

О ПРОГРАММИРОВАНИИ

В 1953 году группа выпускников математико-механического факультета Ленинградского университета была направлена на работу в Ленинградское отделение Математического института им. В. А. Стеклова АН СССР (ЛОМИ), в отдел приближенных вычислений. Отделом руководил Леонид Витальевич Канторович. Он предложил нам заняться освоением только появившихся тогда ЭВМ. Леонид Витальевич увлек нас своими идеями. Понятия "схемы, уровни, крупноблочность" сразу вошли в наш обиход и определили направление работы. Вместе с нами знакомился Леонид Витальевич на семинарах с системами команд первых машин. Вместе слушали лекции по математической логике, которые по просьбе Леонида Витальевича прочитал в ЛОМИ Николай Александрович Шанин. Эти занятия велись наряду с плановой вычислительной работой, и естественно, что часть программ реализовала эти вычисления на ЭВМ. Но с самого начала вопросы программирования Леонид Витальевич рассматривал и шире, и глубже.

Речь сразу пошла о создании специальной математической символики и разработке систем автоматизации программирования. Это был зародыш будущих алгоритмических языков программирования, причем языков функционального типа, которые только теперь начинают обретать права гражданства, поддержанные сравнительно недавними успехами математической теории вычислений и прогрессом вычислительной техники. Много раз Леонид Витальевич развивал

свою главную мысль о том, что эта символика должна быть удобной как для человека, так и для машины. Иными словами, с одной стороны, она должна отражать систему представлений человека, понятия и методы математической теории вычислений, с другой стороны, она должна отражать особенности выполнения вычислений на машине, допускать эффективную машинную реализацию. В свою очередь, обоснованные и продуманные конструкции языка неизбежно должны оказывать влияние на проектирование новых машин.

Хотя машин в Ленинграде в то время не было и со своими программами нам приходилось ездить в Москву, в ЛОМИ сложилась в целом благоприятная для программирования творческая обстановка. Леонид Витальевич сам великолепно знал и чувствовал вычислительные потребности математики. Кроме того, в семинарах и обсуждениях принимали участие другие известные специалисты по различным математическим дисциплинам: В.Н. и Д.К. Фаддеевы, В.Н. Кублановская, С.Г. Михлин, А.А. Марков, Н.А. Шанин и др. Такая аудитория обеспечивала надежный выбор модельных задач, вычислительные механизмы которых исследовались и формализовались и на которых опробовались языки и системы. В сферу общих интересов попадали не только численные вычисления, но и другие виды интеллектуальной деятельности человека, где в той или иной мере возможно было использовать ЭВМ. Интерпретаторы и трансляторы, аналитическое дифференцирование, поиск первообразной, доказательство теорем, получение символьных решений для дифференциальных уравнений и другие символьные вычисления были представлены в работах того времени. Особенно привлекали Леонида Витальевича задачи, находящиеся на грани разрешающей способности машины или даже за ее пределами. Найти в механизме вычислений основное сдерживающее звено и придумать способы преодоления препятствия — мы видели, как это получалось у Леонида Витальевича. Решения бывали разными: либо специальные численные методы, либо новые методы программирования и новые конструкции в языках, либо идеи о новых вычислительных устройствах. И при этом неизменно высокая универсальность сочеталась с надежной практической эффективностью.

В 1954 году вышла книга А.А. Маркова "Теория алгоритмов". Обсуждение проблем алгоритмизации вычислений и вообще всего круга вопросов, связанных с конструктивным направлением в математике, наложило свой отпечаток на развитие программирования в

Ленинграде. Первоначальные попытки построения практических функциональных систем определялись естественным стремлением использовать в программировании привычный математический аппарат и, в частности, понятия и символику близких областей. Так, операции над алгоритмами, систематически использовавшиеся, например, в упомянутой книге Маркова, представлялись удобным и естественным инструментом для описания вычислительных процессов и в программировании. Это привело к разработке своего рода алгебры программ. Практический опыт программирования показал важность проблемы представления и описания данных и вызвал появление алгебры данных. Эти алгебры разрабатывались над укрупненными, агрегированными объектами, внося иерархическую структуру в языки и освобождая верхние уровни от ненужной детализации. Все это породило свою методологию исследований и в теории и практике трансляции, и в практическом программировании, в известном смысле близкую той идеологии, которая развивается ныне в теоретическом программировании.

На основе теоретических результатов сейчас возникает обсуждение многих проблем, которые рассматривались в то время с позиций практики и математической интуиции. Вспоминается такой эпизод.

При анализе многих вычислительных механизмов выяснилась некоторая общая структура, которая была введена как операция над процедурами и названа рекурренцией. Теперь можно было бы сказать, что речь шла о неподвижной точке рекурсивного уравнения

$$F \Leftarrow p \rightarrow f; \sigma \circ [\varphi, F \circ \varphi]$$

и, стало быть, о введении в язык ограниченной рекурсии. Этим уравнением описывается, в частности, интегрирование по частям и другие математические вычисления. Для операции рекурренции было разработано и реализовано достаточно эффективное в вычислительном отношении итеративное представление. Но только в последнее время, после работ Д. Скотта и др. появилась возможность строго доказывать эквивалентность итеративного представления рекурренции и анализировать подобные факты. "Леонид Витальевич, вам очень нужна эта рекурренция?" - спросил, раскатывая "р", Андрей Андреевич после долгих обсуждений вводимой операции. Сам он полагал, что в языке можно ограничиться итерацией. "Это полезно во многих вопросах", - ответил Леонид Витальевич. "Ну ,

хорошо", - согласился, наконец, Андрей Андреевич с молодым докладчиком. Снова и снова удивляешься силе и глубине творческой мысли Леонида Витальевича, позволившей ему 25 лет назад верно представить принципиальные пути развития зарождающейся науки и немало сделать в этом направлении.

Когда в 1960 году на конференции в МГУ был взят курс на построение трансляторов с Алгола и концентрацию усилий вокруг этих работ, было до слез жалко, что программирование отходит от математики в дремучую аморфность алголоподобных языков. В ответ на наши жалобы Леонид Витальевич тогда сказал: "Машины должны работать". По-видимому, это действительно был самый реальный путь - объединенными усилиями специалистов многих стран в кратчайшие сроки разработать математическое обеспечение для машин и заставить их работать в полную силу. Ориентированные на машины алгоритмические языки получили тогда широкое распространение и были развиты в большей степени по сравнению с другими подходами.

Однако в настоящее время в программировании намечаются другие тенденции. Особенностью схемных крупноблочных систем программирования являлось то, что универсальные средства усиления выразительности включались в языки под строгим контролем эффективности. Представляется, что многие решения актуальны и сегодня. Так, уже первые системы реализовали то, что сейчас в функциональном программировании называют вычислениями с задержкой. В системе для аналитических выкладок была разработана специальная алгебра программ, реализующих такие вычисления. Схемы Канторовича, методы включения в языки ограниченной рекурсии, методы трансляции, многие другие принципы, опробованные в крупноблочных системах, несомненно найдут применение и дальнейшее развитие в новых системах программирования функционального типа.

Л.Т.Петрова