**Является ли математика частью информатики?**

Цейтин Г.

Я не согласен с утверждением, что информатика (какой бы термин для нее ни использовали) - это набор практических навыков и решений, в лучшем случае - искусство, и никакого фундаментального научного содержания она иметь не может. Я думаю, что, несмотря на возможные терминологические недоразумения, мы все более или менее одинаково понимаем, о какой области деятельности идет речь. Информационные технологии в современном мире - это уже давно утвердившаяся реальность, так же, как и то, что существуют профессионалы, специализирующиеся именно в этой области, что необходимо готовить специалистов в этой области, писать книги и статьи, издавать журналы, оценивать профессиональный уровень и т.п. Раз есть область, должно быть имя, чтобы ни с чем не спутать. Термин информатика (французское informatique) представляется мне достаточно удачным, и, во всяком случае, лучшим, чем американское computer science.

Прежде, чем пытаться уточнить содержание этой области (насколько это вообще возможно), хотелось бы проследить, как вообще формировалось самоосознание этой области, как особой области знаний, а также общественное признание ее самостоятельности (сегодня такое признание - уже свершившийся факт).

Но ни самоосознание области, пионеры которой были и считали себя специалистами в других областях, ни выделение ей места под солнцем в ряду других специальностей, не могли быть простыми. Новая область требует выделения ей отдельных ресурсов, и людских, и материальных, а это не могло происходить бесконфликтно и вполне добросовестно. Я имею в виду не только СССР, где все эти проблемы многократно усиливались научным монополизмом и порожденными им интригами. Эти проблемы существовали и в более благополучных странах.

Нового предмета, отличного от всего, что было прежде, информатика не создала. Программа вполне подходит под математическое понятие алгоритма (с некоторыми уточнениями из-за параллельного исполнения или недетерминированности), так где же новый предмет? И представители традиционных областей, стремясь удержать под своим контролем ресурсы, выделение которых на новые приложения диктовалось практическими потребностями, пользовались этим аргументом. Как говорил один мой коллега-матфизик: "И чего это все так носятся с этим системным программированием? Это ведь всего-навсего программирование для системы машин!" А другой коллега, просматривая проект учебных программ по информатике, заявлял: "Это какая-то эклектика, просто собраны вместе кусочки, принадлежащие другим дисциплинам". Впрочем, подобная аргументация известна еще из пушкинской "Сказки о царе Салтане", где "ткачиха с поварихой, с сватьей бабой Бабарихой" развенчивали (и небеcкорыстно) одно за другим все чудеса, о которых рассказывали заморские гости.

Надо признать, что и представители новой области допуcкали натяжки ради того, чтобы организационно выделить свой предмет. Мне приходилось видеть математические работы, где поверхностно формализовывались некоторые, уже устарелые, программистские концепции, а затем доказывались "сногсшибательные" результаты, основанные на совершенно нереальных примерах (в математике это нормально, но на основе этого следовало просто заменить первоначальные понятия, чтобы они не включали подобные случаи). И все это делалось ради того, чтобы заявить о принадлежности своих (не бог весть каких) результатов новой перспективной области. Впрочем, эти болезни были постепенно преодолены.

А как осознает себя эта область теперь, когда организационное признание состоялось? К сожалению, осознания особого предмета по-настоящему нет. Авторитетные специалисты, пришедшие из других областей, зачастую связаны прежними представлениями, а молодежь, похоже, просто работает, не задумываясь об отграничении предмета.

А необходимость в таком осознании есть. Действительно, компьютерная программа может рассматриваться как разновидность алгоритма, но почему в таком случае возникают все новые и новые языки программирования и новые концепции, например, объектно-ориентированное программирование? Ведь в принципе любая программа эквивалентна некоторой машине Тьюринга, так что вроде бы ничего нового все эти языки не несут! А дело в том, что даже в одном и том же предмете с разных точек зрения могут быть важны разные стороны, и то, что важно с точки зрения математики, не совпадает с тем, что важно с точки зрения информатики. Различие между математикой и информатикой в оценочных критериях в свое время достаточно четко описал Ласло Кальмар\*, пришедший в программирование из математической логики.

Главное отличие от математики, хотя бы и рассматривались одни и те же объекты, состоит, на мой взгляд, в том, что в информатике определяющим является человеческий фактор. Программы пишутся людьми, часто большими коллективами; даже если программу пишет один человек, он пользуется полученными от других людей знаниями и приемами, и, возможно, получил первоначальное задание от кого-то другого. Программа имеет жизненный цикл: после создания она может модифицироваться, переноситься в другую среду, стыковаться с другими программами, и в конце концов выходить из употребления (тоже по разным причинам). Учитывает ли эти реальности математическое понятие, претендующее на определение программы? Разве может алгоритм (в точном математическом понимании) меняться? Если что-то изменится, это будет просто другой алгоритм, который не надо путать с первым. Если же мы хотим, чтобы программа, например, адаптируемая к другому окружению, оставалась тем же самым объектом, то понятие программы должно существенно измениться (в общем, некоторые математические подходы к описанию этого тоже возможны, но никогда теоретики со стороны математики этого не предлагали). Важной особенностью программ является то, что они могут иметь ошибки, и, во всяком случае, необходимы меры для уменьшения их числа и ущерба от них. Это тоже не из математики. Вспомним, кстати, и о проблеме двухтысячного года.

Для чего придумываются новые и новые языки программирования? "Для большего удобства", возможно, ответят. А что такое удобство? Это снова человеческий фактор. Если всерьез, то это более точное соответствие элементов предлагаемого языка и тех понятий и знаний, которыми пользуется человек, ставя и решая ту же задачу. Значит, надо понимать структуру человеческого знания и человеческого мышления. Целый ряд особенностей новых языков программирования, возможно, казавшихся первоначально случайными удачными находками, отражает глубокие черты организации человеческих знаний и человеческого языка. Но достаточного внимания этому не уделяется. (Один мой коллега из промышленности, нуждавшийся в специалистах по языкам программирования, спрашивал меня, не на филологическом ли факультете их искать.)

Кроме индивидуального фактора, есть еще и социальный. Уже упоминалось о разработке программ большими коллективами, а также о сопровождении программ. Есть еще необходимость переносить программы на вновь разрабатываемую технику или включать в новые системы. Надо считаться и с тем, что программы, работающие на одной машине или иным образом взаимодействующие либо совместно пользующиеся ресурсами, могут отражать интересы разных людей, возможно, находящихся в конфликте, и, значит, надо заниматься защитой программ.

Когда стал широко входить в жизнь Интернет, стала насущной необходимостью разработка подходов, обеспечивающих правильное взаимодействие большого числа независимо разработанных программ (или их элементов). Одновременно увеличение скорости машин сделало менее существенной экономию команд на нижнем уровне. В сумме это привело к использованию структур данных, которые раньше считались бы неприемлемыми из-за неэффективности, и, соответственно, к новой организации языков. Широкое распространение модульности вместе с приемлемостью больших накладных расходов на межмодульное взаимодействие привело к уменьшению размеров отдельных модулей, и, как следствие, к упрощению синтаксиса языков. Кстати, в этом я вижу и причину создания и последующего широкого распространения языка Java.

Кому же заниматься исследованием человеческих и социальных факторов в информатике? Снова чудится мне указующий перст чиновного скептика, внушающего, что это дело таких-то и таких-то уже существующих наук: психологии, социологии, и т.п. Не получится! И не только потому, что во всех этих науках есть свои интересы и предпочтения, а их представители могут и не понять важности проблем, диктуемых задачами информатики (у меня есть определенный отрицательный опыт). Достаточно рассмотреть, к чему привело отождествление информатики с математикой.

Информатика получила от математики ряд результатов и теорий, нашедших широкое применение, в особенности в теории языков и трансляции, а также (в меньшей мере) по верификации программ. Вместе с тем, поскольку это делали математики (или люди, относившие себя к математике) выбор задач диктовался желанием получить результаты, интересные в математическом смысле, а другие, не менее важные для информатики, задачи оставались без внимания. Теория языков и трансляции, например, оказалась слишком раздутой, а вопросы модульности (что на сегодня важнее) не получили должного развития. Преувеличена была и роль логической верификации - на практике требования к программам редко оформляются в логических понятиях. Отрицательную роль сыграла и ориентация на "диссертабельность". Программистские работы, содержавшие достаточный творческий вклад, обладавшие и новизной, и полезностью, и делавшие ее автора достойным ученой степени, искусственно подводились, ввиду отсутствия надлежащих рубрик, под вычислительную математику, экспериментальную физику и т.п., и люди, причастные к прохождению работы, закрывали глаза на то, что она заявленной специальности не соответствует. Это, в свою очередь, приводило к появлению, в силу прецедента, диссертаций, не содержавших серьезного вклада ни в "титульную" область, ни в информатику. С другой стороны, программистские работы иногда снабжались "орнаментальной" математикой, когда определения искусственно стилизовывались под математические теории.

Нет оснований рассчитывать и для других смежных областей, что вопросы, практически важные для информатики, будут успешно решаться в рамках этих областей. Отсутствие полномасштабного самоосознания информатики как особой науки начинает мешать ее развитию.