

Комбинаторная логика, или настойчивое приглашение к компьютерингу

Предисловие к 3-му изданию книги «Комбинаторная логика в программировании».—3-е изд., дополн. и перераб.— М.:АО «Центр ЮрИнфоР», 2008.— X+384 с.

Вольфенгаген В.Э.

Кафедра ПКИ и ИТ, Институт «ЮрИнфоР-МГУ»,
Москва, 119437 РФ,
vew@jmsuice.msk.ru,

WWW: <http://www.wolfengagen.mephi.ru>

Аннотация Компьютеринг и его развитие ставит целый ряд вопросов, на большую часть из которых ответы либо неполны, либо неизвестны. Некоторые из них: что такое ‘вычисление’? что такое ‘информация’? что можно узнать, пользуясь компьютерингом? чего нельзя узнать, пользуясь компьютерингом? – имеют фундаментальное значение.



В настоящей работе основное внимание уделено выяснению технологических особенностей вычислений с объектами. Их взаимодействие рассматривается в аппликативной среде, что позволяет выяснять внутреннюю структуру обычных операций, знание которой позволяет понять их свойства. Обсуждается выбор исходных константных сущностей, считающихся исходными и называемыми *комбинаторами*. Эти исходные сущности используются как основные “строительные блоки”, вступающие в аппликативной среде во взаимодействие друг с другом. В результате взаимодействия возникают конструкции, дающие представительные наборы обычных операторов и погруженным вычислительным системам.

Об авторе. Вольфенгаген В. Э., д.т.н., профессор (vew@jmsuice.msk.ru). Заведующий кафедрой перспективных компьютерных исследований и информационных технологий (ПКИ и ИТ) в Институте «ЮрИнфоР-МГУ». Область его научных интересов составляют компьютерные науки и информационные технологии, включая аппликативные вычислительные системы, λ -исчисление, комбинаторную логику, системы типов. Проекты РФФИ 93-01-00943-а (ЛАМБДА), 96-01-01923-а (КООАМ), 05-01-00736-а.

Анализ сущностей. При анализе сложных сущностей или явлений обычно стремятся найти способ их реконструкции из более простых сущностей или явлений, между которыми можно установить или выделить отношение – их устойчивую и воспроизводимую связь. Точно также действуют и при анализе результата рассуждений, полученного логико-математическими средствами. Как известно, любая попытка проведения конструирования такого результата опирается на постулаты – аксиомы и правила вывода, – каждый из которых представляется достаточно элементарным и интуитивно оправданным. Другими словами, в качестве исходных выбираются наипростейшие предложения, из которых, пользуясь логическими средствами, конструируется либо подтверждение, либо опровержение гипотетического результата. Для многих логика в целом и математическая логика в частности представляются довольно сложными для интуитивного восприятия и непосредственного применения, причем в особенности это относится к теории доказательств.

Моделирование сущностей. Иной путь подсказывается опытом - нельзя ли этот гипотетический результат смоделировать не рассуждением, а вычислением? Подобно тому, как исследуя свойства вещества, можно многое о нем узнать, пользуясь системами рассуждений, но можно попросту попытаться его синтезировать из более мелких сущностей. В частности, молекулу строят из атомов, а атомы рассматривают с позиций их конструирования из более мелких частиц и так далее, пока не удастся найти предел такого дробления, если только он возможен. Как признается в настоящее время, в математике роль своего рода мельчайших частиц играют константы и переменные, их связь посредством отношений ведет к конструкциями, для которых

можно говорить об их истинности или ложности, далее из них, посредством нормированных аксиомы и правилами рассуждений, выводят подтверждение или опровержение гипотетического результата. Хотя константы и переменные и относят к фундаментальным для математического знания понятиям, но обычно делают это номинально. Это отягощается появлением в логике операторов, различным образом связывающих переменные. Идея связывания чрезмерно усложняет логико-математический язык, делая его доступным тем посвященным, которые не пожалели сил на его изучение и детальное практическое овладение. Само связывание переменных все еще нельзя считать достаточно изученным и тем более доведенным до элементарного понимания.

“Тонкая” структура компьютеринга. Конечно, хорошим вопросом является вопрос, откуда взялись константы и переменные, но он достаточно хорош, чтобы пытаться давать на него исчерпывающий ответ. Обычно наличие и существование констант и переменных принимается на веру и особого протеста не вызывает. Но, пожалуй, единственным способом построения из них математического или математизированного знания без привнесения дополнительных усложняющих предположений дает комбинаторная логика. Эта формальная система открыта сравнительно недавно, она пережила несколько периодов своего развития, включая становление, распространение и известную популярность, сопровождавшиеся также и известной потерей интереса со стороны математиков, но ставшая высоко цениться в компьютерных науках. В настоящее время интерес к комбинаторной логике начинает вновь нарастать, в известной мере стимулируясь развитием функционального, или, по иной терминологии, аппликативного подхода к программированию. Многих, работающих в компьютерных науках, привлекает простота, конструктивность и концептуальная ясность функционального программирования и лежащей в его основе комбинаторной логики, хотя для последней это только одна из многих областей применения.

Комбинаторная логика и аппликативный компьютеринг. *Аппликативный компьютеринг* предполагает комбинационное построение вычисления как относительно самостоятельного блока, пользуясь уже имеющимися блоками вычислений, причем все переменные в каждом блоке вычисления связаны, а сам он – замкнут. Для осуществления аппликативного компьютеринга применяют аппликативные вычислительные системы (АВС).

Все, что есть в *комбинаторной логике* – это константы, переменные и способы их комбинирования. Агрегации из констант и переменных возникают посредством применения к ним комбинаторов – совершенно небольшого числа констант специального вида. Остальные логико-математические средства строятся исключительно на их основе. Конечно, требовательного математика тут же заинтересуют вопросы возможности построения оснований математики и, следовательно, всего математического знания, но к настоящему времени сложности это процесса известны, а поиски путей их преодоления продолжаются и это все еще остается открытым вопросом.

Развитие комбинаторной логики. Имеется большое число книг, в которых вопросы развития комбинаторной логики так или иначе освещаются. Нужно отметить, что в них обычно речь идет о формальной системе, родственной комбинаторной логике – лямбда-исчислению, в котором существенно используется идея связывания переменных оператором функциональной абстракции. Практически ни в одной из имеющихся книг, ни тем более в многочисленных статьях на эту тему не дается действительно элементарного изложения ни комбинаторной логики, ни лямбда-исчисления, хотя некоторые из них на эту роль и претендуют. Кроме того, абсолютное их большинство издано на английском языке. Все это свидетельствует о высоком интересе к методам комбинаторной логики и о еще большей заинтересованности специалистов привлечь внимание талантливых и математически одаренных молодых людей, чтобы развивать эту дисциплину в целом спектре направлений.

Особенности изучения комбинаторной логики. Есть много причин, почему нужно изучать, развивать и разрабатывать комбинаторную логику, искать ее новые приложения и применения. Назовем среди них самые важные: основными разрабатываемыми в ней понятиями являются

представления об объекте и функции. На сегодня не имеется общепринятой теории объектов, рассматриваемых как математические, то есть идеализированные сущности. Хотя о функциональном анализе и теории функций часто говорят как о завершенных дисциплинах с хорошо осмысленным и разработанным кругом вопросов и представлений, но при попытках их применения, в особенности, в бурно растущей области информационных технологий, сразу наталкиваются на узкие места, ограничения и недостаточность их арсенала изобразительных средств. Так что обе эти дисциплины, несомненно, остро нуждаются в их развитии и адаптации к нуждам информационных технологий. Комбинаторная логика, несомненно, является теорией функций, а ее особенность заключается в том, что она оказалась адекватной информационным технологиям, все больше претендуя на роль их фундаментальных оснований. Тому имеются хорошо известные подтверждения.

Что же дает комбинаторная логика? Часто задается вопрос, что же дает комбинаторная логика? Прежде всего, концептуально ясную картину конструирования объектов как информационных сущностей, а также возможности изучения динамики их поведения. По-видимому, такая дисциплина, как динамика объектов ожидает своего часа для своего возникновения и применения в информационных технологиях. Необходимость в этом считается давно назревшей, но все еще не выраженной с должной требовательностью и последовательностью. Изучение комбинаторной логики позволяет понять структуру современного компьютеринга, в который составной частью входят компьютерные науки. Действительно, совсем немногие отчетливо осознают первостепенную роль конструктивного структурирования накопленного математического знания. По-видимому, совершенно недостаточно окажется организовать глобальную базу математического знания, составленную преимущественно из публикаций на эту тему, представленных в известных цифровых форматах. С неизбежностью возникают вопросы установления связей между отдельными фрагментами этого знания, но как это практически осуществлять – остается открытым вопросом. Хотя в области онтологий и таксономических форм установления структуры знаний имеются определенные продвижения.

В современных информационных системах с базами данных все еще идут по традиционному пути отделения данных от вычислений. По мере роста объема хранимых данных вычисления становятся своего рода инородным телом, которые только по необходимости приходится выполнять над данными. Считается, что наращивание объема данных само по себе ведет к информированности, и в настоящее время плоды этого распространенного заблуждения приходится пожирать в полной мере. Объем накопленных данных столь велик, что обнаружить среди них нетривиальную связь становится делом сложного, дорогостоящего и трудоемкого исследования, безусловно требующим написания вычислительной части в виде программного обеспечения, что является творческим актом.

В отличие от такого подхода, комбинаторная логика предлагает однородную трактовку данных и вычислений: и те, и другие являются объектами. Тем самым вычисления не отделяются от данных, становясь атрибутивными. Это подчеркивает идею фундаментальности компьютеринга, информационных процессов, естественный характер которых был установлен только в новейшее время, в особенности при изучении сложных молекулярных структур органического мира.

Комбинаторная логика и программирование. Говоря о приложениях комбинаторной логики в программировании, следует иметь в виду, что становление программирования как дисциплины происходило в технологическую эпоху до появления глобальной сети Интернет. Развитие программирования шло от математики и обслуживания расчетно-вычислительных задач, характеризующихся большим объемом арифметических операций. Традиции использования переменных повлияли на концепцию алгоритмических языков, основанную на вложенных блочных структурах, вызовах подпрограмм и связанных переменных. Набор инструкций фиксировался заранее, а использование переменных могло приводить к возникновению нежелательных побочных эффектов. При этом предполагалось, что основной объем как данных, так и знаний находится вне

компьютера, их носителем является именно человек, который применяет свой интеллект для создания того или иного алгоритма.

Попытки более интенсивного вовлечения компьютеров в подобную деятельность сначала привели к появлению баз данных, применявшихся для накопления того, что понималось под данными. Идея данных представлялась интуитивной ясной, о проблемах же, вызванных разрастанием объема баз данных до поры не заботились. Базы данных с успехом применялись для управления запасами, кадрами, финансами, а их функционирование обеспечивалось администраторами баз данных. На их плечи ложилось решение задачи организации взаимодействия баз данных с внешним миром. Такую задачу и таким способом можно было решать, пока влияние внешних факторов на базу данных оставалось сравнительно малым, а структура базы оставалась либо стабильной, либо мало меняющейся. В этих условиях возникла реляционная модель данных, которая посредством многосортной логики предикатов первого порядка объясняла механизмы накопления, модернизации и преобразования данных, а также удовлетворяла нуждам автоматизации извлечения ответа на запрос, касающийся связи между определенными наборами данных. Сложилась теория баз данных, основанная на этих принципах и при очень важном предположении о том, что объем оперативной памяти компьютера относительно мал, а сама она дорогостояща, зато объем внешней памяти велик и недорог.

С началом резкого удешевления компьютеров, упрощения их производства и налаживания выпуска дешевых модулей оперативной памяти большого объема эти предположения перестали работать, а концепция баз данных стала меняться в сторону все большего привнесения в сферу данных хранимых процедур, предназначенных для их обработки. Все это вызывало неудовлетворенность состоянием дел в области разработки систем управления базами данных и предоставляемыми ими возможностями, которые довольно далеко отстояли от повседневных практических потребностей.

Настоящий же кризис в теории баз данных разразился с появлением Интернет и началом массового выхода пользователей в глобальную сеть. Туда же, вслед за пользователями, стали перемещаться и объемы хранимой информации, и процедуры их обработки. Кризис на деле оказался значительно глубже и масштабнее, он, прежде всего, затронул само программирование и способы мышления программистов. Программирование перестало быть уделом круга искусственных интеллектуалов, а стало массовой профессией, обслуживающей потребности бурно разрастающейся индустрии информационных технологий. Под разрастанием объемов мультимедийной информации, буквально наводнившей интернет, стало изменяться и представление о данных и способах их обработки и использования. Индустрия данных стала во многом индустрией развлечений, унаследовав многие ее черты и особенности. От решения принципиальных вопросов, имеющих научную ценность, интерес сместился к повседневному потреблению информации, потоки которой продолжают расти. Для осмысления информации потребовались специальные инструментальные средства, осуществляющие поиск и селекцию релевантной информации пользователям информации из гигантских хранилищ глобальной сети Интернет. Информация более, чем когда либо, стала определять деловой успех и сама стала предметом беспрецедентной по своей интенсивности купли-продажи.

Пользователи не хотят больше просто ждать, когда программисты снабдят их готовыми инструментами, а проявляют и желание, и высокую готовность самостоятельно конструировать ту информационную реальность, которая им по вкусу. Возникли информационные сервисы, позволяющие конструировать полезные приложения без особых усилий на написание и отладку программ. Сервисы рассматриваются как самостоятельные строительные блоки, естественно, свободные от побочных эффектов, пользуясь которыми нетрудно организовать интересующее пользователя информационное обслуживание. Способ комбинирования сервисов оказался достаточно простым, что гарантировало работоспособность результирующей конструкции.

Развитие сервисов и сервисно ориентированных архитектур (SOA) дает новый стимул применению методов комбинаторной логики в информационных технологиях. Она вновь оказывается своего рода асимптотой, к которой стремится развитие информационных технологий. В системы программирования стали усиленно встраиваться существенно аппликативные возможности – функциональная абстракция, каррирование, функциональный стиль мышления и встраивания функций в программный код. Функциональный стиль программирования обретает заслуженное внимание, а его доля в общем объеме написанного программного кода понемногу, но стабильно растет.

Комбинаторная логика и информационные технологии. Комбинаторная логика долгое время служила уделом математиков и специалистов в теоретических компьютерных науках. Описанная – и не раз, – история ее развития обычно отражает становление чисто математических идей и взаимодействия с математической логикой и решением ее фундаментальных вопросов. Все еще на изучена и не написана история взаимодействия комбинаторной логики с информационными технологиям и их взаимного влияния. Возможно, просто потому, что соответствующая этому кругу вопросов дисциплина находится в начальной стадии своего развития.

В нее входят такие направления, как взаимодействие с программированием, базами данных и базами знаний. Не лишне отметить, что постепенно центральными разрабатываемым понятием во всех этих областях становится представление об объекте. Правда, эти представления, взятые из различных направлений, не совпадают во всех деталях и не обладают надежным математическим основанием, но имеющееся пересечение, фактически, идентично представлению об объекте в аппликативных вычислительных системах, среди которых лямбда-исчисление и комбинаторная логика.

Для того, чтобы справиться с возрастающей требовательностью к строгости и математическому основанию со стороны компьютерного сообщества, предпринимались различные попытки развития специализированных вариантов этих исчислений и, в первую очередь, лямбда-исчисления. Несмотря на определенный успех на начальной фазе, эти диалекты базовых формализмов не получают массового принятия. Прежде всего, их продвижение тормозится громоздкостью предлагаемых построений, которая, с позиций оценки выразительных возможностей, не дает принципиального продвижения или хотя бы такого технологического преимущества, ради которого стоило бы потратить значительные усилия на изучение их основ и овладение практическими методиками. В последнем случае дело обстоит особенно неблагоприятно, а об элементарном изложении результатов обычно говорить и вовсе не приходится.

Изменение представлений об объекте. К сожалению, к настоящему времени все еще не возникло теории объектов, обслуживающей компьютеринг и компьютерные науки и удовлетворяющей основные потребности отдельных дисциплин и направлений. В особенности, это относится к информационным технологиям. Программирование требует своего переосмысления и в условиях массового применения глобальной сети. Нужно создать адекватные именно ей информационные технологии, а не просто трансформировать уже известные для работы в изменившихся условиях. Произошли столь радикальные изменения, что предстоит научиться работать с частичными элементами и частично определенными функциями. Другим вызовом является нарастающее распространение информации чисто виртуального характера. До сих пор не было столь массового появления виртуальных объектов и не накоплен ни опыт работы с ними, ни математическое понимание сути происходящего. Университетское образование не закладывает основы для возможного продуцирования новых информационных технологий, а все это, по сути, отдано на откуп транснациональным корпорациям, делящим глобальный информационный рынок и обеспечивающим его компьютеринговое обслуживание. Вместе с тем только несколько большее, чем обычно, уделение внимания методам аппликативных вычислительных систем, дало бы свои плоды. По существу, все объекты комбинаторной логики, за исключением немногих базисных констант, являются виртуальными, а диапазоны, по которым пробегают переменные, потенциально ничем



не ограничены. Вопрос существования объектов, или указания действительных объектов является для нее естественной задачей, подход к решению которой известен. Заодно это позволяет глубже проникнуть в понимание природы вычислений, которая, как с удивлением выяснилось, практически не изучена. Сам формализм комбинаторной логики является синтаксически ясным, наипростейшим из известных и, при сохранении выразительных возможностей базового варианта, достаточным для построения обширной части оснований математического знания. Даже первые попытки изучения возможностей этого формализма обогащают интуицию пониманием того, что компьютеринг является не просто суммой технологий, но имеет под собой фундаментальное основание. Комбинируя простые сущности, удается построить видимое многообразие того, что является содержанием информационных технологий. Другими словами, конструируются путем комбинирования базисных форм объекты данных, объекты метаданных, обслуживающие их операции, сопутствующие им алгоритмы. Такое единство комбинаторной логики выглядит несколько неожиданным и, возможно, незапланированным ее создателями.



Дисциплина компьютеринга и потребности общества. Ничего похожего не дает ни одна из дисциплин, предложенных в современных рекомендациях по преподаванию в университетах компьютерных наук, программной инженерии, информационных технологий. Все они вместе взятые создают общее представление о разнохарактерности отдельных дисциплин компьютеринга, непроницаемости понятийных перегородок, возникающих и укрепляющихся между ними и концептуальной независимости развития каждой из них. Отсюда ведет свое начало узкая специализированность и потребность в пожизненном и непрерывном переквалификации занятых в информационных технологиях и компьютеринге. Все это гипертрофированно увеличивает нагрузку на образовательные учреждения, а возрастание доли времени, затрачиваемой на всевозможные переобучения и переквалификации становится заметным тормозящим фактором в развитии самих информационных технологий, удешевлении и повсеместности предоставляемых ими услуг и возможностей. Похоже, что время систематических, перманентных модернизаций распространенных и общепринятых дисциплин и норм университетского образования в области компьютеринга подходит к своему завершению, теряя свою эффективность. По видимому, более продуктивным будет более значительное смещение парадигмы, позволяющее с единых позиций рассматривать все многообразие форм проявлений компьютеринга и информационных технологий. Экономия усилий при этом существенна и окажется заметной во всех сферах экономики.

Нельзя сказать, что компьютерное сообщество бездействует в этом отношении. В мировой практике сложились системы обучения идеям аппликативных вычислений на основе систем функционального программирования, которые распространяются бесплатно и развиваются и разрабатываются коллективными усилиями. В этой области накоплен значительный потенциал для более решительного продвижения на рынок информационных технологий, что в последнее время становится настолько заметно, что с этим не могут не считаться и крупнейшие мировые производители программного обеспечения различного назначения, включая инструментальные средства и отдельно взятые системы программирования. Эти усилия начинают все больше складываться, готовя и определяя изменения уже в недалеком будущем.

Благодарности. Содержание книги обсуждалось в связи с подготовкой и проведением конференции “Аппликативные вычислительные системы” (ABC’2008) (см. URL <http://http://jurinfor.exponenta.ru/ACS2008/>). Многие из высказанных пожеланий учтены, но часть из них осталась нереализованной, поскольку потребовались бы радикальные изменения, что просто требует написания совершенно иной книги.

Выражаю признательность проф. Х. Барендрегту, проф. Р. Хиндли и проф. Дж. Селдину за обсуждение ряда вопросов, связанных с аппликативными вычислительными технологиями. В особенности это касается идей о связях естественного компьютеринга, многообразия известных моделей вычислений и комбинаторной логики.

Многократные обсуждения с С. В. Косиковым связей компьютеринга, аппликативных вычислительных систем и теории категорий сыграли двойную роль: в ответ им была написана книга [25], а подготовка нового, 3-го издания пошла более быстрыми темпами.

Искренняя признательность выражается А. Д. Лаптеву, В. Н. Назарову, В. В. Рословцеву, с которыми обсуждались отдельные идеи компьютеринга, нашедшие отражение в настоящей работе.

Настоящее издание книги “Комбинаторная логика в программировании” смогло быть подготовлено совместными усилиями и лишь благодаря массе времени и энергии, затраченной сотрудниками НОУ “Институт Актуального Образования “ЮрИнфоР-МГУ”. Особенно большую работу проделал А. Е. Зайцев, его творческое отношение способствовало тому, что издание книги приобрело настоящий вид.

Список литературы

1. Barendregt H., Wiedijk F. *The Challenge of Computer Mathematics*. – Transactions of the Royal Society, Vol. 363, №1835, 2005. – pp. 2351-2375.
<ftp://ftp.cs.ru.nl/pub/CompMath.Found/Barendregt-Wiedijk.pdf>
2. Bell G., Dourish P. *Yesterday's tomorrows: notes on ubiquitous computing's dominant vision*. – Personal Ubiquitous Comput., Vol. 11, №2, 2007. – pp. 133-143.
DOI <http://dx.doi.org/10.1007/s00779-006-0071-x>.
3. Berners-Lee T., Hall W., Hendler J., O'Hara K., Shadbolt N., and Weitzner D. *A framework for Web science*. – Foundations and Trends in Web Science, Vol. 1, Issue 1, 2006. – pp. 1-130.
<http://www.nowpublishers.com/web/>
4. Cardelli L., Davies R. *Service combinators for Web computing*. – HP Labs Technical Reports SRC-RR-148, June 1, 1997. – 15 p.
<http://www.hpl.hp.com/techreports/Compaq-DEC/SRC-RR-148.html>
5. Carpenter B. *The Internet Engineering Task Force: Overview, Activities, Priorities*. – ISOC BoT, 2006-02-10, 2006.
<http://www.isoc.org/isoc/general/trustees/docs/Feb2006/IETF-BoT-20060210.pdf>
6. Curry H. B. *Functionality in combinatory logic*. – Proc. National Academy of Sciences of the USA, Vol. 20, 1934. – pp. 584-590.
7. de Bruijn N. G. *Lambda-calculus notations with nameless dummies: a tool for automatic formula manipulation*. – Indag. Math. 1972, №34, pp. 381-392.
8. de Bruijn N. G. *A survey of the project Automath*. – In: To H.B. Curry: Essays in combinatory logic, lambda calculus and formalism, Academic Press, 1980. – pp. 579-606.
9. Denning P. J. *Computing is a natural science*. – Commun. ACM, Vol. 50, №7, 2007. – pp. 13-18.
DOI <http://doi.acm.org/10.1145/1272516.1272529>
10. Hindley J. R., Lercher B., Seldin J. P. *Introduction to Combinatory Logic*. – London: Cambridge University Press, 1972.
11. Kennedy A. *Functional Pearls: Pickler Combinators*. – Journal of Functional Programming, special issue on Functional Pearls, Vol. 14, №6, Cambridge University Press, Nov 2004. – pp. 727-739,
12. MacLennan B. J. *Molecular Combinator Reference Manual*. – UPIM Report 2, Technical Report UT-CS-02-489, Department of Computer Science, University of Tennessee, Knoxville, 2002. – 16 p.
<http://www.cs.utk.edu/~mclennan/UPIM/CombRef.pdf>
13. MacLennan B. J. *Combinatory Logic for Autonomous Molecular Computation*. – Preprint of paper invited for Information Sciences, 2003.
<http://www.cs.utk.edu/~mclennan/UPIM/CLAMC-IS.pdf>
14. MacLennan B. J. *Molecular Combinatory Computing for Nanostructure Synthesis and Control*. – IEEE Nano 2003, San Francisco, August 12-14, 2003.
<http://www.cs.utk.edu/~mclennan/UPIM/MacLennan-MCCNSC.pdf>
15. Schönfinkel M. I. *Über die Bausteine der mathematischen Logik*. Math. Annalen 92, 1924. – pp. 305-316.
16. Scott D. S. *The lattice of flow diagrams*. – Lecture Notes in Mathematics, 188, Symposium on Semantics of Algorithmic Languages. – Berlin, Heidelberg, New York: Springer-Verlag, 1971, pp. 311-372.
17. Scott D. S. *Relating theories of the lambda calculus*. – Hindley J., Seldin J. (eds.) To H.B. Curry: Essays on combinatory logic, lambda calculus and formalism. – N.Y. & L.: Academic Press, 1980, pp. 403-450.
18. Seldin J. P. *The Logic of Curry and Church*. – In: (Dov Gabbay and John Woods, eds.) *Handbook of the History of Logic*, Vol. 5, Elsevier, 2006. <http://people.uleth.ca/~jonathan.seldin/CCL.pdf>
19. Selinger P. and Valiron B. *A lambda calculus for quantum computation with classical control*. – Mathematical Structures in Computer Science, 16(3), 2006. – pp. 527-552.
20. Вольфенгаген В. Э. *Конструкции языков программирования. Приемы описания*. – М.: АО «Центр ЮрИнфоР», 2001. – 276 с. Издание поддержано грантом РФФИ, проект 01-01-14068-д.
21. Вольфенгаген В. Э. *Комбинаторная логика в программировании. Вычисления с объектами в примерах и задачах*. – М.: МИФИ, 1994. – 204 с.; 2-е изд., М.: АО «Центр ЮрИнфоР», 2003. – 336 с.
22. Вольфенгаген В. Э. *Методы и средства вычислений с объектами. Аппликативные вычислительные системы*. – М.: JurInfoR Ltd., АО «Центр ЮрИнфоР», 2004. – xvi+789 с.
Издание поддержано грантом РФФИ, проект 03-01-14055-д. 1
23. Исмаилова Л. Ю. *Логика объектов*. – В кн. [22], с. 613-630.
24. Косиков С. В. *Логика функциональности*. – В кн. [22], с. 595-612.
25. Косиков С. В. *Информационные системы: категорный подход*. – Под ред. Л. Ю. Исмаиловой. – М.: «ЮрИнфоР-Пресс®», 2005. – 96 с. 1
26. Шаумян С. К. *Аппликативная грамматика как семантическая теория естественных языков*. – М.: Наука, 1974. – 204 с.

четверг, Август 21, 2008 12:21