучение, планировалась разработка следующих версий. Покупатель получал красиво оформленную коробку, где, помимо самой программы, находились: руководство пользователя, паспорт, регистрационная карточка.

Компания выпустила (начиная с 1992 года) пять версий нового Лексикона для DOS, причем их подготовкой руководил Сергей Панкратов, а сам Веселов был занят уже разработкой Лексикона для Windows (в частности, Лексикона 2.0) и развитием интегрированной системы “Мастер”.

Тем временем пиратство и активное продвижение на российский рынок пакетов программ Microsoft Office делали свое дело. Выдерживать конкуренцию становилось все труднее. Вскоре Веселов перешел в компанию IBS, а в “Микроинформе” приняли решение закрыть “программное” направление [1, 9].



Тем не менее “русский народный текстовый редактор” жив. Отказавшись от дальнейшей работы над продуктом, фирма “Микроинформ” стремилась сохранить торговую марку и в результате передала права на нее дочернему предприятию компании “АйТи” — фирме “Арсеналь” [1], которая продолжила разработку нового Лексикона для Windows.

### Литература

1. Дубова Н. Лексикон в коробке // Computerworld Россия, № 29/2000.
2. Катаев А.И. Текстовый процессор Лексикон (от “н” до “с”). М.: Радио и связь, 1992.
3. Фигурнов В.Э. IBM PC для пользователя. Изд. 6-е, перераб. и доп. М.: ИНФРА-М, 1995.
4. Веселов Е.Н. Интегрированная система “Мастер” для ПЭВМ. М.: Финансы и статистика, 1989.
5. Кренкель Т.Э., Козан А.Г., Тараторин А.М. Персональные ЭВМ в инженерной практике. М.: Радио и связь, 1989.
6. Сыромолотов Е.Н. Текстовые редакторы Лексикон и Чирайтер. М.: Военная академия химической защиты им. С.К. Тимошенко, 1990.
7. Смородинский А.В., Посудина А.Ю. Текстовые процессоры для IBM-совместимых ПК // Интеркомпьютер, № 2/90.
8. Почти забытый? // Информатика, № 48/2000.
9. Козловский Е. Как нам купить и обустроить компьютер. Изд. 3-е, перераб. в корне и обильно доп. М.: ABF, 1997.



Д М И Т Р И Й

Л О З И Н С К И Й

**Д**митрий Николаевич Лозинский родился 27 июня 1939 года в Москве. В 1961 году окончил механико-математический факультет МГУ им. М.В. Ломоносова. С 1965 года работал в Главном вычислительном центре (ГВЦ) Госплана СССР. Позднее после ряда реорганизаций оказался в Министерстве экономики России. В ГВЦ первое время занимался экономическим моделированием. В 1966 году начал изучать программирование и примерно через год полностью перешел на этот вид деятельности [1]. Занимался задачами обработки данных.

Осенью 1988 года Дмитрий Лозинский обнаружил в Госплане вирус Vienna и написал антивирусную программу, которой дал название Aidstest [1—4]. Поскольку все время появлялись новые вирусы, ее пришлось совершенствовать. В 1990 году, в связи с невозможностью распространять программу в одиночку, Лозинский заключил договор с Научным центром СП «Диалог» при Вычислительном центре Академии наук (ВЦ АН) СССР. Теперь компания, с которой тогда был заключен договор, носит название ЗАО «ДиалогНаука», а Дмитрий Николаевич Лозинский является председателем совета директоров этой организации.

Официальный (и триумфальный) выход Aidstest на рынок состоялся в конце 1990 года на выставке Softool в Москве. «Диалог» начал продавать программу всего за три рубля, и люди часами стояли в очереди, тянувшейся через весь выставочный зал. И это за программой, в которую ее автор никогда не встраивал никакой защиты от копирования! Принципиально. По мнению Лозинского, бессмысленно ставить защиту от копирования — будут «ломать», да

еще и злиться, что пришлось ломать. А честные люди и так купят.

Технологические решения, которые были реализованы в программе Aidstest, давно себя исчерпали. Уже в 1996 г. «ДиалогНаука» начала распространять антивирус Doctor Web, который разрабатывается в Санкт-Петербурге командой под руководством Игоря Данилова. В течение некоторого времени «по инерции» поддерживались оба продукта, но в 1998 г. Aidstest официально сошел с дистанции. В течение нескольких последних лет сотрудники «ДиалогНауки» старательно занимаются антирекламой продукта, который в свое время принес всенародную известность и уважение не только автору, но и фирме. Надо сказать, что такая антирекламная кампания начала приносить ощутимые успехи лишь в последнее время.

Aidstest запомнился пользователям не только надежностью, низкой ценой и регулярностью выхода очередных версий. В течение многих лет в комплект Aidstest входил файл, содержащий неформальные описания обезвреживаемых вирусов с комментариями Дмитрия Николаевича. Множество пользователей, обновив версию программы, первым делом прочитывали этот файл, содержащий блестящие — иногда ироничные, иногда весьма эмоциональные замечания.

«...видно, что автор очень спешил напакостить, не успев в достаточной мере изучить MS-DOS».

«Весьма тяжелое впечатление оставляет сочетание хорошего уровня программирования и столь примитивного результата».

«...при загрузке зараженной программы выдает текст

«СЛАВА ГЕРОЯМ!».

Некто неизвестный в порыве благородного гнева заменил этот текст на «АВТОР ВИРУСА — КРЕТИН» и пустил дальше новую разновидность... «И что ты смотришь на сучок в глазу брата твоего, а бревна в твоём глазу не чувствуешь» (Матфея 7,3)».

«...против подлости программиста пользователь абсолютно беззащитен».

«Настоятельно советую до изготовления следующих версий [вируса] поучиться программированию,

чтобы не использовать 10 команд там, где вполне достаточно двух».

«На этот раз автором одержана большая победа — заражаются и EXE-программы. Большая просьба к автору этого вируса, как и ко многим остальным, — не выпускать в свет вирусы с такими грубыми ошибками. Мне жаль тратить лишнее время еще и на оживление ваших вирусов из неработоспособных программ».

«...приписывает к началу AUTOEXEC.BAT: @echo off  
echo Tina, do you love me?  
pause >nul  
Тина, не стоит любить таких!»

«Имеется указание о том, что это версия 3, однако ничего принципиально нового обнаружить не удалось. Система виснет так же быстро».

«...[вирус]издает серию пискон, устанавливает дату «1 апреля» и через некоторое время выдает на экран текст, соответствующий уровню интеллекта автора».

В апреле 1996 года в журнале «Деньги» опубликовано интервью с Лозинским «Человек, который поймал вирус» (в рубрике «Моя работа»), из которого читатели узнали, что он думает о своей деятельности: «Я делаю одну и ту же работу уже восемь лет. Сегодня я обязан — интересно мне это или нет — заниматься антивирусными программами. Никогда в жизни я столько не работал над одной и той же задачей. Скучно это...»

В августе 1999 года председателем совета директоров ЗАО «ДиалогНаука» Дмитрию Лозинскому в соответствии с Указом Президента Российской Федерации был вручен орден Дружбы.

#### Литература

1. Страница [www.dialognauka.ru/personal/lozinsky.htm](http://www.dialognauka.ru/personal/lozinsky.htm)
2. *Островский С.Л.* Компьютерные вирусы // Информатика, № 32/2001.
3. *Козловский Е.* Как нам купить и обустроить компьютер. Изд 3-е, перераб. в корне и обильно доп. М.: АБФ, 1997.
4. *Фигурнов В.Э.* IBM PC для пользователя. Изд. 6-е, перераб. и доп. М.: ИНФРА-М, 1995.
5. Файл AIDS.VIR.TXT из последней версии программы Aidstest.

# Л И Н У С

Линус Торвалдс начал работу над операционной системой Linux в 1991 году, когда был еще студентом Хельсинкского университета [1]. В этом деле ему помогали программисты из разных стран мира. Можно сказать, что Linux представляет собой версию операционной системы Unix, созданную “на общественных началах” [2].

Впрочем, Питер Салус, директор компании Specialized Systems Consultants, выпускающей журнал Linux Journal, считает, что работа над системой Linux началась еще в 1979 году, на конференции ассоциации Usenix в Торонто. Одним из участников конференции был Эндрю Танненбаум, профессор Амстердамского университета.

“Представители корпорации AT&T (*American Telephone and Telegraph*) обнародовали новую структуру цен для операционной системы AT&T Unix System V, — вспоминает Салус. — Была установлена стоимость полной коммерческой версии этой системы — около 40 тыс. долл. на каждый процессор. Льготная цена для учебных заведений составила 7,5 тыс. долл. Легко себе представить, как было воспринято это известие” [1].

Ни одно учебное заведение не могло себе позволить платить такие деньги, но многие преподаватели считали, что их студенты должны изучать Unix. Танненбаум нашел выход из положения: он разработал систему Minix, усеченную версию Unix, которая могла работать на настольном компьютере.

Торвалдс, устав от бесплодных попыток получить машинное время на машине MicroVAX (фирмы DEC), принадлежащей университету, где он учился, сначала решил заняться системой Minix. Однако она, являясь весьма подходящей для обучения, не годилась в качестве полноценной операционной системы.

“Линус Торвалдс... не пытался создавать Linux с нуля. Вместо этого он начал использовать тексты и идеи Minix — небольшой Unix-подобной операционной системы для ПК. В конечном счете весь текст Minix пришлось выбросить или полностью переписать, но пока он присутствовал в проекте, то служил, подобно



“бегункам” для ребенка, и эти “бегунки” в конце концов превратились в Linux” [3], а точнее, в операционную систему Linux Version 0.02 — своего рода ядро, содержащее все основные компоненты Unix [1].

Возможно, Linux так бы и осталась на этой начальной стадии своего развития, если бы не Интернет. Именно благодаря сети Интернет о новой системе узнали десятки тысяч специалистов.

“Linux стал первым проектом, в котором были предприняты осознанные и успешные попытки использовать весь мир как источник талантов... Линус стал первым, кто понял, как нужно играть по новым правилам, которые возникли благодаря широкому распространению сети Интернет...”

Стиль разработки, предложенный Линусом Торвалдсом, воспринимался как нечто удивительное. Никакой спокойной и благоговейной атмосферы... Вместо этого сообщество Linux напоминало огромный гонимый восточный базар, с множеством разнообразных программ и подходов (который символизировали узлы с архивами Linux, куда отправляли свои решения все, кто хотел), откуда логически связанная и стабильная система, казалось, могла возникнуть только благодаря чуду, да и то не одному” [3].

Успех этой операционной системы объясняется несколькими факторами.

“Одна из главных причин успеха Linux заключается в том, что ее от-

крытые исходные коды предоставляют программистам возможность влиять на нее, — в альтернативных системах это исключено...

У Microsoft есть повод для беспокойства хотя бы потому, что Linux стала очень популярным инструментом обучения в школах США и некоторых стран Западной Европы” [4].

Важнейшей особенностью Linux является то, что она распространяется бесплатно по открытой лицензии GNU в рамках Фонда бесплатного программного обеспечения (Free Software Foundation) [2, 5]. Лицензия GNU разрешает бесплатное распространение программного обеспечения при условии, что вместе с загрузочными модулями программ поставляются и их исходные тексты. Коммерческое использование программного обеспечения при этом сильно ограничено.

[GNU (от *GNU's Not Unix*, т.е. “GNU — это не Unix”) — проект создания некоммерческой Unix-подобной системы, начатый в 1983 году. Объединение в 1991 году разработок GNU с ядром Linux превратило незавершенную систему GNU в полноценную ОС. Количество пользователей систем “семейства” GNU/Linux очень велико (более десяти миллионов).]

Сейчас за разработку ядра операционной системы по-прежнему отвечает Линус Торвалдс. “Линус придиричиво следит за тем, чтобы ядро Linux оставалось настолько компактным и удобным, насколько возможно, чтобы, как говорится, весь пар не ушел в гудок”, — отмечает Салус [1].

## Литература

1. Брандел М. Линус и его Linux // Computerworld Россия, № 27—28/2000.
2. Фролов А.В., Фролов Г.В. Создание web-приложений: Практическое руководство. М.: Издательско-торговый дом “Русская редакция”, 2001.
3. Реймонд Э.С. Собор и базар // Открытые системы, № 9—10/1999.
4. Кирюхин В.М. XIII международная олимпиада школьников по информатике // Информатика, № 37/2001.
5. Мостицкий И.Л. Новейший англо-русский толковый словарь по современной электронной технике. М.: ЛУЧШИЕ КНИГИ, 2000.

## ЧТОБЫ ПРИОБРЕСТИ ЗАИНТЕРЕСОВАВШИЕ ВАС КНИГИ НАЛОЖЕННЫМ ПЛАТЕЖОМ:

1. Укажите в купоне ваши фамилию, имя, отчество, индекс и почтовый адрес.
2. Выберите нужные вам книги из списка (не более 5 названий в один купон) и укажите соответствующий номер лота и количество экземпляров.
3. Вырежьте и вышлите купон в конверте с пометкой «Книга — почтой» по адресу: 150000, Ярославль, ул. П. Морозова, д. 5, ая «Первое сентября». Телефоны для справок: (0852) 45-07-77, (095) 249-28-77.

### КУПОН Книга – почтой Первое сентября

Объединение педагогических изданий

#### ЗАПОЛНЯЕТСЯ ПЕЧАТНЫМИ БУКВАМИ!

|            |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |
|------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| ФАМИЛИЯ    |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |
| ИМЯ        |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |
| ОТЧЕСТВО   |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |
| ИНДЕКС     |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |
|            |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |
|            |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |
| НОМЕР ЛОТА | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| НОМЕР ЛОТА | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| НОМЕР ЛОТА | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| НОМЕР ЛОТА | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| НОМЕР ЛОТА | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

00-3-00

ОБРАЗЕЦ: 0123456789

Указанная в купоне цена включает в себя стоимость доставки наземным транспортом от издательства до вашего почтового отделения (при авиаперевозках сумма налогового платежа увеличивается на авиатариф). Рассылка производится только на территории РФ. Если полученная бандероль содержит неполный заказ, это означает, что заказ разбит на две или более бандероли.

| Номер лота | Название  | Цена     |
|------------|---|----------|
| 10-01-0    | С.Л. Соловейчик. Пушкинские проповеди   | 55 руб.  |
| 10-02-0    | С.Л. Соловейчик. Последняя книга  | 65 руб.  |
| 10-05-0    | С.Л. Соловейчик. Педагогика для всех  | 75 руб.  |
| ▼ 10-03-0  | Школа сотрудничества<br>(Сборник статей по педагогике сотрудничества)         | 38 руб.  |
| 27-01-9    | Я иду на урок физики. 7 класс (в 3-х книгах)                                  | 114 руб. |
| 29-01-0    | Я иду на урок химии. 8–11 классы  | 38 руб.  |
| 29-02-0    | Я иду на урок химии.<br>Летопись важнейших открытий. XVII–XIX вв.             | 38 руб.  |
| 21-01-0    | Я иду на урок математики. 5 класс   | 38 руб.  |
| 21-01-7    | Я иду на урок математики. Тесты. 5 класс                                      | 15 руб.  |
| 21-02-0    | Я иду на урок математики. 6 класс   | 38 руб.  |
| 21-03-0    | Я иду на урок математики. Алгебра. 7 класс                                    | 38 руб.  |
| 24-01-0    | Я иду на урок русского языка. Диктанты. 5–9 классы                            | 38 руб.  |
| 12-01-0    | Я иду на урок биологии. Зоология. Беспозвоночные                              | 38 руб.  |
| 12-02-0    | Я иду на урок биологии. Зоология. Рыбы и земноводные                          | 38 руб.  |
| 12-03-0    | Я иду на урок биологии. Зоология. Пресмыкающиеся                              | 38 руб.  |
| 12-04-0    | Я иду на урок биологии. Человек и его здоровье                                | 38 руб.  |
| 19-02-0    | Я иду на урок истории.<br>Древнейшая и древняя история                        | 38 руб.  |
| 19-04-0    | Я иду на урок истории.<br>Материалы по истории средних веков                  | 38 руб.  |
| ▼ 19-03-0  | Л.А. Кацава. Россия в 1917–1918 годах. 10–11 классы                           | 38 руб.  |
| 14-01-0    | Я иду на урок географии.<br>Физическая география материков и океанов          | 38 руб.  |
| 14-02-0    | Я иду на урок географии.<br>История географических открытий                   | 38 руб.  |
| 14-03-0    | Я иду на урок географии. Физическая география России                          | 38 руб.  |
| 20-02-0    | Я иду на урок литературы. 5 класс   | 38 руб.  |
| 20-03-0    | Я иду на урок литературы. 9 класс   | 38 руб.  |
| 20-01-0    | Я иду на урок литературы. 10 класс  | 38 руб.  |
| 20-04-0    | Я иду на урок литературы.<br>Готовимся к экзаменационному сочинению. 11 класс | 38 руб.  |

▼ Тираж ограничен.

Указанные цены действительны до 31 января 2002 года

Если вы еще не успели подписаться на нашу газету, то это можно сделать с марта 2002 г.!

В 2002 году "Информатика" предложит подписчикам сотни интересных и полезных материалов. Помимо наших постоянных рубрик, мы активно развиваем специальные проекты. Расширится список курсов, публикуемых на "Страницах повышения квалификации". По сложившейся традиции в № 8/2002 будет опубликован план тематических выпусков очередного "Жаркого лета".

**индекс подписки – 32291**

в каталоге «Роспечать. Газеты и журналы. 2001» см. с. 88  
в каталоге «Роспечать. Газеты и журналы. 2002» см. с. 95

Ф. СП-1

Министерство связи  
Российской Федерации  
"Роспечать"

АБОНЕМЕНТ на газету

**32291**

**ИНФОРМАТИКА**

(индекс издания)

наименование издания

Количество комплектов

на 20\_\_ год по месяцам

|   |   |   |   |   |   |   |   |   |    |    |    |
|---|---|---|---|---|---|---|---|---|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|   |   |   |   |   |   |   |   |   |    |    |    |

Куда

(почтовый индекс)

(адрес)

Кому

(фамилия, инициалы)

**ДОСТАВОЧНАЯ КАРТОЧКА**

ПВ

место

ли-тер

на газету

**32291**

**ИНФОРМАТИКА**

(индекс издания)

(наименование издания)

Стои-мость

подписки \_\_\_\_\_ руб.  
пере-адресовки \_\_\_\_\_ руб.

Количество комплек-тов

на 20\_\_ год по месяцам

|   |   |   |   |   |   |   |   |   |    |    |    |
|---|---|---|---|---|---|---|---|---|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|   |   |   |   |   |   |   |   |   |    |    |    |

Куда

(почтовый индекс)

(адрес)

Кому

(фамилия, инициалы)



## Начало

**В** начале 1970-х годов появились первые микрокомпьютеры, а в январском номере американского журнала Popular Electronics за 1975 год была помещена первая часть статьи, в которой описывался набор для сборки "первого в мире мини-компьютера, способного konkurrировать с коммерческими моделями" [1]. Это был компьютер Алтайр-8800.



Алтайр-8800

Новая машина (которую впоследствии назвали еще сибиряками) обладала весьма ограниченными возможностями. Обратная память Алтайра имела объем всего 256 байт, экран отсутствовал, а покупка приходилось снабжать машину блоками расширения памяти, мониторами и т.д.

Кроме того, компьютер выпускался без программного обеспечения, и если его владелец хотел, чтобы он делал что-то полезное, ему было необходимо подготовить соответствующую программу (принимать в двойном коде).



Молодые программисты Пол Аллен и Билл Гейтс (в нижнем ряду крайние справа и слева соответственно) написали интерпретатор Бейсика для первого в мире ПК Altair компании MITS.

В то время в компьютерной фирме Honywell, располагавшейся недалеко от Бостона, работал молодой программист Пол Аллен. Узнав о новинке, он отправился к своему другу, переводчику Уильяму (Биллу) Гейтсу. Обсудив ситуацию, молодые

люди решили создать для новой машины программное обеспечение. О следующем их шаге, наверное, всегда будут упоминать в рассказах, связанных с историей компьютерной техники. Гейтс и Аллен позвонили в компанию MITS и сообщили ее владельцу Эду Роберту, что подготовили программу, способную переводить операторы языка Бейсика в машинный код для Алтайра, причем готовы продать ее Роберту. Тот, не задумываясь, ответил, что купит интерпретатор Бейсика, способный работать на Алтайре.

Гейтс и Аллен быстро взялись за дело. Надо сказать, что будущие еще ученики средней школы в Сиксте они вступили в сообщество жавороков. Их школьный клуб арендовал компьютер (покупая машинное время), и Гейтс самостоятельно научился программировать, когда ему было всего 13 лет. Другим даже основали компанию Traf-o-Job и пытались продавать результаты своих компьютерных исследований транспортных потоков в небольших городах, расположенных близ Сикста. Особого успеха эти попытки не имели, однако слух о способностях двух молодых людей широко распространился, и им предложили написать программу для компаний, обслуживающих электростанции северо-восточной части США и Канады. Никто не знал, что тогда мы были еще школьниками! — вспоминал позднее Гейтс.

Приобретенный опыт весьма пригодился при разработке программного обеспечения для компьютера Алтайр. Причем нужно отметить, что Гейтс и Аллен его даже не видели. Все, чем они располагали, — это рукописное, касающееся системы команд микропроцессора Intel-8080 на базе которого был

построен Алтайр, и сведения о характеристиках машины. Для отладки же использовался большой компьютер. Когда первый вариант интерпретатора был получен, другим подготовили соответствующим образом перфокарты и Аллен полетел на самолете в город, где находилась компания MITS, чтобы показать программу Роберту. К тому времени Роберт оборудовал одну из своих машин дополнительной платой памяти, телеprinterом и устройством для чтения перфокарт. Во время полета Аллен сообразил, что если в программе есть хоть одна ошибка, то работать она не будет. Однако телеprinter охнул, и он бумажно составил ее прямо в самолете.

На следующий день Аллен (нес последнее именованное) шепнул интерпретатор Бейсика в машину и запустил программу на выполнение. Он понимал, что если в программе есть хоть одна ошибка, то работать она не будет. Однако телеprinter охнул, и он бумажно составил ее прямо в самолете.

Примерно через 7К. Затем он задал компьютеру простейшую задачу: "PRINT 2 + 2". Машина сразу ответила: "4". Потом разработчик интерпретатора отмечал, что Роберт и его коллеги из фирмы MITS были крайне изумлены, поскольку раньше никто из них не видел, чтобы эта машина что-нибудь делала. Роберт купил программу (на успехе окончательного авторского гонимой с каждого проданной оземлетра), а Аллен и Гейтс вскоре во второй раз в жизни организовали компанию, назвав ее теперь Microsoft [1-3]. Они продавали совершенствованного интерпретатора для Алтайра, а кроме того, написали ряд других программ для малых машин. "Вообще-то это была ерунда, — позднее говорил Гейтс, — но пользователи были довольны" [1]. В течение долгого времени фирму Microsoft возглавлял Гейтс. Затем, с 1998 года, некоторое время ее главным был Аллен. Сегодня председатель правления (совету директоров) и главный архитектор программного обеспечения корпорации Microsoft Билл Гейтс является одним из самых богатых людей в мире (и самым богатым среди тех, кто получил свое состояние не по наследству).

Литература  
1. Язык компьютера. Пер. с англ. М.: Мир, 1989.  
2. Знакомьтесь: компьютер. Пер. с англ. М.: Мир, 1989.  
3. Метельман М.П., Часиков А.Л. Язык программирования Бейсик // Новое в жизни, науке, технике. Сер. "Вычислительная техника и ее приложения". № 1/90.

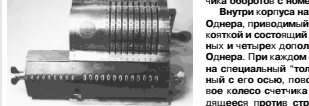
## Вильгельм

### Однер

«Колесо Однера стало основой современных арифмометров. В дальнейшем на своем арифмометре пришли настольные механические и электромеханические машины, а позднее — малые электронные цифровые машины» [1].



помощью прямого оборота рукоятки прибавить к а с помощью обратного оборота — вычитать из а. Таким же образом определялись и все обратные задачи. Для уможнокления а на следовало погасить обе счетки с установкой множителя а на барабане. Если начитывали  $b = 437$ , то переводя каретку в положение 3, нужно было сделать четыре прямых оборота (в счетчике результатов появилось число  $a \cdot 400$ ), затем в положении 2 сделать три прямых оборота (в счетчике результатов к  $a \cdot 400$  добавилось  $a \cdot 30$ ) и в положении 1 — семь прямых оборотов (в счетчике добавилось еще а 7). Таким образом в счетчике оборотов оказался множитель (437), а в счетчике результатов получалось произведение. В данном примере можно было обойтись и меньшим количеством оборотов, сделав в положении 3 и 2 по четыре прямых оборота, а в положении 1 — три обратных оборота.



Внутри корпуса находился барабан Однера, приводимый поворачиванием рукоятки и состоящий из девяти основных и четырех дополнительных колес Однера. При каждом обороте барабана специальный "толкатель", связан с его осью, поворачивал цифровое колесо счетчика оборотов, находившегося против стрелки на крышке корпуса, на 1/10 полного оборота, причем при первом повороте рукоятки в окое окно появилось последовательно белые цифры 1, 2, ..., 8, 9, а при обратном — красные цифры 1, 2, ..., 7, 8 и белая цифра 9.

В 1890-х годах Однер и его компания налаживали выпуск арифмометров — по 500 штук в год, причем скоро Однер стал единственным хозяином предприятия.

В 1914 году в одной только России насчитывалось более 25 тысяч арифмометров Однера. Большими сериями они выпускались и за рубежом. В первой четверти XX века арифмометры Однера являлись единственным широко применявшимся математическими машинами [5, 6]. Вообще же колесо Однера использовались и в других счетных машинах [1-3, 7], имеющих существенные конструктивные отличия от арифмометра.

Литература  
1. Борозина Р.А., Малиновский Б.И., Рабинович З.Л. Вычислительная техника // Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 1.  
2. Борозина Р.И. Энциклопедия кибернетики. Киев: Гл. редакция Украинской советской энциклопедии, 1975. Т. 1.  
3. Детская энциклопедия. М.: Изд-во Академии педагогических наук РСФСР, 1959. Т. 8.  
4. Гутер Р.С., Поланов Ю.Д. Об абзак до компьютера. Мад. 2-е изд., испр. и доп. М.: Знание, 1981.  
5. Плещинский Ю.И. Энциклопедия о компьютерах. Киев: Изд-во "Техника", 1983.  
6. Механические калькуляторы // Информатика. № 25/2001.  
7. Детская энциклопедия. М.: Изд-во Академии педагогических наук РСФСР, 1959. Т. 1.



Диagramмическая модель арифмометра В.Т. Однера

Еженедельная газета "Информатика" • Индекс подписки по каталогу "Роспечать" — 32291 в www.1september.ru

**Гл. редактор**  
С.Л. Островский  
**Зам. гл. редактора**  
А.И. Сенюков  
**Редакция:**  
Е.В. Андреева  
Н.Л. Беленькая  
Л.Н. Картвелишвили  
Н.П. Медведева  
**Дизайн и верстка:**  
Н.И. Пронская  
**Корректоры:**  
Е.Л. Володина,  
С.М. Подберезина

**©ИНФОРМАТИКА 2002**  
выходит четыре раза в месяц  
При перепечатке ссылка  
на ИНФОРМАТИКУ обязательна,  
рукописи не возвращаются

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

**Тел.: (095)249-31-38, 249-33-86. Факс (095)249-31-84**

**Адрес редакции и издателя:**  
121165, Киевская, 24  
тел. 249-48-96  
Отдел рекламы  
тел. 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",  
125993, ГСП-3, Москва, А-40, ул. "Правды", 24.  
Тираж 6000 экз.  
Срок подписания в печать по графику 26.12.2001.  
Номер подписан 26.12.2001.  
Заказ №  
Цена свободная

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

Газеты ИЗДАТЕЛЬСКОГО ДОМА: Первое сентября — гл. ред. Е.Бирюкова, Английский язык — гл. ред. А.Громовкина, Библиотека в школе — гл. ред. О.Громова, Биология — гл. ред. Н.Иванова, Воскресная школа — гл. ред. монах Киприан (Яценко), География — гл. ред. О.Коротова, Дошкольное образование — гл. ред. М.Аромштам, Здоровье детей — гл. ред. А.Лекманов, Информатика — гл. ред. С.Островский, Искусство — гл. ред. Н.Исмаилова, История — гл. ред. А.Головатенко, Литература — гл. ред. Г.Красухин, Математика — гл. ред. И.Соловейчик, Начальная школа — гл. ред. М.Соловейчик, Немецкий язык — гл. ред. М.Бузова, Русский язык — гл. ред. Л.Гончар, Спорт в школе — гл. ред. Н.Школьникова, Управление школой — гл. ред. А.Адамский, Физика — гл. ред. Н.Козлова, Французский язык — гл. ред. Г.Чесновца, Химия — гл. ред. О.Блохина, Чудесная газета — гл. ред. М.Аромштам, Школьный психолог — гл. ред. М.Сарган.