

Содержание

Введение	6
Важное замечание.....	6
Второе введение.....	7
Часть первая. Я – программист	11
«Кто вы?».....	11
Чей нос лучше?.....	12
Наука и жизнь.....	16
Где мой дом?.....	19
Работать в корзину.....	22
Часть вторая. Юношеское увлечение: Лисп	27
Лисп на «Эльбрусе»: первая предыстория.....	27
Лисп на «Эльбрусе»: вторая предыстория.....	31
Лисп для MS-DOS.....	35
Лисп для .NET: Scheme.....	38
Зачем?.....	41
Часть третья. Первая любовь: Ада	43
Введение.....	43
ASIS: интерфейс будущего для языка из прошлого.....	45
ASIS-for-GNAT: начало.....	48
Двум богам нельзя служить.....	53
.NET и Ада.....	55
AdaCore, GNAT и среды программирования.....	56
И один в поле воин.....	65
Чем плох «чёрный ящик».....	68
Bring Ada to .NET!.....	73
Один в поле не воин.....	79
Заключение.....	80
Часть четвертая. Редкая профессия	82
Введение.....	82
Летает или не летает?.....	84

Что нам стоит дом построить?	85
По рукам!	88
Глаза бояться, а руки делают.....	89
Первые радости.....	91
Что такое идентификатор?	93
Компилятор как таковой: таблицы и деревья.....	95
Лебедь, рак и щука, или Гадкий утенок.....	99
Стиль программирования: на вкус и цвет товарища нет	105
Программирование «наизнанку»	108
Настоящая работа	111
Быстро сказка сказывается, да не скоро дело делается	113
Кризис.....	116
Как отремонтировать подгнивший дом	118
Последнее «прости»	120
Любимое дитя.....	122
Confidential.....	124
Заключение. Полетит?	126
Постскриптум.....	129
Часть пятая. Lessons Learned	130
Введение.....	130
Организация лексического разбора: отдельно или вместе?	131
Синтаксический и семантический анализ: вместе или отдельно? ...	134
Составные имена: лексика или синтаксис?	140
Таблицы и деревья: две структуры или одна?.....	142
Ошибки и их диагностика: несколько соображений на будущее...	146
Заключение	150
Часть шестая. Третий компилятор.....	152
Что дальше?	152
Второй компилятор.....	152
Инструменты и парсеры.....	156
Третий компилятор	159
Вдогонку: что такое СП	161
Часть седьмая. Русские «плюсы».....	166
Проект Wednesday	166
На следующий день.....	167

Дьявол и детали	172
Поехали!	176
Русские «плюсы»	177
Ох уж эта локализация... ..	189
Перекрыть Енисей или перекрыть функцию.....	191
Операторы сложения и инструкции присваивания	195
Начало: планы, персоны и обязанности.....	201
Что говорит компилятор	205
Как организовать работу: инструменты и технологии	207
«Команда».....	211
Кризис.....	215
Исход и «вторая жизнь»	218
Заключение	221
Редкая профессия?	221
Сделать наоборот	225
Предисловие, или Как вывернуть компилятор наизнанку	226

Введение

Я очень люблю рассказывать. Не то чтобы уж так много захватывающе интересного есть у меня, чтобы поделиться с другими. Просто со временем у любого человека образуется некоторый запас, груз, багаж – называйте как хотите – знаний, опыта и просто жизненных ситуаций, которые могут быть любопытны не только ему самому.

К сожалению, в реальной жизни крайне редко удаётся (почти никогда не) рассказать что хочется и именно так, как хочется – спокойно, обстоятельно, не боясь, что тебя прервут, и будучи уверенным, что будут слушать с интересом, дослушают до конца и правильно поймут. Винить в этом некого: во-первых, чтобы удерживать внимание слушателей на протяжении всего рассказа, необходимы врождённые способности, похвастаться которыми могут немногие; во-вторых, просто крайне редко обстоятельства складываются так, что становится возможным спокойный рассказ перед заинтересованной и доброжелательной аудиторией.

Так или иначе, я придумал простой способ реализовать свою склонность к «рассказыванию»: все, что я хотел бы рассказать, я просто возьму и запишу, а там уж как сложится... Интересным это окажется для кого-нибудь – можно и опубликовать; нет – ну и пусть себе лежат эти тексты у меня в ноутбуке. Переживать не буду: ведь я уже и так сделал, что хотел, – рассказал ☺.

Важное замечание

Обязательно надо сказать вот что. То, что здесь написано, – *не мемуары*. По-моему, ничего нет скучнее, чем описывать свою жизнь, – тем более когда искренне считаешь её вполне ординарной. Да, сейчас, может быть, и любопытно читать воспоминания людей, живших в XIX веке, проникать в их психологию, узнавать детали быта, бесследно исчезнувшие к настоящему времени. Возможно, что-то из сегодняшнего тоже скоро исчезнет и останется лишь в подобных воспоминаниях. Но считать, что обстоятельства *собственной* жизни настолько значимы и достойны того, чтобы о них узнали другие, – увольте, у меня никогда не было подобных претензий. Да и, по совести говоря, нет в них, в этих обстоятельствах, ничего хоть сколько-нибудь интересного...

Нет, ей-богу, пусть это делает кто-нибудь другой. А я просто расскажу о нескольких софтверных проектах, в которых участвовал. И еще попытаюсь более-менее внятно описать идеи, которые кажутся мне интересными и перспективными. И которые обязательно будут – в свой черед – реализованы.

Второе введение

Мне 55 лет.

Вроде бы по всем «обыденным» критериям активный период уже позади. Многие скажут: пора переходить от программирования к более «спокойной» работе: рулить командами разработчиков, заниматься «стратегическим планированием», писать технические проекты и книги, учительствовать, прозревать будущее и вообще подводить итоги: писать мемуары, например ☺.

К счастью или к сожалению, у меня всё складывается как-то совсем по-другому. Еще очень давно я заметил, что моя личная эволюция заметно отстаёт от «обычного» хода событий у людей сходного рода деятельности, да и вообще от временных вех, интуитивно понимаемых как «нормальные». Отставание иногда незначительное, иногда разительное, но оно наблюдается почти во всём и почти всё время.

Я потерял девственность... не скажу когда, но значительно позднее большинства своих ровесников. И ещё позднее начал толком понимать, как вообще следует обращаться с противоположным полом.

Я женился в 30 лет, а сына заимел в 33. Не бог вещь какая задержка, по сравнению с другими, но она всё-таки есть – и хорошо ложится в общую картину. А ведь очень многие мои ровесники и ровесницы – уже давно дедушки/бабушки...

Я впервые побывал за границей в 40 лет – иные к этому времени успели там пожить, второе гражданство получить, а некоторые – даже вернуться домой. (Правда, первой моей за границей оказалась Швейцария. Наверное, именно поэтому я навсегда полюбил эту страну и по сей день считаю ее лучшей в мире.)

Я стал более-менее прилично зарабатывать годам к 44-м, несмотря на широкие к тому возможности для людей моей специальности. (Знаю программистов, которые даже при советской власти умудрялись получать весьма неплохие доходы, устанавливая свои системы по всему Советскому Союзу.)

Я научился водить только в 45 лет и тогда же купил свою первую машину. Некоторые мои одноклассники и однокурсники умели во-

дить и реально водили – не важно, что машины были родительскими (или угнанными ☺) – лет с восемнадцати...

С последним обстоятельством связано одно очень яркое и важное для меня воспоминание. Как-то летом мы с семьёй отдыхали на море в Словении и однажды поехали на экскурсию в Люблян. Дорога проходила по очень живописным местам, среди высоких холмов (почти гор), поросших густым красивым лесом. На какой-то стоянке автобус остановился, чтобы дать пассажирам отдохнуть от долгой поездки, как это обычно бывает. Прогуливаясь по парковке, я увидел два или три «кемпера» – мини-автобуса, предназначенных для семейных путешествий. Очень удобная вещь: и средство передвижения, и достаточно комфортный дом (почти в любом кемпинге его можно подключить ко всем коммуникациям). Такие караваны, конечно, не покупают, а берут напрокат на три-четыре недели и колесят в них по всей Европе. Получается значительная экономия на гостиницах при сохранении свободы от predetermined туристических маршрутов, характерной для автопутешественников.

Люди вокруг «караванов» занимались своими обычными делами – отдыхали, копались в вещах, готовились продолжать свой путь. Я наблюдал, застыв на месте. Мгновенно, потрясающе отчетливо и безысходно мне стало ясно, что такого у меня не случится уже *никогда*. Было абсолютно невозможно представить, что за оставшийся у меня период активной жизни я успею научиться водить (и получить нужный опыт управления машиной!) и попасть в обстоятельства, позволяющие таким образом проводить свое свободное время. Слишком много радикальных изменений должно произойти в жизни, чтобы такое стало возможным...

И что же? К сегодняшнему дню я объехал пол-Европы, от Италии до Норвегии и от Испании до Чехии, накрутил несколько тысяч миль по Америке и совершенно спокойно чувствую себя, въезжая в до сих пор незнакомый европейский город и отыскивая в нем нужную улицу или просто «исторический центр»...

Я поступил в аспирантуру не сразу после окончания МГУ, а лет через десять, уже работая в оборонном НИИ. Диссера у меня тогда не получилось – вместо обычного текста с жёстко определённой и хорошо известной структурой я навалял какие-то фантазии «на свободную тему», которые даже для статьи не годились... В результате, когда я перешел на работу в МГУ, мне заново (из-за истечения допустимого «срока давности») пришлось сдавать кандидатский экзаме́н по специальности. Кандидатскую диссертацию я написал (за

три летних месяца, по вечерам) и защитил, только когда подступило «по самое некуда»: пригласили работать в политехнический институт в Цюрихе, и было просто неприлично ехать туда без учёной степени. Это произошло тоже в 45 лет.

Из-за такой задержки с диссером со мной однажды случился ужасный конфуз. В 1995 году, когда я участвовал в проекте ASIS и сотрудничал (в основном удаленно) с политехническим институтом в Лозанне (EPFL), мне сообщили, что в Мадридском университете состоится некий workshop, в котором приветствовались бы выступления на темы, связанные с языком Ада. И говорилось, что возможно участие за счёт приглашающей стороны. Набравшись наглости, я послал туда заявку, и, к моему немалому удивлению, она была принята. Так вот, на этом workshop'e все дружно называли меня «доктор Зуев», хотя, видит бог, никакого повода для этого я не давал. Когда ко мне обратились таким образом в первый раз, я то ли не заметил, то ли не обратил внимания, а потом поправлять говорившего просто не хватало духа. Наверное, испанцы и представить не могли, что такой солидный дядя, выступающий вместе со Столлманом (они его тоже пригласили), – никакой не доктор, а заурядный программист. Впрочем, мой внешний вид многих сходным образом обманывает... ☺

Кстати, именно на этом workshop'e я впервые в жизни делал сообщение на английском. Это было, напомним, в 41 год, и мой «английский» был еще весьма и весьма дурным. Да что там! – и на отечественных конференциях я начал выступать очень поздно (и соответствующий опыт написания статей приобрел далеко не сразу), долгое время держась в тени своего научного руководителя...

Я впервые встал на горные лыжи, когда мне стукнуло 45 лет. Это, конечно, не самое важное в жизни умение (тем более что катаюсь-то я до сих пор как типичный «чайник»), но подобная «задержка в развитии» хорошо показывает общую тенденцию, чрезвычайно для меня характерную.

Ну и так далее... Повторяющиеся ситуации позднего наступления важных событий и изменений в жизни (а их на самом деле было значительно больше, чем я описал) утвердили меня в простой мысли. Подобные «задержки» носят вполне объективный характер и, видимо, присущи мне изначально (может быть, таков Замысел). Поэтому не стоит (да и бессмысленно) расстраиваться, комплексовать по этому поводу и дёргаться, пытаясь подстегнуть естественный ход событий. Надо просто все время *действовать* – спокойно, но неотступно,

активно, но без суеты, в различных направлениях, но целенаправленно – *и всё будет, всё чаемое наступит, всё случится!*

...Только вот все чаще приходит опасение и даже страх: это-то всё понятно и правильно, но это работает, только пока впереди остается хоть какой-то резерв времени. А его-то всё меньше и меньше – ведь странно ожидать, что мне отпущено больше, чем другим! Например, у меня до сих пор нет собственной квартиры или дома – и совершенно неизвестно, хватит ли времени занять какую-нибудь недвижимость. Я до сих пор не миллионер – с ещё менее реальными перспективами. И даже не доктор наук (в российском понимании)!..

Утешает одно: работаю-то я в Швейцарии, а здесь пенсионный возраст для мужчин – 65 лет ☺. Даже если я буду продолжать работать по найму, у меня ещё десять лет работы, это же прорва времени! Если отпустить на каждый жизненный или профессиональный «проект» по полгода (минимальный период, необходимый для любого мало-мальски существенного дела – это тоже вывод из моего опыта), то ещё очень-очень много можно успеть сделать. И, с другой стороны, ведь ужасно жалко было бы провести это время в вялом ничегонеделании (или просто в рутинной работе), не сделав ничего заметного...

Так что если по-прежнему хочется программировать – надо программировать, копать в битах и байтах, осваивать новые языки, паттерны и технологии, не смущаясь тем, что выглядишь «белой вороной» среди молодых ребят. А «стратегии», книги, лекции и прочие более «солидные» виды деятельности (к которым тоже ведь есть склонность!) успеются, тем более что до сих пор получалось совмещать их с любимым делом.

Так что – вперед!

Часть первая

Я – программист

«Кто вы?»

Я – программист.

После такого начала необходимо перевести дух и кое-что пояснить. Во-первых, эта фраза (да еще и вынесенная в заголовок) может сразу вызвать ассоциации с названием знаменитой винеровской книги «Я – математик»¹. Так вот, у меня совершенно нет амбиций такого уровня, чтобы таким нехитрым способом встать рядом с основоположником кибернетики («вас здесь не стояло» – мог бы сказать он). Просто я действительно программист, больше ничего делать не умею, и заголовок честно отражает этот факт. Я хочу поделиться своим опытом, полученным в результате почти непрерывного четвертьвекового программирования – то есть чтения, придумывания, написания и отлаживания программ, а также раздумий над ними и любования ими.

Во-вторых, могу поклясться, что это название я придумал сам. Более того, долгое время (около года) файл с этим текстом состоял из этого единственного утверждения. Однажды набрав данную фразу, я немедленно погрузился в мучительные раздумья и, наконец, махнув рукой, решил отложить попытку написать что-то, отличное от очередного технического проекта. Как известно, Word автоматически присваивает новому файлу имя, взятое из первой фразы текста. Так что файл с гордым местоимением в качестве имени проболтался у меня в ноутбуке довольно долго, пока, наконец, я не вернулся к нему опять. Вот тут-то я вспомнил, что и здесь не первый. Кен Томпсон, один из авторов системы UNIX, своё выступление при вручении ему премии имени Тьюринга начал точно так же: «Я – программист. Именно так я пишу в анкетах в графе “род занятий”. Как программист я пишу программы».

Поистине шагу нельзя ступить, чтобы не толкнуть классика! Всё, решительно всё уже сказано...

¹ «Отвратительная, саморекламная книга Винера» – слова академика Дороницына (сам слышал).

Когда-то давно, еще в прошлые времена меня, новоиспеченного начальника сектора закрытого НИИ, направили на так называемые «курсы повышения квалификации руководящих работников». Там в числе прочих нам читали курс под названием «Психология управления», или что-то в этом роде. Лектор в качестве примера самоидентификации рассказывал о результатах некоего уличного анкетирования, в котором задавался единственный вопрос: «Кто вы?» По умолчанию предполагалось, что, отвечая на него, респондент непременно скажет о самом для себя главном – о том, что составляет существо его личности¹. Не помню, каковы были результаты этого опроса; приводились примеры ответов: «Я – грузин», «Я – жена Иванова». Лектор на полном серьёзе сетовал, что никто не сказал «Я – комсомолец»...

Именно тогда мне стало абсолютно ясно, как бы я ответил на такой вопрос.

Чей нос лучше?

Наверное, споры о том, какой язык программирования лучше, начались с того момента, когда был придуман второй язык программирования.

Возразят: кому какое дело до того, на чём написана моя программа, – главное, чтобы она делала то, что от неё требуется. Знал одного программера, который довольно долго писал заказные программы для Windows на языке Ада, и у заказчиков даже и мысли не возникло поинтересоваться языком реализации: программы работали надёжно и без сбоев. Однако это справедливо только для крайне ограниченного множества программных изделий. Можно согласиться, что какая-нибудь несложная утилита с интерфейсом командной строки может быть написана хоть на Перле, хоть на языке ассемблера. То же с известной натяжкой относится и к задачам чисто вычислительного характера. Но этим, по существу, круг «индифферентных» к языку задач исчерпывается. Например, решение о необходимости сколько-нибудь развитого интерфейса сразу сужает выбор инструмента до одного-двух языков. Учёт же дополнительных обстоятельств – тре-

¹ Совершенно не понятно, правда, на чем основана уверенность, что человек в несколько секунд адекватно это сформулирует, – по-моему, большинство должно ответить: «Я такой-то», просто назвав свою фамилию. Удивительно самоуверенная наука – психология.

бования производительности, взаимодействие с другими программами, переносимость, ожидаемый характер сопровождения, будущие модификации и расширения – как правило, не оставляют какой-либо альтернативы.

Я сам видел текстовый форматировщик (документатор), написанный на Фортране, и пользовался им. Конечно, можно написать компилятор на Бейсике (если очень постараться) – но насколько быстро он будет работать? Задачи сортировки удобно программировать на Borland Pascal – но как такую программу перенести под UNIX? Вполне возможно реализовать СУБД на Си++ – но как будет сопровождать и развивать эту СУБД персонал, который даже Си не знает? С другой стороны, на чём ещё, кроме как на Си, можно писать системные программы под UNIX, где всё «заточено» под этот язык? И на чём разумно программировать библиотеку утилит для анализа Ада-программ, как не на самой Аде?

Так что я могу определённо утверждать, что в девяноста случаях из ста программист реально *не волен* выбирать. Любая сколько-нибудь нетривиальная программа, предназначенная не для самоутверждения и самоудовлетворения автора (или, по крайней мере, не только для этого), а для реального использования другими людьми или другими программными системами, должна характеризоваться определенным набором свойств: конкретной функциональностью, связанной с предметной областью, адекватным задаче способом организации интерфейса, заданной средой исполнения, сроками создания, наконец. Учет этих обстоятельств почти автоматически исключает какой бы то ни было произвол и приводит к совершенно конкретному выбору – операционной системы, прикладных библиотек и, разумеется, используемого языка программирования. К тому же зачастую язык реализации явно определяется заказчиком, и обсуждать это вообще бессмысленно (тем более что вы как программист можете с заказчиком ни разу и не свидеться – между вами не один посредник).

А если программист начинает работать в некоторой организации (а в организации или, в крайнем случае, в серьёзном коллективе только и возможно сделать сколько-нибудь значимую программную систему – исключения настолько редки, что своей малочисленностью только подтверждают общую тенденцию), то он неизбежно вынужден следовать некоторым традициям, свойственным этой организации, принятым правилам и соглашениям, которые почти всегда подразумевают использование определённого ЯП. Более того, чем «круче» и респектабельнее фирма, тем с большей вероятностью она имеет то,

что называется отраслевыми стандартами, «соглашениями о стиле программирования» и т. п. Так что если даже эта контора и использует ваш любимый язык, то не очевидно, что вам понравится то, *как* она его использует.

Мне кажется, что сама готовность вступить в спор о том, какой язык лучше «вообще», без анализа требований, обсуждения сферы применения (предметной области), программной и аппаратной платформ, переносимости результирующей программы и т. п., свидетельствует об определенной незрелости программиста как профессионала. Настоящий специалист, поняв, что данная задача требует использования нелюбимого языка, лишь поморщится (может быть, пробормотав про себя: «Ненавижу» ☺), раскроет руководство и через день-два напишет и откомпилирует первый фрагмент будущей программы. Легендарный канадский хоккейный вратарь Кен Драйден сказал в одном из интервью: «Я ненавижу стоять в воротах, но это моя работа».

Иногда приходится слышать такое обоснование предпочтения одного языка другому: язык X скоро умрёт, так как появился язык Y, который «гораздо лучше». Некоторое время назад я предложил одному неглупому студенту мехмата участвовать в работе по написанию комментариев к Стандарту Си++. Поначалу с энтузиазмом согласившись, он вскоре заявил, что эта работа потеряла всякий смысл: Си++ скоро станет никому не нужен из-за появления... нового языка C#!

Несмотря на анекдотичность высказывания, оно далеко не уникально – подобное приходится слышать довольно часто. На это можно возразить, что в короткой, но насыщенной истории информатики не было еще *ни одного случая*, чтобы появление нового языка привело к исчезновению какого-либо «старого». Язык PL/I вобрал в себя многие свойства Фортрана и Кобола, и что же? – эти языки выжили и чудесно себя чувствуют по сию пору! Сам PL/I умер вовсе не потому, что возникло что-то лучше него, – просто он *как таковой* оказался слишком громоздким, сложным и бессистемно построенным. Ни Си, ни Паскаль вовсе не «уничтожают» друг друга – оба вполне живы как сами по себе, так и в своих диалектах и ближайших потомках. При этом появление Си++ вовсе не отодвинуло Си на задворки программирования – язык живет, используется, развивается, для него регулярно выходят новые стандарты ISO. Ада, практически целиком включив основные концепции многих ЯП, не «отменила» ни Паскаль, ни любой другой свой прообраз. Примеры можно множить. Так что оставим броские заголовки типа «Java отменяет Си++» или «C# – убийца Java» на совести журналистов.