

# “Мир информатики”

Журнал для тех, у кого информатика – любимый школьный предмет

Выпуск № 13, июнь 2017 г.

011000100010101001001111100100011001001000101001001001011

## История информатики

### История одного байта

#### Предисловие

Приведенная ниже история была опубликована в Интернете в далеком 2001 году. Её автор — Дмитрий Галущенко. Но события, аналогичные описанным в ней, как следует из комментариев пользователей, происходили со многими программистами в 80–90-х годах прошлого века, когда приходилось учитывать серьезные ограничения по памяти, времени работы программы и т.д.

Итак, ...

Мне не хватало байта. Всего одного. Да, да — того самого, что из восьми бит состоит. Что? Нет, я не псих, хотя одному богу известно, сколь тонкой была граница, отделявшая меня от этого состояния. Но все по порядку.

Я программер. Но не просто программер. Я принадлежу к касте, которую иногда называют “системщиками”, иногда “кристалльщиками”. Вы знаете, что это такое? Я объясню, если потерпите. Мне никак не обойтись без специфики, но иначе вы не сможете понять дальнейшее.

Мы программируем чипы-однокристаллки, грубо говоря, это когда весь комп в одном кристалле. Программная память и память данных разделены и не взаимодействуют между собой. Программа не может быть запущена в оперативке. Глубина программного стека ограничена. Максимум, на что я могу рассчитывать, это восемь уровней вложения, причем я не могу изменять предельную глубину стека. О, вы не подумайте чего! У меня бездна ресурсов. Оперативки аж 128 байт! Это на все про все. Переменные, там то да се... Представили, да? С программной памятью тоже неплохо. Аж восемь килобайт. И пользоваться ей совсем несложно. Сначала нужно программно врубить нужный банк памяти, запустить в нем нужную процедуру, а по выходе из нее не забыть вернуться туда, где был. Да, еще надо иметь в виду, что в пределах банка я могу перемещаться только джампами и вызовами процедур, а переходы по условиям возможны только в пределах одной страницы, т.е. 256 байт.

Это значит, если я сравниваю два байта и надо ветвиться, но если метка не находится в пределах 256 байт, то это письмо на деревню дедушке, причем компилятор только в половине случаев предупредит, мол, широко шагаешь парень, штаны бы поберег. И это только цветочки! Ягодки я вам сейчас выложу, чтобы вы ими в полной мере могли насладиться. У меня нет команды вычитания. Вообще! Только add. Уж про такую роскошь, как умножение или деление, я вообще молчу, это для лентяев. Зато мне нужно обеспечить десятичную

математику. Вы проникаетесь потихоньку? Коды таких игрушек вылизываются так, что вам и не снилось, особенно если приходится решать задачи на пределе оперативной и программной памяти. Исходники переписываются далеко не один раз. Мне мало просто решить задачу. Я должен впихнуть ее в этот чертов кристалл! Ограничение по переменным, по размеру кода в целом, по размеру каждой процедуры и по числу вызовов. Малейший недосмотр и... стек продавлен, и тебя вышвыривает черт знает куда. И компилятор не поможет. Такое он не ловит... Вы думаете — это все? Не-е-ет, дорогие мои. Моя игрушка работает в реалтайме... Это когда напротив каждой крохотной процедурки моего кода нужно подсчитать и проставить время ее исполнения в миллисекундах. Мои модули не должны работать больше фиксированного времени, потому что мне надо еще сканировать кейборду и дисплей, поспевать за датчиками и выдавать управляющие сигналы, а все остальное должно работать, никак не мешая сканнингу, иначе я прозеваю нажатие кнопки или дисплей станет неприятно мерцать, меняя яркость. Но и это еще не все! У меня есть интерфейс. Обычный писишный<sup>1</sup> RS232C — так называемый “компорт”. Но если вы думаете, что это отдельный чип, мол, сунул ему байт, принял из буфера байт, то вы заблуждаетесь. Себестоимость. Я всё это делаю ручками, телепая единственный бит порта. Ручками кручу диаграмму стартов, стопов и данных. Итак:

— если я накатал код, решающий задание, но он не влезает в память — задача не решена. Если при этом я создал большее число переменных и они не помещаются в оперативку — задача не решена;

— если все оки-доки, но процедуры слишком длинны, разрушается диаграмма реалтайма — задача не решена;

— если процедуры слишком коротки, их много, исчерпывается стек — задача не решена.

Любой средней руки программист, поставь его в подобные условия, застрелится на второй день. Вообще, по моему убеждению, парни, работающие в этой области, имеют стальные нервы и неукротимую волю к победе. Мы редко общаемся с обычными программистами — нам в общем не о чём разговаривать. И не потому, что мы снобы или гордецы. Совсем необязательно. Нам трудно представить себе программирование под неисчерпаемыми ресурсами на языках высокого уровня. Мало винта? Купи другой, в чем проблема? Мало рамы? Купи еще, толкни в слот. Меги кодов? А я причем? Это компилятор виноват. Купите машину побольше. Это как разные планеты, и я надеюсь, вы поняли, почему.

Это как красивый белый океанский лайнер в бескрайних океанских просторах — плыви, куда хочешь. А вы попробуйте на нем в финских шхерах порулить. Или вдевать нитку в иголку среди ночи. Причем черную нитку.

Конечно, мы тоже имеем наборы кристаллов и выбираем их перед разработкой с большей дотошностью, чем жених невесту, и гадаем на кофейной гуще и прочих подручных средствах, а хватит ли? Но, если выбор сделан — назад ходу, как правило, нет.

Ну что ж. Я ввел вас в предметную область и могу продолжить свой рассказ.

Кристалл не понравился мне сразу. Я сразу понял — тесноват. Все на пределе. Законных 20% запаса по ресурсу, на возможные ошибки — не соблюсти. Однако остальные кандидаты были сильно избыточны, и потому дороги и нерациональны. Себестоимость решила все, я, наконец, выдал свое согласие, и мощные и гордые красавцы Intel’ы и Моторолы последних моделей остались за бортом.

Поначалу все шло как надо. Пару месяцев работы и кристаллы были нафаршированы кодом, испытания прошли нормально, платы разведены и отработаны, медленно провернулись шестерни и, набирая ход, закрутились, приводя в действие сложную и громоздкую машину производства. И только у меня изредка ёкало сердечко, ведь все сделано впритирку! Три, от силы пять процентов, ресурсов остались свободными. А это чертовски мало, поверьте мне на слово.

---

<sup>1</sup> От сокращения РС — personal computer. — Прим. ред.

Хотя работа была сделана весьма неплохо, и я получил свое законное вознаграждение в виде порции удовольствия от сделанного. Конечно, пара мест сделана довольно авантюрно, местами рыхловато, но зато и несколько изюминок получилось.

И тут... Тут все и началось. Выяснилось, что недостает очень важной функции прибора, которую проморгали постановщики. Причем даже не они, а заказчик. Это он вдруг вспомнил, что вот, мол, ребята, тут еще вот такая мелочь должна быть. Ну, сущая ерунда, чесслово, но без нее никак. Ну, забыли при постановке про нее, с кем не бывает? Но это ж несложно совсем добавить, по сравнению с остальными вашими наворотами? Опять же, слава богу, что не датчик забыли. Это всего лишь программа!

Ох уж эти постановщики!!! Их карма понять, что надо заказчику! Даже если для этого надо распилить ему черепушку и просеять через сито все ее содержимое!

Но криком делу не поможешь. И я на две недели засел дома, запретив меня беспокоить. В мозгу завелись маленькая сирена и светофорчик. Би-и-п! Вспышка красного! Первый программный банк исчерпан! Репакинг. Оптимизация размещения кода по страницам и банкам памяти. Би-и-п! Оперативная память исчерпана! Пересмотр функциональности процедур. Эту переменную на фиг. И без этих можно обойтись, если тут по другому пути пойти.

Би-и-п! Провал стека! Прям мордой в дно. Как это!? Я уже на восьмом уровне!? Би-и-п! Выход меток за пределы видимости внутривстраничных переходов! Би-и-п! Нарушение условий реалтайма, процедуры жрут слишком много времени! Би-и-п! Исчерпан второй банк памяти!

И так много-много раз. Кропотливо, байт за байтом я вдвигал тело этой Проклятой Новой Функции, непрерывно переупаковывая размещение кода по страницам, банкам, оптимизируя размер кода, график реалтайма, использование оперативной памяти, а то и попросту переписывая модули с нуля. Может, вы думаете, восемь килограмм бинарного кода — это мало? Ха! Инструкции ведь одно и двухбайтовые. Это вам не трехбайтовый зайлог<sup>2</sup> или даже интеловский восьмидесятник. Временами ко мне забегал приятель, как, мол, и что, но я был мало расположен к трепу “за жизнь”.

Через полторы недели я понял, что дело худо. Я располагал более чем полдюжиной решений и уже стоял на месте. Я знал каждую процедуру и функцию наизусть, а каждый байт в лицо! Все было впустую. Картинка замерла. Она не хотела оживать! Мне не хватало одного байта. Это показывали все варианты решений. Всего одного байта!

Забежав к другу, я сбросил ему все варианты решений с временными диаграммами и планами банков и вновь вернулся в свою берлогу. Спать. Во что бы то ни стало — спать. Нужна ясная голова. Нужна новая идея. Я опустошен. Следующие несколько дней не дали ничего. Я собирал и разбираал код, нанизывая его, как сверкающие бусины, и упаковывал его в фигурки разнообразной формы, пытаюсь найти форму, в которой все эти элементы головоломки сложатся в одно целое без единого зазора и также без зазора войдут в заготовленное для них место. Места не хватало... Одного байта! Я мял код, как глину, я выделял с ним все что угодно, но... один, всего один байт! Наверное, нечто похожее испытывает музыкант, написав симфонию, пытаюсь найти одну единственную ноту, чтобы заставить звучать свое произведение. Или художник ищет тот самый, неповторимый мазок, который оживит картину. А без этого все мертво и весь труд годится только на помойку...

Как-то в полусне я оделся и вышел на улицу. Под ногами мерзко визжал снег. Кругом все было серо и как-то тускло. Мне больше не нужен был ни компьютер, ни распечатки. Вся схема была во мне... Или вне меня? Она помигивала разноцветными просвирками, имея форму причудливых и чем-то даже красивых трехмерных фигур, или это какие-то ажурные конструкции? Тоненько попискивали контрольные маркеры временных отметок, висящие впритирку к этим кристаллам странных, каких-то завораживающих форм. И все это летело, куда-то перемещаясь и вращаясь в каком-то странно меняющем форму канале? Трубе? Оно

---

<sup>2</sup> Речь идёт о процессоре фирмы Zilog. — Прим. ред.

проносится вплотную ко всем его стенкам как по команде невидимого штурмана, в нужный момент разворачиваясь по непонятно какому наитию, чтобы выступающая грань не зацепилась за препятствие. Но каков его капитан или кто там? Штурман? Ведь не пройдет же! Там нельзя пройти! Но нет! Чудесным образом все сооружение как-то грациозно изворачивается, ровно в нужное мгновение и беззвучно проскакивает. Нет — величаво минует препятствие... А впереди следующее... И, вдруг, край ажурного сооружения своим крохотным выступом цепляется. Визг и грохот! Лопаются и сминаются сверкающие нити, и все дробится на миллионы осколков...

Господи! Это же стек! Оно обходит стек! Вот значит, как это выглядит! Когда я исчерпываю глубину, она видимо с треском цепляется и разрушается, ломая свои ажурные конструкции...

В каком-то месте я замечаю скамейку с какой-то одинокой старушкой на ней. Мне нужно тоже посидеть... Напротив стоит продавщица цветов, приплясывая от холода. Какая-то неприятная, чуть не мерзкая. Да и старуха чего-то уставилась на меня? Кто она? Чего сидит здесь? Небось, от нечего делать. Это они вечно толпятся в магазине, и из-за них ничего не успеваешь купить. И визжат в троллейбусе, чтобы им уступили место. Небось, смотрит эти идиотские новомодные сериалы, как их там? Мария? И ещё кто там плачет? Да что они все понимают!? Кто это может понять, сколько знаний и труда надо, что бы выстроить такое? Сколько бессонных ночей надо провести? Сколько читать? Причем ежедневно, и вовсе не идиотский роман о любви и дружбе?

Да кто вообще в состоянии это понять!? Эти новоявленные пижоны, называющие себя программистами? Коряво пишущие на фокспрах<sup>3</sup>, клипперах<sup>4</sup> и бейсиках? И везде задающие вопросы: а скажите, какую команду мне надо набрать? А какой хелп почитать? А когда мануал на русский переведут? А этот их так называемый “софт”? Великие стотысячевариантные вечном живые склады и бухгалтерии? Нетленные творения. Все на одно лицо. Если там и есть различия, так в корявости и глючности кода. Глюк на глюке сидит и багом погоняет... Вот заставить бы их высекать их коды в камне, как древние камнетесы, чтобы хоть немного задумались о том, чего пишут... Или эти технокрысы? Это ж надо — вирусы писать, чтобы, значит, гадостью людям сделанной прославиться! Тьфу!

Нет. Не хочу я сидеть на этой скамейке, в компании с этой старушкой. Куда же я шёл? Вспомнил. К другу я иду. Может, подскажет чего? Проклятый байт! Чего я взъелся то так? Все своим делом занимаются, с чего бы худшим, чем я? Талантливых людей хватает везде. Что-то сильно меня, видать, припекло...

Приятель, открыв дверь, молча смотрит на меня.

— Ну?

— Что “ну”?

— Скажи мне только одно? — мрачно говорю я, проходя в квартиру, — ты можешь дать мне *один байт*? Всего один. Я готов отдать тебе за него все что угодно. Мне байта не хватает, понимаешь? Ну не влезаю я в кристалл!

Друг какое-то время молчит.

— Я смотрел твои коды.

— И что? Нашел, что-нибудь?

— Нет, — тихо говорит он и, помолчав, продолжил — безукоризненно связанное кружево. Ни единой петли. Стыков не видно. Филигранная работа. Штучная. Прям лепота. На каждую строчку можно поставить знак качества. И высечь в мраморе. И однородно все, ни уплотнений, ни пустот. Монолит, но эластичный. Но...

— Что “но”? Да не тяни ты kota за хвост! Не мотай душу! И без тебя тошно! — взрываюсь я.

---

<sup>3</sup> Искаженное Foxpro — система управления базами данных. — Прим. ред.

<sup>4</sup> Clipper — система управления базами данных. — Прим. ред.

— Большинство мест я не могу понять... Не понимаю... Только вижу, как это красиво. Неосвязаемо как-то. Не ухватить сознанием... Вроде вот-вот, но оно улетает... Это как снежинка, когда пытаешься взять ее в руку. Или как звуки еще непонятой, но уже осознаваемой музыки...

— Что за чушь ты несешь!? Какая к черту снежинка?! Какая музыка!? Ты можешь мне помочь или нет!?, — грохочу я. — Ты же друг мне. Помоги, а? Ты только скажи, куда мне втиснуть этот байт, — я с надеждой смотрю на него. — Ведь у тебя свежий, незамыленный глаз. Ведь один, всего один оператор, без которого можно обойтись и все! Всё! Проблема решена, понимаешь? И я тебе по гроб жизни, навсегда, — шепчу я, ухватив его за рукав, — ведь там же до черта строк, а я же просто человек, ошибся, пропустил, не заметил... А? А мне ничего не надо. Ни славы, ни похвал. Я только хочу, чтобы оно улетело. Чтобы отпустило меня... Ну нет у меня сил больше. Ну пойдем, давай ты еще коды помотришь.

— Откажись, — говорит он.

— Отказаться? От чего? — не понимаю я.

— Возьми другой кристалл.

— Ты сошел с ума!?!? Как это — другой!?!? Кучи наделанных плат, монтажники, наладчики, зарплаты, детали!? Это что — шуточки!?!? Ты думаешь это на компе — хочу — пару строк добавил, и никто не заметит!?!? Затрат ноль, а добавляй хоть мегабайты кода!? А люди? Они ведь верят мне! Я ведь сказал — “Да!”. Я ведь согласился, хотя и видел, что запаса ресурса почти по нулям будет! А у них уже готово все! Корпуса, металл, питание. Они ждут только кода! Ты знаешь к чему может привести смена кристалла? Ты знаешь, сколько будет стоить этот байт! Один байт! — Я сам оглох от своего крика.

— Дурак — слово шлепком падает на землю, как тюк мокрого белья. — Отступись! Забудь! Мы с ребятами уже три дня, как смотрим эти коды. Я собрал всех, кого мог. Я сразу понял, что с тобой неладное. У тебя НЕТ ошибок! Ни одной. Более того, мы не понимаем, как тебе вообще удалось *это* запихнуть.

Колени мои подогнулись, и я то ли сел, то ли сполз на стул. Я глубине души я знал это. А потом я стал говорить. Это был странный монолог. Как будто кто-то кричал, шептал и снова орал моим голосом: “...думал все это время. Я понял, что не так уж важен этот проект, как мерило моей ответственности. Ну не решится он, ну переделается там как-то всё. Черт с ним! Не так уж это важно. Позор там мой. Дело в другом. Во мне. Ты знаешь, что я уже очень давно не раб, который делает, то что скажут, оправдывая это зарабатываем денег для семьи. Очень долгое время я наемник. Мои услуги, мои программы стоят очень дорого. Ты же знаешь, я не берусь за простые вещи. Пусть хоть озолотят. Я на себя и *для себя* работаю. Да мне уже давно плевать на деньги! Они практически не имеют надо мной власти! Мне другое надо! И я тщательно скрываю это. Потому, что интересную задачу я готов делать бесплатно, а то ещё и приплачу за нее. Ты же сам знаешь, каково это!? Ну?! Ты ведь тоже не для денег это делаешь! Есть только одна вещь, которую я никогда и никому не говорил. Я, когда делаю, — лечу. И не ври мне, что не знаешь, что это! Все знают это! Только предпочитают не помнить или не верить! Тебе тоже знакомо это! Это как в детских снах. Помнишь? Мы взмываем высоко, высоко и несемся, визжа от переполняющего восторга! А под нами проносятся леса, горы и моря! Ты думаешь, это был сон!? Нет! И я давно понял это! Только сказать боялся. Стыдился, дурак! Но теперь мне все равно! Это душа наша летит! По- настоящему! А разум говорит, что мы спим. Понимаешь? А почему, когда мы вырастем, перестаем летать? А!? Почему?! Почему нам перестает сниться этот сон? Не знаешь? А я знаю! Потому, что душа наша тяжелеет, потому, что ценности, деньги, условности этого общества захватывают над нами власть и душа наша больше не в силах поднять этот груз! Как же! Мы ж прям, распластываемся, чтобы стать ковриком, о который вытрут ноги! О, какие веские причины, такие аргументированные объяснения, почему это было необходимо именно так прогнуться и как мы это ради кого-то это делаем. Мы врём сами себе каждый день, убеждая сами себя, что живём правильно. А я не хочу, не могу больше врать! Ты не понимаешь, как это относится к этому дьявольскому байту!? Все очень просто. Я уже давно могу летать! И работа помогает мне в этом. Да-да! В

пике высшего напряжения при решении своих задач я взлетаю. Это невозможно описать! Но я не могу лететь постоянно. Я снова опускаюсь. И так до нового кода, в который надо что-то вложить. Я не знаю, что. Какой-то кусок себя, что ли. Но в этот раз я попался. Меня сгубила гордыня. Ну как же! Я ведь гуру, умеющий снисходительно тыкать чайников носом и походя разрешать их проблемы! Мне так нравятся их взгляды на меня как на божество. Ведь мы тщеславны, и я не исключение. Но сейчас все не так! Ты думаешь, проблема в этом одном байте и как его засунуть? Нет! Я не могу его запихнуть! Но это может сделать *не я!* Понимаешь? Решение есть! Я это чувствую! Только я не способен его найти! Для этого я должен стать другим! Не собой! И кто-то или что-то чётко поймал меня, на этот *один байт!* Ты же отлично знаешь, что я умный и хитрый! Если бы задача не решалась, я бы ушёл, ускользнул, сорвался с крючка! Но я считал, что она решается, и меня подсекли! Поймали на этот байт, как в сеть. И байт этот, это размер ячейки сетки, через который я не могу улизнуть. Слишком далеко я зашёл... И я не смогу уйти и снова быть свободным, если не изменю что-то в себе! Полностью изменить себя, понимаешь? Стать другим человеком! И тогда, может быть, передо мной откроется дверь... Я не знаю куда. Я не знаю, что за ней... И я не знаю, как и что я должен сделать для этого... Да... И еще цена... Я и это понял. Я не смогу быть как прежде. Я не смогу летать больше. Все будет кончено”.

Я медленно поднялся и, ссутулившись, пошел к двери.

— Прощай, — глухо сказал я в пустоту.

— Кретин! — неслось мне вслед. — Ты же сдохнешь над этой программой! Сдохнешь! Ты в зеркало на себя посмотри! Психушка для тебя — милость! Делай что-нибудь! Делай! Сходи к своей девушке, поговори с ней, сходи в кино, отвлекись.

Но я уже ничего не слышу. “Господи, если ты есть, — помоги”. Только на улице я спохватываюсь, что забыл перчатки и шапку. А зачем они мне? Разве это главное? А что главное? Зачем все? Кому все это нужно? Людям? Да наплевать им! Это *мне* нужно! Лично мне! Я сам загнал себя в ловушку и сам же не могу из неё выбраться. Что это? Наказание? Урок, чтобы впредь не задавался? Да уж, скорее так. Гонору у меня хоть отбавляй. Стоп, стоп. Как он сказал? К девушке?

...В офисе тепло и уютно. Калорифер. Чистенько. Жужжат компы. Папочки, стоечки. Девочки поят меня кофе, подкладывают булочки, которые я пожираю с жадностью, перемазавшись в шоколаде. Они подливают и подкладывают, сердобольно глядя на своего опустившегося коллегу. Девочки тоже программистки, чего-то там офисное набивают, на радость кадрам и бухгалтерии. Они аккуратны, при макияжах, отлично, словом, выглядят, особенно по контрасту с моей многодневной щетиной (а может уже и бородой?). Я кратко и с неохотой отвечаю на вопросы, что, мол, меня до жизни такой довело. “Не решается. Ассемблер. Со стеком проблемы. Байта не хватает”. Одна из них, Оксана, кажется, её зовут, говорит с украинским приятно-округлым выговором: “Який такий стек? Зачем он тебе нужен? Мы с Олей, она на Клиппере, я на Фоксе, никакого стека у нас нет. Может, и тебе не надо? Вечно вы, мужики, себе пакость какую выдумаете. Сами же и мучаетесь, да нас мучаете, скажи, Оля?”.

— Что?! — Только кресло мешает мне свалиться на пол. Какой-то противный, каркающий клетот рвется из меня: “Вы пишете *без стека!*?”

А вот это уже истерика.

Я снова на улице. Милые, милые наши дамы! Как вы приятны в вашем неведении. И как это здорово, что вы этого не знаете. Вам и не нужно это знать. Сходить с ума от нерешенных задач, как и философских вопросов, — это привилегия мужчин. Конечно, бывают и исключения. Но они скорее подтверждают правила. Если в высокоуровневых языках дамы ещё попадают, к сям и ассемблерам практически исчезают, то в нашей области я не слышал о них вообще. И это правильно! Нечего валить на женщин ещё и эти проблемы.

А ноги несут меня куда-то, мысли текут сами по себе. Мне они неинтересны, я человек конченный. Я не смогу с этим жить. Буду влачить существование, все равно кем, но уж к компьютерам этим, на пушечный выстрел не подойду, это уж точно. Поделом. Нечего было

строить из себя крутого. А эта моя снисходительность сноба? Мол, все знаю, все мне по плечу. М-да. Доигрался, козел? Я бреду в этом абсолютно чужом для меня мире, в котором ни одна живая душа не в состоянии меня понять. Друг и тот не смог. И никому нет до меня дела.

Но что это? Я здесь вроде был? Цветы. Скамейка. И бабушка на ней. Как будто и не уходила. Ведь мороз же? В нерешительности я присаживаюсь на край скамейки. Ого! Руки-то замерзли и уши тоже. И тут, как будто что-то толкнуло меня, я встал, подошел к продавщице цветов и на последнюю трешку (а зачем она мне?) купил розы и подошел к бабушке. В голове у меня судорожно билась мысль: что я делаю? Зачем? Она ждала, подняв ко мне лицо. И я выдал из себя: “Извините. Я могу подарить Вам цветы? Я плохо подумал о Вас тогда”.

Она нисколько не удивилась. И сказала: “Где же ты был так долго, сынок? Я замерзла ждать тебя”.

Сказать, что я удивился, значит не сказать ничего. Я был потрясён, ошеломлен, раздавлен! А она продолжала: “Тебе ведь плохо, сынок?”. Она смотрела на меня с участием. В ее глазах светилась мудрость, доброта и... любовь. Вы понимаете!? Ко мне любовь. И тогда я сказал: “Да! Мне плохо. Мне очень плохо”. Я не боялся и не стыдился. Что-то как будто упало с меня, отскочив, как шелуха. И я стал рассказывать. Сбивчиво, торопясь и захлебываясь.

Я рассказывал бабушке, как я программирую однокристалльные микропроцессоры. Она внимательно, не перебивая, слушала меня. Она всё понимала! Каждое мое слово! Это я видел по её глазам. Я говорил и говорил. А она вела меня куда-то, и я ел, что-то очень вкусное, а потом мы пили чай с каким-то необыкновенным вареньем на крохотной, но такой уютной кухне. Наверное, это была очень странная картина. Полусумасшедший программист и старушка, его внимательно слушающая. А потом говорила она. Я не помню, о чём. Я только помню, что это было очень важное и нужное мне, что я черпал из этой кладези мудрости, которую можно обрести, только потеряв столь много, но обретя любовь.

И вдруг. Снова стала разворачиваться внутри меня странная, невесомая и в то же время прочная конструкция. Она разворачивалась мощно и грациозно, окруженная великолепием огней. Каждая ее грань, каждый элемент были совершенны и неповторимы! И легонько вибрируя, она порождала музыку. И всё это вместе наполняло меня необычайным трепетом и восторгом! Это я! Я создал её! Это мной отшлифованы все её грани! Ну почему этого никто не видит!? Ну посмотрите же! Разделите со мной мое счастье! Теперь я не боялся. Я *знал*, что она полетит! И она поможет и мне оторваться от земли. С ней и я полечу к звездам!

И я снова шел по улице. Но совсем по другой. А точнее — просто в другом мире. Потому что этот был прекрасен! Снег брызгал разноцветными искрами тысяч неповторимых красок и такой неповторимой музыкой звучал под ногами. Это как будто ваш старенький компьютер с CGA монитором, вдруг стал показывать миллионы цветов. Впрочем, что за чушь я несу? Это много, много лучше.

Создавайте свои корабли. И пусть они путешествуют в необычайных мирах. Я был неправ. Каждый из нас может путешествовать и жить в этих мирах. И это не важно, как и где вы их создаете. Мы создаем программы в той же мере, в какой они создают нас. И настоящие программы создают не хитростью, и для этого мало разума знаний и смекалки. Они должны пройти через сердце, потому что являются порождением нашей любви.

Потому, что программы, которые мы создаем, чистый продукт творчества. Именно поэтому они столь привлекательны. Не нужно ни молотка, ни зубила, ни кистей и красок, чтобы выразить в ней главное — себя! И это не важно, что, может быть, понять красоту ваших кодов сможет не так уж и много людей. Если, написав свою программу, мы стали лучше, то это правильная и хорошая программа! Но если вы думаете, что нужно меньше труда, то вы ошибаетесь. И если вы не готовы или не хотите в своем творении оставить часть своей души и любви, не готовы к тому, чтобы изменить себя, то лучше не пишите программ. Поищите себя, в чем-нибудь другом...

**Послесловие автора.** Прекрасно! Это и есть настоящая ЖИЗНЬ. Безудержный восторг, охватывающий тебя в момент, когда твое творение оживает, можно сравнить только с умением

летать. Эмоции и чувства бьют через край. Ты уже не можешь их больше в себе сдерживать. Ты пытаешься поделиться с другими этим счастьем, но наталкиваешься на стену непонимания, раздражения. В такие минуты хочется забыть их всех, отрешиться от этого жестокого, мертвого мира, закрыться в комнате, наедине со своим компьютером, чтобы продолжить жить в своем прекрасном идеальном мире. Это, как наркотик, притягивает, и ты не можешь отказаться. Ты начинаешь агрессивно нападать на тех, кто пытается посягнуть на твоё счастье. Временами кажется, что ты одинок, остался один во всем этом, непонимающем тебя, мире. В такие моменты спасает одно — надежда на то, что всё же есть небольшая горстка людей, способных понять тебя, таких же, как и ты, — низкоуровневых программистов.

**Послесловие редакции.** Конечно, описанная история — это не только о программировании. Она вообще — о профессиональном мастерстве и о стремлении к совершенству. Об автомобильном мастере, который на слух может определить неисправность двигателя, о компьютерщике, который по телефону консультирует клиента и, так сказать, “руками того” исправляет неисправность, о художнике, способном нарисовать такую картину, что кажется нарисованное реальным...

.....

*Приведенная ниже статья была опубликована в 2008 году в газете “Троицкий вариант. Наука”. Её автор — Михаил Владимирович Донской (1948–2009) — известный российский учёный, программист, заведующий лабораторией Института системного анализа РАН, член Российской академии Интернета, автор шахматной программы «Каисса» (первый чемпион мира среди шахматных программ), создатель и глава информационно-технологической компании «ДИСКО».*

## **Жизненный цикл программиста**

У каждой профессии есть свой романтический период и есть период, когда она превращается в рутинную. Быть шофером в начале прошлого века было трудно и почетно. Сегодня автомобиль может водить любой желающий, а в большинстве районов США жизнь без автомобиля практически невозможна. Так профессия шофера прошла полный цикл от интеллектуальной и романтической до бытовой и повседневной за какие-то 60 лет. Цикл профессии авиационного пилота тоже близится к окончанию и займет те же 60 лет. Но время ускоряется, и новые профессии имеют гораздо более короткий цикл. Особенно это верно по отношению к профессиям, связанным с информационными технологиями.

Так получилось, что время моей жизни практически совпало с жизненным циклом моей профессии. Я — программист. Сами компьютеры появились в 40-х годах, т.е. в то же десятилетие, когда я родился. Оглядывая свою профессиональную жизнь, хочу напомнить, как менялась профессия программиста. Когда я школьником учился программировать на М-20, в СССР программистами были известные математики, на ходу выдумывавшие то, чему сейчас учат в школе. В группе программистов ИТЭФ, где для вычислительных работ ядерной физики стояла эта самая М-20, придумали массивы, списки, необходимость использования подпрограмм и многое другое. Один из моих учителей, Г.М. Адельсон-Вельский, придумал хэш-память. Подробности можно найти в книге другого моего учителя А.С. Кронрода — «Беседы о программировании». Еще до Э. Дейкстры основные принципы структурного программирования были изложены А.Л. Брудно в книге «Программирование в содержательных обозначениях». Там же была создана первая шахматная программа. А ведь в то время программировали в кодах, память под программы и переменные распределяли своими руками, и известны случаи, когда на одно и то же место грузились разные подпрограммы и всегда работала только последняя. Всерьез была распространена так называемая «польская игра», когда надо было уложить заданный алгоритм в минимальное число ячеек памяти.



В итоге тогда шахматная программа ИТЭФ, предшественница «Каиссы», умещалась в памяти М-20, а именно в 4096 ячейках, каждая из которых имела 48 разрядов (теперь это называют битами). Где-то рядом уже существовал Алгол-60, но им «настоящие» программисты не пользовались, поскольку техники отладки практически не было. Чуть позже большую популярность получила статья «Почему настоящие программисты не пишут на Фортране».

Мои студенческие годы пришлись на целый ряд советских машин — Раздан-3, Минск 1, 2, 22, 32, Урал-14, все они имели пульт, за которым сидели программисты, а программы и данные вводились с перфокарт/перфолент. АЦПУ — устройство «широкой» печати — появилось только в конце 1960-х. Для того чтобы быстрее писать программы для этих машин, мы сами разрабатывали операционные системы. Тут уже требовалась высокая техника программирования, поскольку эффективность операционной системы была необходима для самой возможности ее использования.

Рассказывают, что в операционной системе «Пульт», написанной в Вычислительном центре АН СССР для БЭСМ-6, был счетчик ошибок оператора, и при достижении некоторого порога система выдавала «вежливое» сообщение: «А если ты — дурак, то не садись за “Пульт”». Когда директор ВЦ академик Дородницын инспектировал систему, он понажимал несколько раз случайные кнопки и был крайне огорчен полученным результатом.

О серьезности задач, которые приходилось решать на тогдашних компьютерах, говорит то, что одним из моих проектов в студенческое время была система инверсного поиска патентов для экспертов. Кстати, ВМК еще не было, было отделение вычислительной математики на мехмате, но я учился на отделении математики. Сдавая зачет по программированию, я должен был апеллировать к своему профессору М.Р. Шуре-Буре, поскольку его аспиранты, принимавшие зачет, программировать почему-то не умели. И, вообще, на мехмате программирование считалось чем-то вроде предательства чистой математики, и всерьез на моем курсе им занималось не больше десятка человек. Была даже частушка: «Меня милый не целует, не садится близко, говорит “я — математик, а ты — программистка”». А потом 90% выпускников с моего курса пошло-таки работать программистами.

Мне посчастливилось заниматься в семинаре по эффективным алгоритмам, на котором моими сокурсниками было придумано несколько классических алгоритмов. М. Кронрод построил оптимальный алгоритм упорядочения, Е. Диниц и А. Карзанов создали целую серию алгоритмов по потокам в сетях. А. Карзанов потом стал автором классических работ по линейному программированию. Мой диплом представлял оптимальный алгоритм решения задачи о назначении и состоял из полутора страниц. Конец моих студенческих времен совпал с революцией в компьютерах. Появились компьютеры «общего пользования» с системами разделения времени. Это IBM 360, ICL 4-70, ЕС ЭВМ. Писать в кодах для таких машин стало принципиально невозможно, и на передний план вышел (как наименьшее зло) язык ассемблера. Были и другие языки программирования (Фортран, Кобол, Алгол, PL-1), но они не позволяли эффективно контролировать оттранслированный код. Сосед по кабинету в ИПУ М. Фурман на мой изумленный вопрос, как ему удастся программировать на PL-1, просто заметил, что он в уме транслирует все операторы, прежде чем написать их.

За 15 лет работы с ассемблером мы общими усилиями овладели им так, что он стал языком более высокого уровня, чем все вышеперечисленные. Под термином «овладеть языком» я имею в виду не то, что мы досконально знали его синтаксис и семантику, а то, что были наработаны библиотеки подпрограмм, приемы программирования, идиомы и многие специфические приемы, так что программы писались легко и свободно. И, главное, еще легче отлаживались и адаптировались. Те, кто писал на Фортране, оценят последние свойства.

Именно за эти годы мною и моими товарищами под руководством В. Арлазарова были написаны «Каисса», «ИНЕС», АСУ МНТС (Международного научно-технического сотрудничества для ГКНТ СССР) и много конкретных прикладных систем. Где-то в это время нам пришлось расстаться с привычными перфокартами и пересест за дисплеи, между прочим

— алфавитно-цифровые. Сделанная в ИПУ «Каисса» стала первым чемпионом мира среди шахматных программ. Кроме удовлетворения амбиций она принесла еще много друзей по всему миру, поскольку в те времена создание хорошей шахматной программы было делом сложным, и сформировался своего рода теневой клуб авторов и знатоков шахматных программ. Среди них были знаменитые в мире информационных технологий люди — Клод Шеннон (автор теории информации), К. Томпсон (автор операционной системы Юникс), Д. Леви, М. Ньюборн, А. Марсланд, Б. Миттман, Ф. Фридель (автор ChessBase) и многие другие.

СУБД «ИНЕС», в которой я занимался системными вопросами — генерацией и дистрибуцией системы, системой поддержки версий, для чего была написана Архивная Система, — и АСУ МНТС, устанавливая которую мне пришлось по всем министерствам и республикам СССР, принесли мне много хороших знакомых по всей стране. В любой город СССР можно было поехать, и везде встречали очень тепло, даже когда устанавливаемые мною системы были принимающим, мягко говоря, не слишком нужны (как сейчас сказали бы, АСУ МНТС снижало коррупционную емкость планирования научных командировок за границу).

И мое тогдашнее хобби — спортивный бридж — тоже было источником многих дружб и знакомств. Не случайно, когда мои американские друзья приезжали в СССР, они после очередной случайной встречи с кем-нибудь на улице спрашивали меня: «Тебя все здесь знают?».

С Клодом Шенноном связана одна из самых удивительных историй в моей жизни. Меня с ним познакомили в 1980 г. на чемпионате мира среди шахматных программ в Линце. Каждый чемпионат имеет своего почетного гостя, и в том году им был Клод. Услышав его имя, я подумал: «Как! Он еще жив?». Ведь работы Шеннона по шахматному программированию относились к году моего рождения, т.е. для меня он существовал в очень давней ретроспективе. Оказалось, что ему в год моего рождения было меньше тридцати, и в 80-м он был еще очень нестарым человеком. Когда же пришла моя очередь быть почетным гостем чемпионата мира 1999 г. в Падерборне, я прочел в глазах молодых шахматных программистов все тот же немой вопрос: «Как! Он еще жив?». И, поняв, что с момента моих публикаций уже прошло больше 20 лет, я вспомнил Шеннона и успокоился.

В начале 1970-х появились машины серии «Ряд». Так получилось, что во время моего распределения после МГУ мне пришлось быть свидетелем, как А.С. Кронрод боролся за продолжение проектирования и производства оригинальных советских машин (он даже предлагал назвать серию «АС» — автоматическая советская — по своим инициалам) против В.М. Глушкова и Л.Т. Кузина, которые ратовали за копирование IBM. Одним из аргументов у последних было то, что можно будет воспользоваться всем математическим обеспечением, созданным для IBM, и ликвидировать небольшое отставание в вычислительной технике, которое имелось в конце 1960-х.

Глушков и Кузин победили (а судьей был председатель ГКНТ Кириллин), но всё оказалось не так-то просто. Первый компьютер серии был запущен титаническим трудом инженеров-электронщиков в жаркое лето 1972 года на ВДНХ (после чего они искупались в фонтане Дружбы народов), а массовая работа на нем началась только в 1979 году. Все это время я неплохо зарабатывал лекциями по ОС ЕС ЭВМ. Документация по системе переводилась моими однокурсниками и другими людьми, не представлявшими себе, что такое компьютер вообще и операционная система в частности, и разобраться по такой документации было невозможно.

Глушков и Кузин просчитались именно в этой компоненте — культуре пользования. Теперь я понимаю, что неправ был и Кронрод, за которого я «болел», потому что надо было и копировать IBM, и делать свои машины именно для сохранения культуры. А в итоге к 80-м мы потеряли культуру проектирования элементов, потом и культуру проектирования устройств, а сейчас от нас уходит (вместе с носителями — людьми, которые умеют это делать) культура создания операционных систем. В итоге вместо того, чтобы догнать кого-то, мы

отстали в этих компонентах навсегда. И, повторюсь, не потому, что нет нужных производств или знаний, а потому, что почти не осталось людей, которые это умеют делать.

А в 80-х началась эра языка Си на машинах, скопированных с PDP и IBM PC. Мы потеряли весь свой ассемблерный «языковый запас» и так и не достигли аналогичного уровня инструментария на Си. Это была своего рода эмиграция. Привыкнув к детальному пониманию, как происходят реальные вычисления в памяти, пришлось отвыкать и работать в гораздо более абстрактных сущностях. Зато остался интерес к базовым понятиям программирования, выходящим за пределы конкретных языков, операционных систем и устройств. Как любят говорить мои сотрудники, «в конце концов в компьютере биты бегают».

Одним из важнейших принципов программирования для меня является необходимость создания собственного инструментария для любого проекта, начиная с проектов средней величины. Это мы красиво называли созданием специализированного (для проекта) языка программирования, включавшего специфичные структуры и классы данных и базовые процедуры. Крупные проекты требуют параллельной разработки специфических (для проекта) отладочных средств. Все это не исключает, а даже предполагает создание абстрактных библиотек, а также инструментальных и отладочных средств, используемых во всех проектах. В сущности, только последнее и делает ремесло программиста или бизнес-фирмы по разработке программ прибыльным делом.

Кроме того, есть общие методические принципы создания программ, не вполне осознаваемые даже хорошими программистами. Примером может служить проблема принадлежности объектов друг другу, а также совмещение двух структур любой программы - иерархии вызова подпрограмм вместе с объектами, принадлежащими подпрограммам, и иерархии объектов по принадлежности друг другу. Примером отказа от самой идеи справиться с этими проблемами является наличие в некоторых языках механизма сборки мусора, что является молчаливым признанием возможности присутствия в среде объектов, не принадлежащих ни подпрограммам, ни другим объектам. С другой стороны, система счетчиков использования объектов давно известна, но применяется она в основном только для объектов, которые могут принадлежать нескольким владельцам.

С течением времени программирование из тонкого ремесла, иногда восходящего к искусству, становилось ремеслом все более рутинным. Если до середины 80-х еще реальны были программы, созданные если не одним человеком, то хотя бы в рамках одного коллектива, то в дальнейшем в производство шли программы, построенные по принципу «Лего», а именно — собранные из различных полуфабрикатов (библиотек и компонент), разработанных в разных уголках мира.

Как ни странно, это сделало ценность программистов с хорошим математическим (не скажу образованием, а подходом) гораздо выше. Их стали называть по-разному — системными аналитиками, руководителями проектов, системными архитекторами. И наряду с программистами, умевшими «выполнить проект» — реализовать конкретное техническое задание, — потребовались именно такие «абстрактные» специалисты, умевшие совсем другое. А именно, разбить процесс создания большой системы на проекты, выбрать для них инструментарий, подобрать исполнителей, суметь их проконтролировать и, в конечном счете, обеспечить работоспособность созданной системы. И сегодня таких специалистов приблизительно столько же, сколько было программистов в начале моего трудового пути.

Только просьба не путать системных архитекторов и системных администраторов. Эти две почетные профессии не имеют практически ничего общего. Более того, мой короткий опыт работы, близкой к системному администрированию, показал мою полную профнепригодность в этой области. С другой стороны, мне неоднократно удавалось проектировать и внедрять большие системы. Кстати, проблема выбора инструментария для создания большой системы является одной из ключевых. В прежние времена стандарты на компоненты и их описания были столь слабыми, что проще было написать свое, чем пользоваться чужим. Сейчас усилиями крупных фирм по созданию программного обеспечения, в первую очередь усилиями Microsoft, этот процесс стандартизовался, и пользоваться чужими компонентами стало легче.

Другой вопрос, что большинство доступных компонент далеки от идеала, но плохой стандарт все равно лучше, чем его отсутствие.

Вернемся к 1980-м. Еще до перестройки мы — отдел ВНИИСИ под руководством В. Арлазарова — локально победили институтскую и академическую бюрократию за счет того, что на игольчатом принтере «Электроники» смогли изобразить шрифт печатной машинки. В то время, например, было запрещено подавать к защите диссертации, напечатанные на компьютере, но с нашим шрифтом понять, что это — печать компьютера, а не машинки, без специальной экспертизы было нельзя. Аналогичным образом дело обстояло со многими другими документами — планами, отчетами, выездными характеристиками и т.д.

До создания этого шрифта любой бюрократ находил повод придаться к документу из нескольких страниц и требовал его перепечатки, получая передышку на пару часов, а то и дней. Но после создания шрифта исправленный документ ложился ему на стол через пять минут, и он понимал, что ищет работу не мне, а себе. Тут-то в документе все волшебным образом становилось нормальным. Что он при этом думал о моей квалификации как машинистки, остается тайной. Как известно, персональные компьютеры победили Советский Союз (не только вышеописанным способом, а, главным образом, отменой монополии на информацию и разрушением барьера между безличными и личными деньгами).

В начале российской эпохи персональных компьютеров, случайно или не случайно совпавшей с кооперативным движением, ко мне обратился прекрасный менеджер Е. Соколинский, возглавлявший кооператив «Перспектива», с предложением реанимировать «Каиссу» для ПК. Для этого мне нужно было из работавшего в свое удовольствие ученого стать начальником группы программистов, да еще и создать эту группу с нуля. Уговорив меня, Соколинский нашел изумительный способ формирования группы. Мы дали объявление в газеты о платных курсах шахматного программирования. Стоимость месячного обучения для наших слушателей составляла 200 рублей, что по тем временам было существенной суммой. Занятия шли шесть дней в неделю, и кооператив доплачивал за аренду аудиторий и компьютеров немалую сумму. Из десяти слушателей, которых мы тщательно отобрали, только один человек пропустил одно занятие, потому что у него в этот день был выпускной из Физтеха. Потом мы всей группой перешли в СП «Параграф».

В конце 1980-х, когда я оказался в СП «Параграф», он представлял собой своеобразную сборную лучших московских программистов. В «Параграфе» того времени работали Е. Веселов (автор «Мастера» и «Лексикона»), А. Чижев (автор многих русификаторов, в частности знаменитой «Беты», он же автор альтернативной таблицы кодировки кириллицы) и др. В качестве помощницы у Веселова в «Параграфе» работала О. Дергунова, получившая известность уже в Майкрософте. Игры продавал В. Савюк, потом раскрутивший марку «Денди». В общем компания подобралась неплохая.

По дороге пришлось пережить очередной крутой поворот — появилась Windows 3.1, и пришлось от традиционного процедурного программирования переходить к системам, управляемым потоком событий. Сегодня они привычны и понятны, а тогда ушло много усилий на понимание, «куда лошадь запрягать», а именно как устроен порядок исполнения кода в таких системах. Поток управления в них весьма неочевиден, и проблемы многопоточности и синхронизации вышли на первый план.

У меня в «Параграфе» был отдел шахматного программирования, в котором «Каисса» получила вторую жизнь в качестве программы для IBM PC. Хотя мы и сделали в отделе шахматную программу — реинкарнацию «Каиссы» для IBM PC, которая достойно сыграла на компьютерной олимпиаде 1990 года, заняв третье место, интерес быстро сдвинулся в сторону пользовательского интерфейса, поскольку графические оболочки Мака и Windows очень манили в эту сторону. Наш отдел, в котором работали А. Дубец, М. Караев, В. Кокин, И. Шабалин и другие, открыл целое направление графических редакторов. Мы сделали редактор формул, а уже уйдя из «Параграфа», и редактор факсов, а потом и новую версию «Лексикона». Оказалось, что общее всех этих редакторов — разбиение на три составляющих: данные, их отображение и собственно редактор, преобразующий данные согласно действиям

пользователя, — является фундаментальным для систем пользовательского интерфейса. Недаром операционная система Symbian базируется на таком разбиении.

В это же время пришлось осваивать C++. Мое знакомство с этим языком началось с экскурсии в офис Bell Laboratories, которую мне устроил в 1989 г. автор Юникса Кен Томпсон. Мы с сыном жили у Кена в гостях, и в воскресный вечер он предложил прокатиться в офис. Офис был безлюден, и я с интересом смотрел на технические чудеса, которых там хватало. В какой-то момент Кен показал на дверь кабинета со словами: «А здесь сидит чудак, который думает, что на его языке будет программировать весь мир». табличка на кабинете гласила, естественно, «Б.Страутсруп». Потом пришлось-таки учиться программировать на C++. Язык очень коварен. На нем должны программировать либо начинающие программисты, которым важно быстро получить результат любыми средствами, либо очень опытные. Создание больших систем на C++ программистами среднего класса может приводить к самым печальным последствиям. Однажды в книжном магазине Стэнфордского университета я видел книжку по C++, напомиравшую сборник кроссвордов. там приводилось множество выражений на C++, выглядевших очень естественно, но транслировавшихся в умопомрачительный набор команд. Зато C++ позволил вернуться к эффективному созданию инструментальных средств. Наборы идиом, библиотечных классов, правила пользования — все это стало багажом наших программистов, сделав их работу более легкой и приятной.

После ухода из «Параграфа» я не смог найти другую работу, в основном по принципу «двух медведей в одной берлоге», когда начальник не хотел иметь в команде еще одного лидера. Поэтому в 1994 г. мне пришлось заняться бизнесом, организовав свою фирму «ДИСКО» (Donskoy Interactive Software Company), существующую по сей день. Фирма занимается разработкой программ на заказ. Основными клиентами являются крупные фирмы, работающие в области информационных технологий. Связано это в первую очередь с тем, что доказывать разумность нашей ценовой политики клиентам из других отраслей крайне сложно. Они искренне полагают, что любую программную систему можно сделать одному человеку за месяц. Ситуация усугубляется тем, что рынок полон дешевых предложений, связанных либо с самонадеянностью вчерашних студентов, либо, что еще хуже, с сознательным затягиванием клиента с целью дальнейшей раскрутки его уж на совсем большие деньги. Это напоминает «бесплатные» лекции по народной медицине, где вход формально свободен, а выход фактически с пустым кошельком.

Фирмы в отрасли информационных технологий гораздо более адекватно оценивают и стоимость работ, и их исполнителей. Рынок наш небольшой, все фирмы на виду, репутации известны. Известны, к сожалению, только внутри отрасли. тем не менее, заказов хватает. До кризиса доткомов «ДИСКО» работало в основном на рынке США, но после него пришлось переориентироваться на российский рынок. Одну вещь после этого перехода пришлось прочувствовать сразу. В Америке ни один менеджер не ведет переговоры вне рамок своей компетенции и, особенно, вне рамок своего бюджета. В России, особенно на первых порах, много раз приходилось, уже придя к соглашению по всем параметрам проекта - техническим требованиям, цене, срокам, — слышать замечательную фразу: «А теперь я пойду согласовывать это с начальством». Эффективность переговоров с такого рода менеджерами, мягко говоря, невелика. Отсюда — нацеленность «ДИСКО» работать с крупными кампаниями, про которые ясно, кто есть кто.

В начале этого тысячелетия пришлось сделать еще один крутой поворот. На этот раз — в сторону мобильных устройств и всего, что с ними связано, в первую очередь беспроводными технологиями связи. Поскольку первые заказы были американскими, приходилось убеждать авторов технологий в их «незрелости» для практического использования. Слышать это от маленькой российской фирмы им было странно. К счастью, это потом подтверждалось и другими, более авторитетными источниками. Так было, например, с технологией BlueTooth, про которую было много критики на CeBit-2002. Мы сделали пилотный проект для разных средств связи по заказу ЗСОМ, и если инфракрасная связь и WiFi работали прекрасно, то с BlueTooth были серьезные проблемы.

Однако с 2004 г. с Bluetooth стало все в порядке, а мобильные устройства становятся все популярнее и популярнее. Хотя карманные компьютеры и сходят на нет, их с успехом заменяют (а может, и вытесняют) смартфоны, имеющие все прелести и карманных компьютеров, и мобильных телефонов. Для лэптопов и ноутбуков сейчас тоже очень хороший сезон. А впереди маячат планшетные компьютеры и сетевые и многое другое.

Весь этот зоопарк мобильных устройств объединяет одно существенное свойство — умение работать вдали от офиса. И тут интересно заметить, как многолетнее желание иметь компьютер всегда на связи с Интернетом входит в противоречие со способом пользования мобильным компьютером. И дело не только в том, что пройдет еще существенное время, пока Интернет будет доступен отовсюду — из самолета, из далеких стран и много еще откуда, где он сейчас недоступен, — но и в том, что инструментальные средства Интернета (браузеры и встроенные в них объекты) не слишком пригодны для многих практических нужд. Например, заполнение в Интернете формы из нескольких страниц, особенно если последующие страницы зависят от полей предыдущих, хотя и возможно, но слишком часто приводит к неудачам как вследствие ошибок заполнения, так и вследствие обрывов связи.

Мы прочно стоим на идеологии сеансовой связи. Она состоит в том, что клиент (мобильный компьютер) связывается с сервером только время от времени, синхронизуя данные в обе стороны, т.е. передавая данные, измененные у клиента, на сервер, в ответ получая данные, измененные на сервере. Изысканные схемы обмена могут строиться так, что на клиента передаются только те данные, которые нужны на ближайшее время. Примером может служить сеанс связи руководителя предприятия перед поездкой к партнеру. Во время этого сеанса можно загрузить на мобильный компьютер всю информацию о взаимоотношениях с этим конкретным партнером — договора, сделки, расчеты — и иметь ее под рукой во время переговоров.

Идеология сеансовой связи воспринимается заказчиками не сразу, но постепенно они оценивают все преимущества такого подхода. Возможность выбора способа доступа к серверным данным, скорость передачи и объем передаваемой информации, удобство и эффективность работы с данными на мобильном устройстве — все это делает идеологию весьма привлекательной. У нее есть один недостаток — нужна предустановка клиента на мобильное устройство, но преимуществ все же больше.

В области мобильных устройств ярко проявилось преимущество Microsoft в подходе к созданию операционных систем над всеми остальными. Десять лет назад самым распространенным мобильным компьютером был Палм. И хотя мобильная версия Windows уже существовала, казалось, что она никогда не сможет быть использована из-за непомерных требований к ресурсам мобильных компьютеров. А Палм был на коне, поскольку для него была специально разработана минималистская операционная система, в которой даже не нашлось места нормальной файловой системе.

Одна беда. Программировать для такой системы было непривычно и крайне непросто. В итоге серьезных программ для Палм так и не было создано, он так и остался еженедельником, а не компьютером. А к 2003 г. мощность карманных устройств доросла до мобильного Windows, и, откуда ни возьмись, масса программистов стала делать большие программы для этой системы. Идеология мобильного Windows была понятна и привычна для программистов Windows для ПК. В итоге операционная система Палм сошла со сцены, и скоро за ней уйдет и само устройство.

Похожая история должна произойти с Symbian, операционной системой, установленной на телефонах Nokia и Sony Ericsson. Подход ее авторов тоже был минималистским. Она, конечно, лучше, чем Палм, но все равно крайне трудна для программистов. А именно программисты решают все. Самой лучшей операционной системой последние 30 лет является Юникс, но плохой пользовательский интерфейс привел к тому, что более популярно изделие Microsoft. Кроме того, программисты, пишущие для Юникса, имеют весьма специфический характер. Их почему-то больше волнует идеологическая чистота системы, чем ее преимущества для пользования. Однажды я работал с «юниксоидом», делавшим серверную

систему для салона игровых автоматов. На все мои требования сделать возможной выдачу статистики игр он отвечал, что это уменьшает безопасность системы. То, что в данном случае гораздо большую опасность представлял собой вульгарный сговор персонала с игроками, против которого и нужна статистика, его не волновало. Видимо, в книгах по Юниксу это нигде не написано.

Сила Microsoft не только в больших деньгах, вкладываемых в раскрутку продуктов, иногда не совсем работающих, а и в армии программистов, умеющих работать в этой системе, и в куче полуфабрикатов, которыми могут пользоваться эти программисты. Для меня, например, разработка программы для Windows по себестоимости вдвое дешевле, чем разработка аналогичной программы для Symbian. Нетрудно догадаться, какую систему я рекомендую своим заказчикам.

Пока последний поворот в моей программистской биографии — видео в Интернете. Интернет, точнее — Мировая Паутина, — это особая тема для разговора. Она обладает врожденным пороком. Это система, придуманная для обмена гипертекстовой информацией в распределенных сетях. Однако за свои 13 лет, начиная с появления «Мозаики», паутина эволюционировала в сторону системы доступа к гигантскому хранилищу информации.

С развитием сетей связи характер информации в паутине стал резко меняться. Если сначала была легко отформатированная текстовая информация, то потом стали внедрять изображения, движущиеся изображения, а в последнее время и видео. Настоящие проблемы начались с того момента, когда потребовалась серьезная интерактивность, в начальный стандарт не заложенная. Поэтому под разными масками в статическую информацию стали добавлять программы. Это могут быть интерактивные объекты, флэш, загружаемые программы, что угодно.

В итоге получился суп из топора. Сегодня принятый как стандарт формат представления информации в Интернете (HTML) является сдерживающим фактором для построения интерактивного контента. Но, как и в случае с левосторонним автомобильным движением, сменить его крайне трудно. Ведь можно потерять накопленную за десятилетие информацию, да и пользователей так быстро на новые браузеры не переведешь.

Поэтому создание порталов и сайтов с видео-контентом является непростой задачей не только с информационной, но и с программистской точки зрения. Фактическое отсутствие стандарта и наличие многих разношерстных инструментальных средств, решающих одну и ту же задачу — доставку видео и его отображение в браузере, — делает эту задачу поистине творческой в самом полном смысле этого слова.

Подводя итоги, я хочу показать, как логика развития информационных технологий изменила характер моей профессии. Говорить о профессии программиста вообще можно, но она столь же не конкретна, как и профессия строителя. Человек, кладущий кирпичи, и человек, создающий большие архитектурные проекты, в равной степени могут называться строителями, но это абсолютно разные профессии.

В моем возрасте класть кирпичи уже неэффективно — не хватает скорости мысли, но, с другой стороны, опыт работы позволяет абстрагироваться от мелочей и рассматривать проблемы с системной точки зрения. Для моих американских коллег такой подход очевиден, здесь же многие считают его верхоглядством.

Я давно считаю само собой разумеющимся, что смогу реализовать любой алгоритм. Я имею довольно большой инструментальный набор и знаю, каким инструментом когда пользоваться. Мне не приходится задумываться над тем, как писать циклы, и т.д. Все это дает возможность, думая над программой, делать это с другого уровня.

Приходящая же в профессию молодежь не имеет такого запаса. И не столько потому, что глупее, а потому, что их не так учат. В моей молодости обучение программированию в институтах было вообще смешным — изучались только синтаксисы разных языков на простейших программах. Сейчас дело обстоит чуть получше, но я не слышал, чтобы во время сдачи курсовой или дипломной работы студенту на ходу меняли техническое задание. А мне в жизни приходилось, сдавая большую систему, с удивлением узнавать об изменении формата

входных данных. Я считаю такую ситуацию нормальной, а молодые программисты — издевательством.

Они не понимают, что, если заказчик меняет требования к уже почти готовой системе, это означает, что система ему нравится. Если система ему не нравится, он вздохнет, заплатит за нее и про нее забудет. Все молодые ребята, приходящие ко мне, обладают одним и тем же недостатком. Они устремлены к тому, чтобы их часть программы заработала как можно быстрее, думая, что это — успешный конец работы. Никто до меня их не научил, что работающая программа — это только начало. Дальше в ходе ее использования будут возникать все новые требования, и программу придется непрерывно менять. Поэтому изначально в нее должна быть заложена эластичность, без которой вносить изменения в программу будет крайне сложно.

Кроме того, инструментальные средства, которые они используют, становятся все более крупными, и мало кто понимает, как эти средства организованы внутри, по каким принципам они работают. Это и не требуется, если нужно только чуть-чуть подстроить такие средства, но при создании больших систем отсутствие такого понимания может вести к большим проблемам, начиная с неэффективности и кончая полной неработоспособностью. А понять внутреннюю организацию сложных систем можно только одним способом — самому сделать что-то подобное, пусть и гораздо более простое. Но я не слышал, чтобы студентам задавали в качестве курсовой работы создание простой операционной системы или системы управления базами данных.

В итоге профессия программиста меняет свой характер. Если раньше программисты знали свою программу досконально, то теперь в лучшем случае они умеют эффективно использовать то или иное инструментальное средство. Появились вообще такие странные на мой вкус термины, как программисты на PHP и HTML.

Я пишу эту статью к своему 60-му дню рождения, возраст пенсионный, и, похоже, кончается не только мой жизненный цикл, но и жизненный цикл той творческой профессии, которой я занимался всю жизнь и которая называлась профессией программиста. Сейчас профессия осталась, но, как и профессия шофёра, она не требует творчества и особых знаний, а только определенных навыков. Программирование из искусства становится ремеслом, и я счастлив, что всю жизнь занимался программированием, пока это было так же интересно и почетно, как пилотировать самолеты во времена Экзюпери.

## Задачник

### Ответы, решения, разъяснения

к заданиям, опубликованным в журнале “Мир информатики”  
в марте 2017 года (выпуск № 10)<sup>5</sup>

#### Задание “Три ребуса”

##### Ответы

Ребус № 1: СКАНЕР. Ребус № 2: ПРОГРАММИСТ. Ребус № 3: ДИСКОВОД.

##### Правильные ответы представили:

— Абдувахидова Алина, Абдувахидова Софья, Базанов Илья, Куклева Лидия, Медведева Анастасия, Серов Георгий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Алиев Тимур, Выскочкин Платон, Жучков Иван, Малова Софья, Родионов Александр и Сластухина Анастасия, детский клуб “Крошка Ру”, г. Александров Владимирской обл., преподаватель **Волков А.Ю.**;

---

<sup>5</sup> Продолжение будет опубликовано в следующем выпуске журнала.



- Баймурзин Руслан и Филиппова Елена, Республика Татарстан, Мамадышский р-н, совхоз “Мамадышский”, политехнический колледж, преподаватель **Порываева Н.С.**;
- Барановская Татьяна, Василенко Сергей и Жукова Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;
- Джазга Павел, Кривонос Роман, Кушкин Никита, Махнач Виктор, Панфёров Антон, Пышко Иван, Рытик Юрий и Семёнов Дмитрий, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;
- Дубовиков Максим, г. Пенза, школа № 73, учитель **Диков А.В.**;
- Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;
- Лазаренко Нина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;
- Удалова Елизавета, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

### Ребусы со звездочками. Часть 7

Ответы

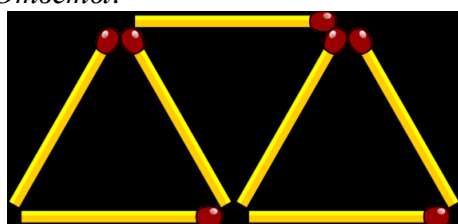
$$\begin{array}{r}
 1. \quad - \quad 1 \quad 0 \quad 1 \quad 0 \\
 \quad \quad \quad 1 \quad 0 \quad 1 \\
 \hline
 \quad \quad \quad 1 \quad 0 \quad 1
 \end{array}
 \quad
 \begin{array}{r}
 2. \quad - \quad 1 \quad 0 \quad 1 \quad 0 \\
 \quad \quad \quad 1 \quad 0 \quad 1 \\
 \hline
 \quad \quad \quad 1 \quad 0 \quad 1
 \end{array}
 \quad
 \begin{array}{r}
 3. \quad - \quad 1 \quad 0 \quad 1 \quad 0 \\
 \quad \quad \quad 1 \quad 1 \quad 1 \\
 \hline
 \quad \quad \quad 1 \quad 1
 \end{array}$$

Правильные ответы прислали:

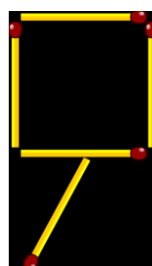
- Абаев Никита, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;
- Асланов Арсен, г. Пенза, школа № 73, учитель **Диков А.В.**;
- Багрова Анастасия, Кушкин Никита, Махнач Виктор и Пышко Иван, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;
- Васина Светлана и Хомутова Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;
- Серов Георгий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;
- Толстоногова Дарья, Совхозная средняя школа, Московская обл., Серебряно-Прудский р-н, поселок Успенский, учитель **Жарикова Е.Н.**;
- Ушаков Александр, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;
- Филиппова Елена, Республика Татарстан, Мамадышский р-н, совхоз “Мамадышский”, политехнический колледж, преподаватель **Порываева Н.С.**;
- Харитонов Никита, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

### Три задачи со спичками

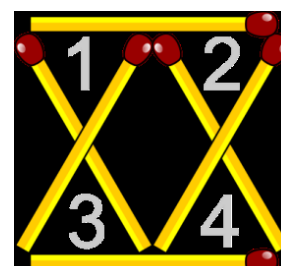
Ответы:



Три правильных треугольника из семи спичек



Часы с маятником



4 равных равносторонних треугольника, лежащие в одной плоскости.

*Ответы прислали:*

- Голубева Елена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;
- Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;
- Куклева Лидия и Серов Георгий, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;
- Толстоногова Дарья, Совхозная средняя школа, Московская обл., Серебряно-Прудский р-н, поселок Успенский, учитель **Жарикова Е.Н.**;
- Филиппова Елена, Республика Татарстан, Мамадышский р-н, совхоз “Мамадышский”, политехнический колледж, преподаватель **Порываева Н.С.**;
- Цаплин Николай, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

### **Задача “Площадь квадрата”**

Напомним условие: “В учебнике, изданном на планете  $\beta$ -Сириус, написано: «Площадь квадрата равна квадрату его стороны. Например, если сторона равна 11, то площадь равна 1001». Почему?”

*Ответ*

Используется двоичная система счисления ( $11_2 = 3$ ,  $1001_2 = 9$ ).

*Правильные ответы представили:*

- Абаев Никита, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;
- Абдувахидова Алина и Абдувахидова Софья, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;
- Асланов Арсен, г. Пенза, школа № 73, учитель **Диков А.В.**;
- Багрова Анастасия, Гашимов Геннадий, Дикий Данил, Кукушкина Карина, Литвищенко Данил, Сагайдак Алексей и Султаншин Ансар, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;
- Васина Светлана и Хомутова Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;
- Кукушкина София, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;
- Ушаков Александр, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;
- Харитонов Никита, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

### **Кроссворд**

*Ответы*

По горизонтали: 4. Программа. 6. Фейсбук. 8. Сноска. 10. Евклид. 12. Каталог. 14. Формат. 16. Отрицание. 19. Синус. 21. Сайт. 22. Лебедев.

По вертикали: 1. Пробел. 2. Три. 3. Импорт. 5. Телефон. 7. Код. 8. Сектор. 9. Кольцо. 11. Истина. 13. Цикл. 15. Мост. 17. Ромб. 18. Ось. 20. Тег. 22. Ля.

*Полностью кроссворд решили:*

- Абдувахидова Алина, Абдувахидова Софья и Куклева Лидия, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;
- Асланов Арсен, г. Пенза, школа № 73, учитель **Диков А.В.**;
- Багрова Анастасия, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;
- Бадаев Александр, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

- Баймурзин Руслан, Республика Татарстан, Мамадышский р-н, совхоз “Мамадышский”, политехнический колледж, преподаватель **Порываева Н.С.**
- Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;
- Полищук Василий, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;
- Харитонов Никита, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

### **Вопросики на сообразительность от Валерия Рубанцева. Порция 3**

1. Как можно съесть слона?

*Ответ:* по частям (ряд читателей предложил ответ, связанный с слоном — шахматной фигурой).

2. На что похожа половина яблока?

*Ответ:* на другую половину яблока.

3. Кто воет вдвое громче, чем собака?

*Ответ:* две собаки (некоторые читатели — знатоки зоологии ☺ — считают, что волк).

*Ответы представили:*

— Абдувахидова Алина, Абдувахидова Софья, Базанов Илья, Серов Георгий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Бадаев Александр, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Дубовиков Максим, г. Пенза, школа № 73, учитель **Диков А.В.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Карпина Нина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

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

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

— Филиппова Елена, Республика Татарстан, Мамадышский р-н, совхоз “Мамадышский”, политехнический колледж, преподаватель **Порываева Н.С.**;

— Фурсова Елена, г. Пенза, школа № 73, учитель **Диков А.В.**

### **Головоломка “Некруглые предметы из кругов”**

*Ответ*

По мнению автора головоломки В.Д. Рубанцева, на картинке можно найти:

- мороженое;
- банан;
- кепку;
- гитару;
- корону;
- конфету;
- свечу;
- телефон;
- песочные часы.

Читатели предложили и другие варианты.

*Ответы прислали:*

— Алиев Тимур, Выскочкин Платон, Желтоухов Владислав, Жучков Иван, Королёва Марианна, Малова Софья, Родионов Александр и Сластухина Анастасия, детский клуб “Крошка Ру”, г. Александров Владимирской обл., преподаватель **Волков А.Ю.**;

— Базанов Илья и Куклева Лидия, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Барановская Татьяна и Жукова Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Голубева Елена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Дубовиков Максим, г. Пенза, школа № 73, учитель **Диков А.В.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Карпина Нина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

— Филиппова Елена, Республика Татарстан, Мамадышский р-н, совхоз “Мамадышский”, политехнический колледж, преподаватель **Порываева Н.С.**

Отметим ответ Тимура Алиева из детского клуба “Крошка Ру” г. Александрова, представившего цветные изображение предметов.

### **Задача “Кто съел варенье?”**

Напомним условие: “Однажды Петя, Вася и Макар остались одни дома. Кто-то из них съел варенье. На вопрос мамы, кто это сделал, они сказали:

Петя: «Я не ел. Макар тоже не ел».

Вася: «Макар действительно не ел. Это сделал Петя».

Макар: «Вася врёт. Это он съел».

Кто съел варенье, если известно, что двое оба раза сказали правду, а третий один раз сказал правду и один раз соврал?”

*Решение*

Допустим, что Петя дважды сказал правду. Тогда получается, что варенье съел Вася. Этот вывод не опровергается высказываниями Макара. Следовательно, варенье действительно съел Вася, и одно из его высказываний правдиво, а второе — ложно.

*Ответ:* Вася.

*Правильный ответ прислали:*

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

— Абдувахидова Алина, Абдувахидова Софья, Базанов Илья, Куклева Лидия, Медведева Анастасия, Серов Георгий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Барановская Татьяна и Жукова Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Гашимов Геннадий, Султаншин Ансар и Чунин Павел, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

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

— Лазаренко Нина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

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

- Филиппова Елена, Республика Татарстан, Мамадышский р-н, совхоз “Мамадышский”, политехнический колледж, преподаватель **Порываева Н.С.**;
- Фурсова Елена, г. Пенза, школа № 73, учитель **Диков А.В.**

Японские головоломки sudoku решили:

- Багрова Анастасия, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;
- Бадаев Александр, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;
- Баймурзин Руслан и Филиппова Елена, Республика Татарстан, Мамадышский р-н, совхоз “Мамадышский”, политехнический колледж, преподаватель **Порываева Н.С.**;
- Дрожжина Виктория и Михайлина Александра, Совхозная средняя школа, Московская обл., Серебряно-Прудский р-н, поселок Успенский, учитель **Жарикова Е.Н.**;
- Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;
- Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;
- Полищук Василий, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

Программу, предложенную для самостоятельной работы в статье “Болтуны рассказывают анекдоты” (рубрика “Школа программирования”), разработали:

- Асатуров Давид, г. Пенза, школа № 68, учитель **Яшина Е.А.**;
- Баранов Николай, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;
- Бубляев Алексей, г. Пенза, школа № 68, учитель **Аверина А.М.**;
- Макаров Сергей, г. Пенза, школа № 73, учитель **Диков А.В.**

Победителями конкурса № 6, задания которого были связаны с системой счисления с отрицательным основанием, стали:

- Баранов Николай, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;
  - Дикий Данил, Коляков Дмитрий, Назаркина Татьяна, Сагайдак Алексей и Чунин Павел, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;
  - Стороженко Степан, г. Пенза, школа № 73, учитель **Диков А.В.**;
  - Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;
- Все они будут награждены дипломами. Поздравляем!

Спасибо всем приславшим ответы в заканчивающемся учебном году! Хорошего отдыха!

### Наше пожелание выпускникам:



## Доска почёта “Мира информатики”

На Доске почёта нашего журнала представлены учащиеся, награжденные дипломами в текущем учебном году<sup>6</sup>.



**Алина Абдувахидова**

Владимирская обл., г. Струнино,  
школа № 11  
(учитель Волков Ю.П.)



**Софья Абдувахидова**

Владимирская обл., г. Струнино,  
школа № 11  
(учитель Волков Ю.П.)



**Алексей Богаткин**

поселок Ардатов  
Нижегородской обл.,  
Ардатовский областной  
многопрофильный техникум  
(учитель Зудин В.П.)



**Галина Друзина**

Московская обл.,  
г. Орехово-Зуево,  
школа № 20  
(учитель Ефимов П.И.)



**Михаил Шестопалов**

Владимирская обл., г. Струнино,  
школа № 11  
(учитель Волков Ю.П.)



**Наталья Юльцова**

г. Саратов,  
Лицей математики и  
информатики  
(учитель Гераськина И.Ю.)

**Спасибо, дорогие ребята! Желаем вам дальнейших успехов!**

<sup>6</sup> См. также предыдущий выпуск журнала.

## Спасибо вам, уважаемые коллеги!

За активную работу с учащимися по материалам интернет-журнала “Мир информатики” редакция журнала “Информатика в школе” выражает благодарность:

- Авериной Анне Михайловне, г. Пенза, школа № 68;
- Артамоновой Виктории Владимировне, средняя школа поселка Новопетровский Московской обл.;
- Багровой Ольге Алексеевне, средняя школа г. Пионерский Калининградской обл.;
- Букиной Елене Юрьевне, г. Зеленокумск Ставропольского края, школа № 1;
- Волкову Андрею Юрьевичу, детский клуб “Крошка Ру”, г. Александров Владимирской обл.;
- Волкову Юрию Павловичу, Владимирская обл., г. Струнино, школа № 11;
- Гераськиной Ирине Юрьевне, г. Саратов, Лицей математики и информатики;
- Демяхиной Оксане Вячеславовне, Республика Коми, г. Усинск, Центр дополнительного образования детей;
- Диденко Виктории Владимировне, Митякинская средняя школа, Ростовская обл., Тарасовский р-н;
- Дикову Андрею Валентиновичу, г. Пенза, школа № 73;
- Долговой Галине Александровне, средняя школа села Восточное Нижегородской обл.;
- Евдокимовой Алевтине Ивановне, средняя школа поселка Осиновка, Алтайский край;
- Ефимову Павлу Игоревичу, Московская обл., г. Орехово-Зуево, школа № 20 имени Н.З. Бирюкова;
- Жариковой Елене Николаевне, МОУ “Совхозная средняя общеобразовательная школа”, Московская обл., Серебряно-Прудский р-н, поселок Успенский;
- Зориной Елене Михайловне, Санкт-Петербург, лицей № 445;
- Зудину Василию Павловичу, Ардатовский областной многопрофильный техникум, поселок Ардатов Нижегородской обл.;
- Каликиной Татьяне Викторовне, Республика Карелия, Сегежский р-н, поселок Надвоицы, школа № 1;
- Комбаровской Светлане Ивановне, г. Воронеж, лицей № 2;
- Муравьевой Ольге Викторовне, средняя школа деревни Муравьево, Вологодская обл.;
- Порываева Нафисе Сабитовне, Республика Татарстан, Мамадышский р-н, совхоз “Мамадышский”, политехнический колледж;
- Черновой Ларисе Ивановне, средняя школа села Сердар, Республика Марий Эл;
- Черноиваненко Ирине Юрьевне, Московская обл., г. Мытищи, ЧУОШ “Логос-М”;
- Шитовой Любове Алексеевне, средняя школа села Горелово Тамбовской обл.;
- Юртаевой Галине Юрьевне, г. Челябинск, школа № 124;
- Ярцевой Ольге Владимировне, г. Ярославль, школа № 33;
- Яшиной Елене Александровне, г. Пенза, школа № 68.

