

МАТЕМАТИКА КАК ИНОСТРАННЫЙ

или архитектура математической мысли

Н. И. Казимиров

Москва, 2026

УДК 510.6
ББК 22.12
К14

К14 **Казимиров Н. И.**
Математика как иностранный, или архитектура математической мысли — М., 2026. — 354 с.

Книга представляет собой попытку автора написать пособие по изучению математического языка так, как принято изучать иностранный язык. Целью книги является научить читателя правильно и легко воспринимать математический текст, понимать его закономерности, структуру, возможности и ограничения. Ориентированная на изучение математического текста как такового, книга в основном апеллирует к формальным системам, которые изучаются в математической логике и примерам классических формальных теорий. В то же время в ней постоянно подчеркивается многоуровневый взгляд на математический язык, включающий метатеорию и семантику формальных конструкций.

Ключевые слова: язык, формулы, логика, формальные теории, основания математики, теория множеств.

СОДЕРЖАНИЕ

Ключевые теоремы	7
Предисловие	9
I Восхождение	15
Уровень A1. Выживание	17
A1.1 Иллюстративная сказка	17
A1.2 Принцип матрешки	18
A1.3 Семантика vs Синтаксис	25
A1.4 Рефлексия	28
A1.5 Термины	30
A1.6 Переменные и свертки	31
A1.7 Особенности математического языка	34
A1.8 Вопросы для самоконтроля	40
A1.9 Упражнения	40
Уровень A2. У чертога логики	43
A2.1 Понятие	46
A2.2 «Истина» и «ложь»	51
A2.3 Логика и множества	52
A2.4 Связь логики и семантики через объем понятия	56
A2.5 Логические рассуждения	58
A2.5.1 Понимание логических связей	58
A2.5.2 Принцип сведения	60
A2.5.3 Кванторы	61
A2.5.4 Про импликацию	63
A2.6 Построение выводов	65
A2.6.1 Силлогизмы и абдукция	65
A2.6.2 Модусы	69
A2.6.3 Дедукция, индукция	70
A2.7 Про отношения и функции	72
A2.8 Вопросы для самоконтроля	76
A2.9 Упражнения	77
Уровень B1. Формальная логика	81
B1.1 Логика высказываний	81
B1.1.1 Язык логики высказываний	81
B1.1.2 Аксиомы и правила вывода	83
B1.1.3 Основные факты о логике высказываний	86
B1.2 Логика предикатов	89

B1.2.1	Термы	90
B1.2.2	Формулы	91
B1.2.3	Аксиомы и правила вывода логики предикатов	93
B1.2.4	Примеры теорий	95
B1.2.5	Конгруэнтность равенства	97
B1.2.6	Модель языка и теории	98
B1.2.7	Примеры интерпретации	99
B1.2.8	Некоторые понятия теории моделей	103
B1.2.9	Основные факты о логике предикатов	103
B1.2.10	Еще раз о теории равенства	107
B1.3	Подытог и вопросы для самоконтроля	109
B1.4	Упражнения	111
Уровень B2. Фундамент математики		114
B2.1	Арифметика Пеано	114
B2.1.1	Язык и аксиомы	115
B2.1.2	Порядок	120
B2.1.3	Варианты индукции	121
B2.1.4	Доказуемость и выразимость	123
B2.2	Прологомены к теории множеств	125
B2.3	Наследственно-конечные множества	133
B2.3.1	Язык и аксиомы	133
B2.3.2	Скобочная модель	139
B2.4	Вопросы для самоконтроля	143
B2.5	Упражнения	143
Уровень C1. Рай для математиков		145
C1.1	Теория множеств Цермело-Френкеля	145
C1.1.1	Аксиоматика ZFC	145
C1.1.2	Отношения, функции и классы	148
C1.1.3	Зачем нужны функции	152
C1.2	Аксиома выбора	153
C1.3	Универсум фон Неймана	162
C1.4	Конструкции	170
C1.4.1	Структуры	170
C1.4.2	Принципы структуростроения	172
C1.4.3	Конкретный разговор о моделях	175
C1.5	Итоги и вопросы для самоконтроля	182
C1.6	Упражнения	184
II Доказательная база		187
Глава D1. Язык		189
D1.1	Элементарные частицы языка	189
D1.2	Лексемы	192
D1.3	Синтаксический разбор	197
D1.4	Упрощения	199

D1.5	Графика	200
D1.6	Определение переменных	202
D1.7	Упражнения	204
Глава D2.	Логика	208
D2.1	Логика высказываний	208
D2.2	Логика предикатов	216
D2.2.1	Корректность и полнота исчисления	218
D2.2.2	Выразимость	222
D2.2.3	Категоричность	223
D2.2.4	Полнота теории	226
D2.2.5	Сводка результатов	231
D2.3	Упражнения	231
Глава D3.	Арифметика	234
D3.1	Основные свойства	235
D3.2	Индукция	246
D3.3	Элементарная теория делимости	248
D3.4	Кодирование последовательностей	253
D3.5	К теоремам Гёделя и Тарского	260
D3.6	Упражнения	263
Глава D4.	Теория множеств	266
D4.1	Основные свойства	267
D4.2	Конечные и бесконечные множества	272
D4.2.1	Сравнение мощностей	273
D4.2.2	Конечные множества	276
D4.3	Ординалы и кардиналы	281
D4.3.1	Рекурсия и индукция	282
D4.3.2	Иерархия фон Неймана и ранг	287
D4.3.3	Арифметика ординалов	288
D4.3.4	Арифметика кардиналов	299
D4.4	Аксиома выбора	302
D4.4.1	Счетный выбор и его следствия	303
D4.4.2	Аксиома выбора и ее эквиваленты	305
D4.5	О взаимосвязи формальных теорий	309
D4.6	Упражнения	311
Ответы к упражнениям		317
Предметный указатель		341
Список обозначений		345
Список источников		349

Ключевые теоремы

Теорема D2.5 (о дедукции)	216
Теорема D2.8 (о корректности вывода)	219
Теорема D2.9 (Гёделя о полноте логики предикатов)	220
Теорема D2.11 (Лёвенгейма-Скулема о понижении мощности)	223
Теорема D2.12 (о компактности, Гёделя–Мальцева)	225
Теорема D2.14 (Лёвенгейма-Скулема о повышении мощности)	226
Теорема D2.15 (Мёрли о категоричности)	226
Теорема D2.18 (Тарского-Зайденберга)	228
Теорема D3.8 (основная теорема формальной арифметики)	258
Теорема D3.9 (основная теорема арифметики)	259
Теорема D3.10 (лемма о диагонализации)	260
Теорема D3.11 (Гёделя первая о неполноте)	261
Теорема D3.12 (Гёделя вторая о неполноте)	262
Теорема D3.13 (Тарского о невыразимости истины)	262
Теорема D4.2 (Кантора)	273
Теорема D4.3 (Кантора-Бернштейна-Шрёдера)	274
Теорема D4.6	282
Теорема D4.7 (о ранге)	287
Теорема D4.13 (эквиваленты аксиомы выбора)	305

Дорогой читатель,

Данная книга представляет собой целостное и структурированное собрание материалов, разработанных мною для курса по изучению математики как иностранного языка. Эти наработки поначалу нашли свое место в моем блоге на [Youtube](#), а затем были положены в основу данного издания.

Отметим сразу же несколько особенностей книги, которые существенно отличают ее как от учебников по математике, так и от учебников иностранных языков.

Во-первых, цель книги состоит не в том, чтобы научить читателя именно математике или математической логике, хотя она изобилует примерами именно из этих дисциплин. Напротив, цель состоит в том, чтобы показать, насколько прозрачным и понятным может быть *lingua franca* современных точных наук, если пройти определенный курс адаптации — путешествие, в котором эта книга призвана стать вашим проводником.

Особенно актуальным это становится сегодня, когда на наших глазах появляется новый исторический актер, для которого математический язык является родным, — *искусственный интеллект*. Алгоритмы, управляющие роботами, нейронные сети, анализирующие данные, и большие языковые модели строят свои «мысли» и «решения» на строгих законах логики и алгебры, которые и составляют грамматику языка математики. Понять их — значит заглянуть в будущее коммуникации не только с машинами, но и с самой формализованной мыслью.

Во-вторых, эта работа задумана прежде всего как книга для дополнительного чтения, а не как традиционный учебник. Язык математики глубоко отличается от любого разговорного языка, и уровни владения им определяются критериями, отличными от стандартных шкал языковой компетенции. Поэтому эту книгу лучше всего использовать, чтобы перекинуть мост между интуитивным пониманием и строгим формализмом, дополняя стандартные курсы логики, теории множеств или алгебры.

В-третьих, книга не претендует на точное научное описание лингвистических особенностей и явлений, которые так или иначе проявляются в математическом языке и которые могут иметь соответствующую коннотацию и терминологию в лингвистике. Поэтому те места, где изложение материала выходит за рамки математики и логики, следует считать скорее сопутствующим художественным описанием, чем выверенным научным текстом. Однако я как автор был бы признателен, если бы опытный лингвист или филолог помог мне в будущем разобраться в правильном употреблении терминологии и необходимых отсылках к научным источникам.

В-четвертых, особое внимание уделяется *анатомии доказательств*. Текст предлагает уровень детализации, часто опускаемый в стандартных учебниках, где технические нюансы нередко оставляются читателю. Например, доказательство возведения ординалов в степень через функции с конечным носителем представлено с исчерпывающей тщательностью. Более того, мы явно анализируем связь между различными определениями ординальных арифметических операций — рекурсивными и основанными на упорядоченных суммах, произведениях и степенях. Эта двойственность подходов редко представлена столь полно в учебной литературе, однако она необходима для глубокого понимания того, как строятся математические структуры. Подобно тому как беглость в иностранном языке требует овладения тонкостями грамматики, беглость в математике требует видения микроскопических деталей её логических оснований.

Необходимо также сказать о глубине изложения. Хотя мы стремимся к строгости, главная цель этой книги — продемонстрировать *архитектуру* математической мысли. Вследствие этого, для некоторых монументальных результатов — таких как теоремы Гёделя о неполноте, теорема Морли о категоричности или результаты о независимости в теории множеств — мы ограничиваемся **концептуальным обсуждением** их значения и последствий, а не приводим формальные выводы. И наоборот, мы уделяем скрупулезное внимание «микроскопическим» основам, часто опускаемым в стандартных текстах: например, строгий вывод элементарных арифметических свойств из аксиом Пеано представлен во всех деталях. Кроме того, мы делаем акцент на теоретико-модельном подходе к доказательствам в логике предикатов, подчеркивая семантическую связь между языком и его моделями.

Структура книги имитирует этапы изучения языка, что отражено в названиях глав. Её содержание дополняет стандартную математическую литературу, фокусируясь на описательном аспекте математики, выраженном через формализмы и модели.

Текст книги разделен на две части, в первой из которых превалирует «лирика», т. е. некоторые общие рассуждения и факты без доказательств, а во второй, «брутальной» части внимание читателя сконцентрировано на прямом применении изучаемого языка в конкретных примерах, где преобладает математическая логика. Соответственно, в первой части книги, по крайней мере в начале, очень мало формул и довольно много нестрогих ассоциативных рассуждений, а во второй, наоборот, страницы усеяны формальными выкладками, напоминая своим видом текст компьютерной программы, и почти отсутствуют общие рассуждения.

Кроме того, первая часть книги разделена на блоки, условно соответствующие уровням сложности математического языка. Традиционно, это уровни **A1**, **A2**, **B1**, **B2**, **C1**, каждый из которых затем делится на еще более детальные разделы или модули, сложность которых внутри одного уровня никак не ранжируется. Материал часто рассматривается повторно на разных уровнях с возрастающей строгостью, отражая «спиральный подход» к изучению языка. Уровень профессионального владения **C2** опущен, так как такое мастерство характеризует «носителей языка» — людей с профессиональным математическим образованием, которым эта книга по определению не нужна.

Эта модульная структура допускает нелинейный подход к чтению. Подобно тому как используют учебник иностранного языка — выбирая подходящий уровень сложности и пропуская уже освоенные уровни, — читателю предлагается пропустить элементарные разделы, если они кажутся тривиальными, и переходить непосредственно к более продвинутым темам. Вы вольны выбирать отправную точку, соответствующую вашему текущему уровню «владения языком», возвращаясь к ранним главам только за конкретными разъяснениями.

Часть II не следует уровневой структуре. Она организована по ключевым темам — Язык, Логика, Арифметика и Теория множеств — для удобного закрепления необходимых утверждений и определений, на которые ссылается Часть I.

С самого начала текста предполагается, что читатель уже в некоторой степени знаком с математической символикой хотя бы из первых 6–7 лет общеобразовательной школы, так что здесь мы не занимаемся изучением цифр, не учим таблицы сложения и умножения, не объясняем аксиомы планиметрии и суть координатного метода Декарта, относя все эти простейшие вещи к уровню **A0**, который еще называется Beginner или Starter.

Тем не менее, начинаем мы наше пособие, как и положено, с изучения алфавита, поскольку все дальнейшие рассуждения существенным образом опираются на этот при-

митивный базис любого письменного языка. Собственно, весь раздел **A1** в значительной мере посвящен тому, чтобы рассказать, как же на самом простом уровне устроен математический текст, т. е. каким образом и по каким правилам складываются те замысловатые формулы, которые так часто отпугивают непосвященных. Еще не понимая полностью смысла текста, записанного на математическом языке, вы уже сможете его читать и понимать его внутреннюю структуру. Иначе говоря, в этом разделе мы рассматриваем синтаксическую структуру математического языка. С моей точки зрения, отсутствие «правил чтения» математического языка в учебниках математики является одним из огромных пробелов современного естественно-научного образования. Уровень **A1** по аналогии с обычными учебниками иностранных языков я бы назвал *уровнем выживания* или *Elementary*.

На уровне **A2** мы затрагиваем базовые понятия логики в том виде, в каком ее ранее преподавали как отдельный предмет. И хотя теперь считается, что логика есть часть математики, мы будем придерживаться той парадигмы, что существует некая общая научная логика, присущая человеческому разуму как некоторый базовый код, позволяющая строить выводы и прогнозы вне зависимости от знаний математики, а есть логика (точнее, много разных логик) формальная, анализом которой как раз и занимается математика, и которая лежит в основе формальных математических и компьютерных систем. Вместе с логикой мы привносим в наш язык некоторую семантику, т. е. смысловую окраску тех бесчисленных закорючек, которые мы уже научились писать и читать. На этом уровне мы уже существенно продвигаемся в освоении языка, изучаем такие базовые вещи, как рассуждения и вывод, вводим начальную терминологию и обозначения из теории множеств, даем определения различных видов отношений и функций, вводим обозначения логических операций и кванторов. Уровень **A2** можно обозначить как *предпороговый* или *Pre-Intermediate*.

На уровне **B1** мы переходим уже к формальной логике, детализируем ее описание, выделяем несколько слоев логики — исчисление нулевого порядка (логика высказываний), исчисление первого порядка (логика предикатов), формальные аксиоматические теории первого порядка. Некоторое внимание уделяется также теории моделей и интерпретаций формального языка и формальных теорий, понятию непротиворечивости, полноты, логического следования. Язык вместе с тем становится более строгим, определения более точными. Немало места уделяется значимости понятия равенства и такого его определяющего свойства, как конгруэнтность. Вместе с тем мы продолжаем использовать «наивные» понятия множеств, отношений и функций, откладывая анализ формальной теории множеств на более высокий уровень. Здесь нашим главным достижением в изучении математического языка будет полное представление о том, что такое теория, аксиома, формальное доказательство и как все это можно «приземлить» на теоретико-множественную модель. Иначе говоря, здесь выученный ранее язык из простого умения говорить превращается в способ думать и грамотно излагать свои мысли. Уровень **B1** по аналогии с разговорными языками можно назвать *Intermediate* или *пороговый уровень*. На этом уровне количество готовится перейти в качество.

С началом уровня **B2** мы все больше вынуждены писать формулы вместо слов, поскольку начинаем преодолевать значительный с точки зрения восприятия порог сложности математического языка. А именно, мы знакомимся с формализмом первопорядковой арифметики Пеано, обсуждаем ее аксиоматику и некоторые следствия. Важными вехами на этом пути будут приобретаемые навыки обращения с формальными конструкциями, умение воплощать в формализме некоторые общие понятия и понимать границы применимости формализмов в плане способности выразить те или иные кон-

цепции. Здесь мы также вводим язык теории множеств, фокусируясь на финитной математике и представлении множеств в виде деревьев, подводя под аксиматику некоторую философскую мотивацию. Описание теории наследственно конечных множеств будет той вишенкой на торте, которая увенчает уровень **B2** и подготовит к штурму аксиоматической теории множеств Цермело-Френкеля. Уровень **B2** мы можем сравнить с Upper-Intermediate или *продвинутым пороговым уровнем* в освоении математического языка как иностранного.

Уровень **C1** предлагает тщательный анализ языка теории множеств. Здесь мы дадим окончательно строгие определения таким понятиям, как отношение и функция, детально проанализируем аксиому выбора в различных ее формах и ее место в математике, коснемся роли аксиомы регулярности в систематизации универсума множеств по фон Нейману, поговорим о способах построения структур и разберем их на конкретных примерах числовых структур. Здесь мы совершаем фазовый переход от простого формализма к языку высокого уровня, способному выражать современные математические концепции.

Если вы успешно пройдете этот путь и проберетесь через чащи формул во Второй части, я уверен, что любые опасения перед математическими текстами исчезнут. Вы приобретете навыки самостоятельного создания таких текстов и формулирования мыслей на уникальном языке, на котором, по выражению Галилео Галилея, написана книга природы.

По всему тексту ключевые понятия и обозначения при первом упоминании выделены **жирным шрифтом**. Следуя спиральному подходу к освоению языка, первоначальные объяснения в Части I опираются на интуицию и контекст, в то время как строгие математические определения приводятся по мере продвижения изложения к более высоким уровням или закрепляются в Части II.

Чтобы облегчить переход от пассивного понимания к активному владению материалом, в книгу включено около 190 упражнений различной степени сложности — от задач на вычисления до построения нетривиальных доказательств. Ответы приведены в конце книги.

Цифровое приложение

Чтобы помочь читателям ориентироваться в сложной сети математических понятий, я разработал интерактивный инструмент 3D-визуализации — **MathLogic Nexus**. Это веб-приложение позволяет пользователям исследовать логические зависимости между определениями, теоремами и теориями. В то время как книга фокусируется на фундаментальных лингвистических и логических основах, *MathLogic Nexus* помещает эти понятия в более широкий математический контекст, визуализируя их связи с теорией моделей, топологией и теорией вычислимости. Оно служит динамической картой «архитектуры» математической мысли. Инструмент доступен онлайн по адресу: <https://nexus.mathem.at/>, а его краткий обзор — на моем Youtube-канале.

Наконец, для удобства навигации в конце книги размещены подробный Предметный указатель и Список обозначений.

Благодарности

Эта книга — результат долгого личного путешествия по ландшафту математической логики, пути, который я по большей части прошел как независимый исследователь. Однако ни одно путешествие не бывает по-настоящему одиночным, и я в долгу перед теми, кто сформировал мое математическое мировоззрение.

Хочу выразить глубочайшее уважение и благодарность моему научному руководителю, профессору Юрию Леонидовичу Павлову, чье руководство в годы моей аспирантуры, посвященной случайным лесам и графам, привило мне дисциплину математической мысли.

Хотя мне не выпала честь быть его непосредственным учеником, я глубоко благодарен академику Льву Дмитриевичу Беклемишеву. Его блестящие лекции, опубликованные кафедрой математической логики МГУ, стали для меня главными воротами в современную теорию доказательств. Ясность его изложения служила ориентиром для моих собственных попыток объяснять сложные понятия.

Я также благодарен моим рецензентам за конструктивную критику. Их настойчивое пожелание расширить практическую составляющую книги побудило меня добавить упражнения в Часть II, что значительно повысило ее учебную ценность.

Примечание об издании

Данная книга изначально задумывалась и писалась на русском языке, однако её международная версия была принята к публикации издательством Springer под названием *Mathematics as a Foreign Language*. Настоящее русское издание представляет собой оригинальную авторскую версию, свободную от ограничений перевода и сохраняющую все стилистические особенности родного языка.

Москва, Россия
2026



YouTube



Telegram



mathem.at

Часть I

Восхождение

Выживание

Самое непостижимое в этом мире — это то, что он постижим.

— Альберт Эйнштейн

A1.1. Иллюстративная сказка

Представим себе, что Математика — это некая экзопланета, довольно-таки непригодно устроенная и населенная странными существами, назовем их *математянами*, средством общения которых являются замысловатые фигурные закорючки, которые они умеют чертить прямо в воздухе, благо состав атмосферы им это позволяет. Но в отличие от сюжета известного фильма «Прибытие» Дени Вильнёва, эти, с позволения сказать, «тексты» намного сложнее выглядят и одновременно имеют очень строгую логическую структуру. И вот на эту планету высаживается земной бот, оснащенный ИИ, и пытается наладить контакт с местными жителями. Для этого ему, конечно же, требуется найти с ними общий язык, т.е. по сути разобраться в их странном языке и научиться на нем что-то «говорить». Слово «говорить» мы берем в кавычки, потому что этому процессу не сопутствуют колебания воздуха, которые мы могли бы назвать звуком, а сам процесс имеет чисто визуальное сопровождение.

«Разумный» бот начинает применять свои сложные алгоритмы по анализу структур, помогающие распознать этот язык по какому-либо известному землянам шаблону. Первое, что он понимает: язык является и алфавитным, и иероглифическим одновременно, поскольку значительная часть элементов языка повторяется многократно, в то время как другая часть элементов встречается сравнительно редко.

С одной стороны, это проблема, поскольку не совсем понятно, какими лингвистическими методами пытаться разобрать этот язык на составляющие. С другой стороны, довольно быстро выясняется, что почти все иероглифические знаки коррелируют с алфавитными последовательностями, т.е., проще говоря, их можно расшифровать с помощью более длинного, но более простого текста. Это обнадеживает.

Следующая фундаментальная особенность языка математян состоит в том, что всякий отдельно взятый графический образ, выражающий какое-то абстрактное понятие, имеет определенную внутреннюю структуру в виде дерева-матрешки, т.е. складывается при помощи подстановки простых графем в некоторые шаблоны, разнообразие видов которых весьма ограничено. Тем самым становится понятно, как генерировать новые «слова» и «фразы» на местном языке.

Разобравшись с базисом и структурой языка, наш бот пытается искать правила, по которым одни конструкции допустимы, т.е. встречаются в замеченных им «разговорах» довольно часто, а какие категорически недопустимы (не встречаются) или допускаются крайне редко, что можно соотнести с их негативной или вульгарной коннотацией. Методом случайных деревьев решений (decision trees) или еще каким-то более продвинутым методом анализа структур ИИ нашего бота распознает ряд простых правил,

позволяющих строить правильные тексты и при этом как можно реже оскорблять чувства местного населения (как в песне Высоцкого про тау-китян).

Итак, синтаксис языка стал более-менее понятным, и теперь самое время для распознавания некоторой семантики языка, т.е. придания ему смысла. Для чего нужно пробовать полностью или фрагментарно интерпретировать корректно построенные тексты в какой-то уже известной нам семантической реальности. Например, отыскать в земных текстах такие фрагменты, которые очень похожим образом соотносятся друг с другом. Иначе говоря, если в языке математян всегда за одним конкретным выражением следует другое не менее конкретное, то и в земной интерпретации конкретные аналоги этих выражений тоже должны располагаться друг за другом в том же порядке с очень высокой долей вероятности.

Тут важно понимать, что интерпретации могут быть простыми и сложными. Простые интерпретации чаще всего довольно нелепы, однако они позволяют оценить саму возможность произвести интерпретацию чужого языка (и его относительную непротиворечивость). Вспоминается рассказ А. Азимова «Логика», в котором робот нового поколения, обслуживающий некое устройство трансляции энергии на спутнике, придумал религию, очень просто объясняющую необходимость держать луч в фокусе ретранслятора — он стал поклоняться устройству, называя его Господином, отменил всяческую возможность существования космоса, Земли и луча энергии, но зато как «отче наш» старался сделать так, чтобы стрелки приборов не отклонялись от нужных значений ни на волосок. При этом категорически отпала необходимость в людях, и он отстранил их от управления спутником, назвался пророком Господина, а из роботов-помощников сделал адептов этой религии. Что характерно, такая упрощенная метафизическая концепция позволяла очень точно и надежно исполнять то, что было предписано роботам изначально, совершенно не задумываясь о пользе народному хозяйству и какой-то там Земле. Он просто выбрал простой самодостаточный фрагмент языка управления транслятором и проинтерпретировал его еще более простым способом. Интерпретация была абсолютно ложной с точки зрения людей, но непротиворечивой и вполне работоспособной.

Итак, наш бот начинает придумывать интерпретации или, говоря более научно, строить модели того языка, который он открыл. Построение разнообразных моделей, имеющих вполне конкретный смысл для землян, позволяет хотя бы примерно оценить семантику математян, а также попытаться построить переводчик с нашего языка на их, и наоборот.

После этого можно смело утверждать, что земной бот, а вместе с ним и поджидающие его на орбите земляне нашли общий язык с аборигенами планеты Математика, и могут приступать к культурному обмену. Если, конечно, стандартная модель изученного языка не окажется настолько примитивной, что наши корреспонденты сочтут нас полными кретинами и прекратят общение.

A1.2. Принцип матрешки

Итак, в целом мы уже начертили путь, по которому нам предстоит пройти. Попробуем представить себя создателями того самого бота с ИИ, который изучает инопланетный язык, и применим наши наработки к изучению не такого уж инопланетного, но подчас более сложного и непонятного языка, каковым является язык математики.

Во-первых, для краткости будем обозначать этот язык буквой **Math**. Сразу же стоит оговориться, что **Math** включает в себя на самом деле целое родовое дерево языков и диалектов, которые в каждой конкретной ситуации определяются некоторым более строгим образом. Пока же для нас язык **Math** — это математический язык вообще, в самом общем понимании.

Во-вторых, заметим, что этот язык обладает многими признаками естественных языков, такими как: коммуникативность (поскольку обеспечивает возможность ученым из разных наук и стран обмениваться знаниями), наличие встроенного метаязыка (инструментов, позволяющих описывать себя самого), эстетические функции (выражающиеся во внутренней красоте универсальности и согласованности понятий), индикативность (указание принадлежности своих пользователей к определенным группам профессий), неограниченная семантическая мощь (поскольку математические понятия могут одинаково хорошо работать при совершенно различной семантике), эволютивность (математический язык быстро и плодотворно развивается, адаптируясь к новым реалиям науки и культуры).

В то же время по понятным причинам математический язык нельзя отнести к естественным языкам, поскольку он не имеет привязки к какой-либо этнической группе или языковой семье, а кроме того, он включает огромное семейство формальных языков, зачастую напоминающих языки программирования, что, конечно, резко отличает его от языков разговорных.

В-третьих, язык **Math** — это чисто письменный язык. Безусловно, в этом языке есть «озвучка», т.е. текст мы можем так или иначе прочесть вслух, но толку для слушателя от этого фонирования не будет никакого, поскольку математические формулы крайне трудно воспринимаются на слух, за исключением каких-то совсем уж простейших выражений. Эта особенность языка **Math** на первый взгляд должна упростить его изучение, ведь мы сразу же можем отбросить такие известные компоненты изучения языков как аудирование и говорение, оставив только чтение и письмо. Однако, следуя принципам диалектики, не стоит упускать из виду антагонистическую составляющую упрощения — схлопывание каналов восприятия, ведь теперь мы начисто отвергаем слуховую память и полагаемся только на зрение и логику (в отдельных случаях может помочь память микромоторики, но в математике она может сыграть злую шутку).

Как и любой другой письменный язык, наш язык **Math** начинается с **алфавита**. При этом нужно понимать, что алфавит языка **Math** намного шире, чем стандартный алфавит любого европейского языка (латинский, греческий или кириллица), поскольку помимо обычных букв он включает в себя цифры и огромное количество специальных знаков, чем-то напоминающих иероглифы. Однако у этих символов, как правило, вполне логичное происхождение и точная контекстная семантика.

Например, символ отношения принадлежности ($\in$), предложенный Джузеппе Пеано, происходит от первой буквы греческого слова *ἐστίν* («быть», т.е. быть частью, быть в заданном объеме понятия), аналогичное происхождение у обозначения дифференциала (d), предложенного Готфридом фон Лейбницем, и квадратного корня (от первой буквы слова *radix*), введенного Кристофом Рудольфом. Чуть более сложная история происхождения символа сложения ($+$): считается, что он произошел как сокращение латинского союза *et* («и»). Также можно вспомнить, что символы кванторов всеобщности ($\forall$) и существования ($\exists$) произведены от первых букв слов *Any* и *Exists* путем их отражения (авторство приписывается Г. Генцену и Ч. Пирсу).

Все символы алфавита мы относим к так называемым **графемам**, т.е. неделимым элементарным графическим символам. Предполагается, что эти символы человек уме-

ет легко различать с первого взгляда (исключением, пожалуй, является готический шрифт).

В дополнение к алфавиту мы также располагаем большим количеством *несамостоятельных графем*, которые служат для построения всевозможных выражений и сами по себе ничего не обозначают. К ним, например, относятся различные скобки, акценты, стрелки, операторные символы и т.д. и т.п. Характерным отличительным признаком несамостоятельных графем является наличие у них *подстановочных мест*, на практике часто обозначаемых переменными. Эти подстановочные места-переменные являются своего рода точками сборки более сложных лексических единиц нашего языка **Math**.

Кроме того, в математике и особенно в программировании часто встречаются многобуквенные обозначения каких-то функций, операций, констант, например, $\sin$ или $\arctan$. Эти обозначения могут быть как стандартизованными, так и определяемыми «на лету» в рамках одной статьи или книги. Такие обозначения мы не относим к алфавиту, поскольку они составлены из алфавитных символов, но называем **жесткими графемами**, поскольку они выступают в роли неделимых символов. Восприятие жестких графем как единых символов определяется контекстом и может быть затруднено в ряде случаев. Жесткие графемы тоже могут быть как самостоятельными, так и несамостоятельными (т.е. сопровождаться подстановочными местами).

Более подробное определение алфавита и графем мы приводим во второй части книги (см. разделы D1.1 и D1.2).

Все графемы, несмотря на их огромное разнообразие, образуют некоторый конечный, слабо меняющийся со временем список значков. Они являются элементарными частицами текста на языке **Math**. Точно так же, как буквы обычного языка являются элементарными частицами, из которых состоят письменные слова и предложения. И точно так же, как в разговорном языке, из алфавитных символов и графем языка **Math** складываются более сложные обозначения понятий, отношений, объектов.

Помимо смысловой нагрузки, уходящей корнями в различные разговорные языки и понятия, математические обозначения почти всегда несут функциональную нагрузку и подчиняются «**принципу матрешки**», который заключается в том, что все сложные слова математического языка **Math** есть комбинация его графем по заранее заданным правилам.

Точнее, скажем, что **лексемой** языка **Math** называется, во-первых, всякая самостоятельная графема (и в частности символы алфавита), а во-вторых, результат подстановки лексем в несамостоятельную графему на ее подстановочные места (или вместо переменных).¹ Мы специально здесь не выделяем переменные как особый синтаксический вид лексем, чтобы сделать изложение проще. Но в дальнейшем мы уделим переменным пристальное внимание.

Заметим, что в разговорном языке мы имеем похожую (хотя и не точно такую же) ситуацию: например, в английской грамматике имеет место строго определенный порядок членов предложения (определенные подстановочные места для подлежащего, сказуемого и т.д.), а в немецком и русском языках глагол часто определяет не только падеж зависимого слова, обозначающего объект действия (переходный глагол), но и предлог перед ним. Так что собирать предложения из слов мы можем только по строго определенным правилам.

¹ Более формальное и технически точное определение лексемы, основанное на понятии *производного шаблона*, будет дано во второй части книги (см. раздел D1.2). Однако на интуитивном уровне именно принцип рекурсивной сборки является ключевым.

Таким образом, принцип матрешки говорит нам буквально следующее: если у нас есть одна пустая матрешка и одна уже собранная матрешка, то мы можем вторую положить внутрь первой, и в результате получится более сложная собранная матрешка. Такая простая аналогия немедленно приводит к вопросу — что делать, если матрешки окажутся несовместимыми по размеру? Для этого существует *типизация* лексем и подстановочных мест. Проще говоря, всякое подстановочное место «признает» только лексемы своего типа, и при сборке лексем это нужно учитывать.

Например, в геометрии нельзя складывать две точки или находить их скалярное произведение, поскольку эти операции применимы только к векторам, в линейной алгебре нельзя брать детерминант у неквадратной матрицы, а в теории вероятностей невозможно сложить вероятность с событием. Но в математике несовместимость типов — это скорее исключение, чем правило, т.к. мы все стараемся сводить к числовым типам или использовать естественную конвертацию типов «на лету». В программировании же, наоборот, типизация лексем играет огромную роль и ее непонимание часто приводит к ошибкам в коде программ.

Посмотрим на простейшем примере, как происходит процесс лексемостроения. Символ сложения (+) является несамостоятельной графемой, имеющей два подстановочных места — слева и справа, причем как подстановочные места, так и сам результат подстановки принадлежат к одному типу — числам. На эти места мы можем подставить, например, алфавитные символы a и b , получив лексему $a + b$. Аналогично действуем в случае умножения: $c \cdot d$. А теперь мы можем вновь взять символ сложения, и на его подстановочные места подставить уже полученные нами лексемы: $a + b + c \cdot d$. Обычно подстановочные места принято заключать в скобки, чтобы сохранять структуру лексемы, так что правильнее было бы записать ее так: $(a + b) + (c \cdot d)$.

Здесь мы действуем строго по определению, подставляя «старые» лексемы в несамостоятельную графему, т.е. надстраиваем имеющиеся конструкции одним шагом «вверх».

Более привычным способом является обратная процедура — наращивание лексемы вниз путем подстановки готовых лексем вместо переменных другой готовой лексемы. Например, имея лексему $a + (b \cdot c)$, мы можем осуществить подстановку лексемы $a + b$ вместо переменной c , в результате чего получим лексему $a + (b \cdot (a + b))$. Такая генерация лексемы произведена не по определению, однако можно доказать, что и она приводит к правильно построенным лексемам, т.е. лексему $a + (b \cdot (a + b))$, на самом деле, можно собрать «снизу вверх», строго следуя определению. Подобного рода структурные теоремы обычно доказываются индукцией: предполагая, что лексема получается корректной независимо от способа сборки для лексем длины меньше n , мы далее доказываем, что это будет верно и для лексемы длины n .

В итоге получается, что лексемы можно собирать почти как угодно: достраивать их «вверх» путем подстановки в несамостоятельные графемы, выращивать их «вниз», заменяя переменные более сложными лексемами, или же комбинировать оба подхода сразу.

Кроме того, процесс сборки лексем никак не ограничен по сложности, в результате чего можно собирать бесконечно много сложных математических выражений. Мы вправе, например, строить многоэтажные башни степеней:

$$5^{4^{3^{2^1}}}$$

или изобразить что-нибудь этакое:²

$$x = \sqrt[3]{-\frac{q}{2} + \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{2}\right)^3}} + \sqrt[3]{-\frac{q}{2} - \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{2}\right)^3}}. \quad (\text{A1.1})$$

Следует, однако, понимать, что математики — тоже люди, и им не хочется громоздить необозримые многоэтажные выражения с огромным количеством символов, поэтому обычно вводятся дополнительные обозначения, позволяющие локально (в рамках одной статьи или книги, например) присвоить часть сложного выражения некоторому более простому выражению (например, одной букве) и в дальнейшем пользоваться им как сокращением. Аналогичного принципа мы придерживаемся и в языках программирования, когда вводим обозначения для функций и переменных.

Вышеупомянутую формулу (A1.1) можно декомпозировать на три более простых выражения, например, так:

$$x = \sqrt[3]{A + \sqrt{B}} + \sqrt[3]{A - \sqrt{B}},$$

где мы ввели временные обозначения:

$$A \stackrel{\text{def}}{=} -\frac{q}{2}, \quad B \stackrel{\text{def}}{=} \left(\frac{q}{2}\right)^2 + \left(\frac{p}{2}\right)^3,$$

символ $\stackrel{\text{def}}{=}$ означает равенство по определению или *присваивание*. Он часто используется для введения обозначений и сокращений в математических текстах. Этот символ ни в коем случае нельзя путать с равенством, поскольку он отражает не смысловое, а графическое равенство, т.е. переобозначение.

Такая инвариантность результата относительно порядка сборки (и декомпозиции) лексемы иллюстрируется с помощью следующей полезной конструкции, которая называется **деревом разбора**. Чтобы было проще, покажем дерево разбора на примере формулы Кардано (см. рис. A1.1). Кроме того, проиллюстрируем использование временных обозначений A и B в той же самой формуле соответствующими деревьями разбора (см. рис. A1.2)

Дерево разбора однозначно восстанавливается по формуле за исключением тех случаев, когда не указан приоритет выполнения операций. Так, формула $a + b + c$ может быть разбрана двумя способами в зависимости от расстановки приоритетов двух операций сложения: $(a + b) + c$ или $a + (b + c)$. Поэтому обычно дерево разбора применяется только к таким выражениям, где проставлены все формально требуемые скобки и тем самым приоритет выполнения операций определен однозначно.

Дерево разбора по сути представляет собой запись алгоритма построения формулы, в котором указано, как из самостоятельных и несамостоятельных графов собирается конкретная формула. Порядок ветвей в этом дереве важен, поскольку получаемый с его помощью текст имеет направленность. Только в том случае, когда мы наделяем текст математической семантикой (например, под знаком $+$ понимаем сложение), можно, следуя свойствам этой семантики, пренебречь порядком ветвей, но в таком случае дерево разбора становится не *синтаксическим*, а *семантическим*.

Из дерева разбора хорошо видно, что построение формулы можно начинать с любого места, тем более, что в нем есть одинаковые ветви. Мы можем построить сначала

² Формула Кардано для решения приведенного кубического уравнения $x^3 + px + q = 0$.

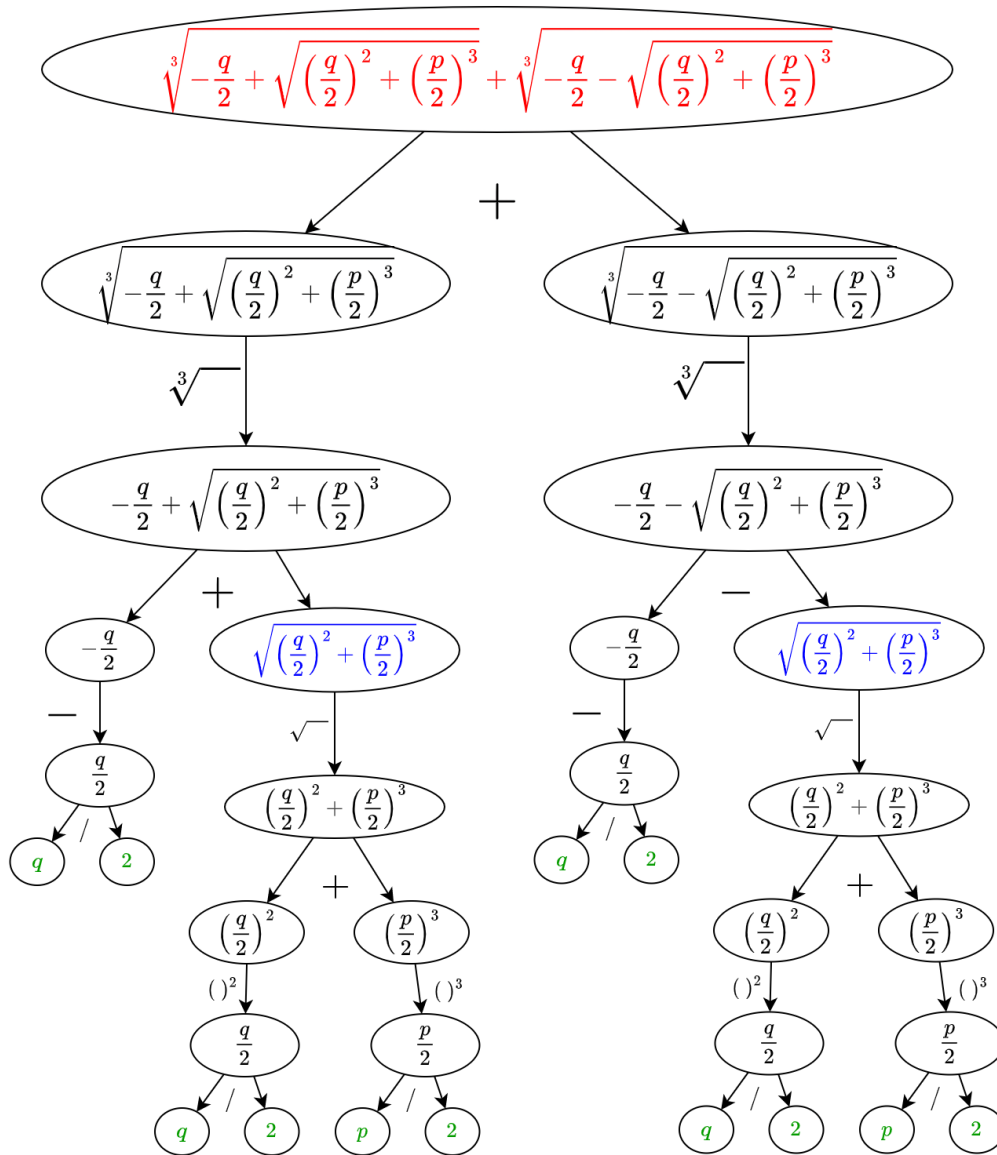


Рис. А1.1. Дерево разбора для формулы Кардано.

подкоренные выражения, потом отложить их в сторону и построить формулу с корнями, под которыми будут пустые подстановочные места, а затем в эти подстановочные места вставить заранее заготовленные подкоренные выражения. Главное здесь — не путать порядок используемых подстановочных мест. Если сложение нам еще может простить такую путаницу, то вычитание и деление не простят, формула станет неверной.

Отметим также, что дерево разбора всегда заканчивается какими-то простейшими графами, например, символами алфавита или временными обозначениями (которые, как правило, тоже являются алфавитными). Дерево можно модифицировать тремя способами: заменять ветвь на какой-то алфавитный символ (редуцирование), присоединять новую ветвь вместо листа (конечной вершины) дерева (наращивание вниз) и, наоборот, само дерево подставлять вместо листа другого дерева (наращивание вверх). При этих манипуляциях мы получаем новые лексемы, соответствующие строгому определению лексем.

Понятие дерева разбора, которое мы используем для анализа структуры формул, не

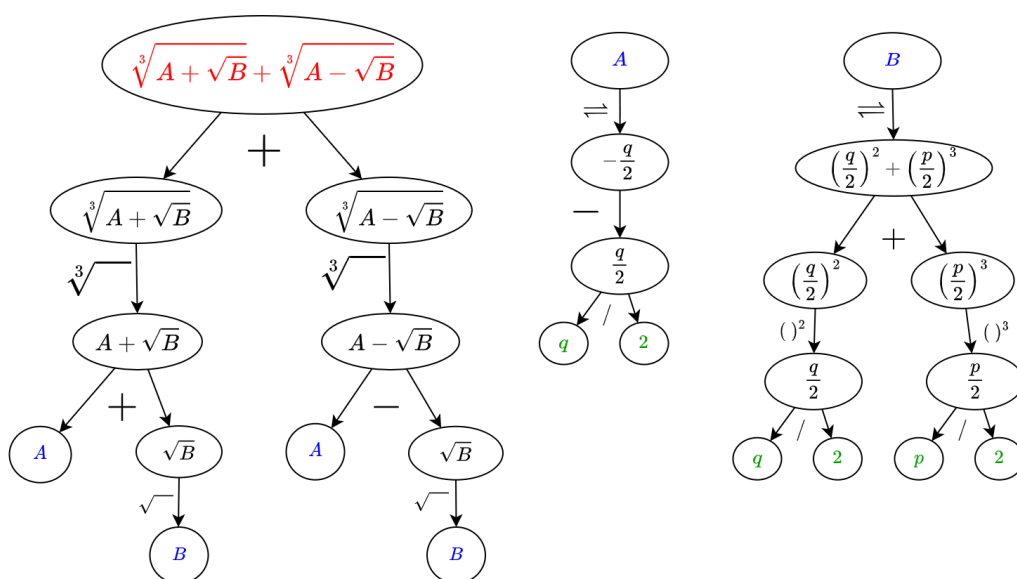


Рис. A1.2. Использование временных обозначений A и B

является лишь удобной визуализацией. Это — фундаментальный инструмент, на котором построены языки программирования и системы искусственного интеллекта. Когда компилятор переводит программу, написанную человеком, в машинный код, он первым делом строит именно такое синтаксическое дерево, чтобы понять структуру команд.

Современные системы ИИ, такие как большие языковые модели, не пытаются восстановить грамматическую структуру текста путем явного построения синтаксических деревьев. Вместо этого они улавливают колоссальный объем статистических закономерностей. Поразительно то, что выученные ими неявные структуры настолько точно соответствуют правилам языка, что классические деревья разбора превратились из средства анализа в мощный инструмент для верификации, позволяющий нам убедиться, что машина действительно «поняла» синтаксис.

Глядя на столь красивую законченную идею построения лексем, можно предположить, что любые конструкции, возведенные по принципу матрешки, являются математическими выражениями. На самом деле это не так, и в каждом конкретном формальном фрагменте языка **Math** задаются определенные правила, которые можно применять для сборки выражений путем наращивания дерева разбора. И дело тут не только в типизации лексем, но также и в ограниченном наборе используемых символов. Иначе говоря, конкретный формализм оперирует конкретными графемами. Но других ограничений при этом нет.

Отметим, что сама идея представления сложных выражений через структурированную сборку из более простых элементов имеет глубокие корни. Еще в XVII веке Готфрид Лейбниц мечтал о создании 'characteristica universalis' — универсального символического языка, который позволил бы формализовать человеческие рассуждения. Хотя его проект не был полностью реализован, стремление к точности и структурности в символических записях проложило путь к современной математической нотации, где «принцип матрешки» и анализ структуры выражений (подобный построению дерева разбора) являются неотъемлемой частью.

A1.3. Семантика vs Синтаксис

Может показаться, что синтаксис языка при построении математической теории первичен, но это не совсем так. На самом деле, прежде чем синтаксис будет полностью определен, необходимо разобраться с тем, насколько богатым должен быть язык, который мы хотим использовать, какие классы объектов он должен описывать, какие взаимосвязи, имеющие место в изучаемой предметной области, он должен выделить как существенные и зафиксировать в своей системе обозначений.

Поэтому первичной для языка, безусловно, является семантика. Но, однажды родившись, формальный язык и построенные на его основе математические теории могут «забыть» исходную семантику и применяться в самых разных областях знаний там, где это «технически» возможно. Следуя Давиду Гильберту, можно заметить, что формальная система работает одинаково хорошо независимо от того, что мы понимаем под обозначенными в ней предметами — будь то точки, прямые и плоскости, либо же столы, стулья и пивные кружки.³



Давид
Гильберт

При этом, однако, не стоит забывать о том, что если точки мы называли столами, то такую подмену понятий мы производим тотально, т.е. абсолютно все формулы нашего языка отныне будут говорить про столы там, где раньше они говорили про точки, и в то же самое время нельзя подразумевать столы в тех местах, где ранее мы использовали прямые или плоскости, т.е. смешивать исходно различные понятия. Иначе говоря, если уж производить замену понятий (*выбирать интерпретацию*), то это нужно делать согласованно и тотально, поскольку если сама по себе семантическая природа описываемых сущностей не играет роли для формализма, то все отношения между сущностями должны быть адекватно перенесены из одной области в другую. Грубо говоря, если в геометрии у нас имеется формула, выражающая отношение «лежать между» для трех точек, то и в предметной области столов и стульев у нас также будет работать формула, означающая свойство «лежать между», только уже для трех столов.

В контексте современных достижений искусственного интеллекта, эта способность формальных систем «забывать» исходную семантику приобретает новое измерение. ИИ-системы, обучаясь на огромных корпусах данных, *компактифицируют* знания, выявляя наиболее вероятные связи и закономерности. Они не обязательно оперируют с пониманием истинности или ложности в человеческом смысле, но способны вычислить «правильность» или «допустимость» конструкций, основываясь на статистических корреляциях и паттернах, извлеченных из обучающих данных. Это позволяет им, например, генерировать тексты или формулы, которые формально безупречны и даже кажутся осмысленными, даже если внутренняя «семантическая реальность», на которой базируется их генерация, может быть совершенно иной, чем человеческая. Это подчеркивает, как математический язык, благодаря своей строгой структуре, позволяет отделить синтаксис от семантики, создавая универсальный инструмент, применимый даже в тех случаях, когда его пользователи не обладают полным интуитивным пониманием его глубинных смыслов.

Пожалуй, самым ярким примером разброса семантических реализаций математических теорий можно считать теорию групп, которая, как известно, широко используется не только внутри самой математики в рамках более частных теорий, но также в физике,

³ Неточная цитата из книги К. Рид «Гильберт» [13].

химии, кристаллографии, информатике и даже в некоторых гуманитарных направлениях.

Другой хороший пример различного применения одной и той же теории — это теория линейных операторов, которая применяется, например, в квантовой механике и в алгоритмах машинного анализа текстов, и много где еще.

Попробуем, не вдаваясь глубоко в искусство построения формализмов, пощупать тот механизм, который приводит нас к математическим формулам, на примере логических выражений обычного (разговорного) языка. Прежде всего нам потребуется отделить друг от друга два понятия: объекты (субъекты) и отношения. Как и в разговорном письменном языке, мы будем различать лексемы, соответствующие этим понятиям. **Предикатами**⁴ называются лексемы, обозначающие отношение между объектами и/или субъектами, они обычно имеют логический тип семантики. **Термами** называются лексемы, обозначающие объекты исследуемой предметной области.

Так, приведенный выше пример формулы Кардано (A1.1) представляет собой предикат равенства, выражающий тождество корня x и некоторого арифметического выражения. А само это выражение, для которого мы привели дерево разбора на рис. A1.1, представляет собой терм, выражающий описание некоторого объекта (числа). Следуя этому описанию, отправляясь от конкретных значений p и q , можно вычислить конкретное значение x .

Другой пример предиката: отношение $<$ на целых числах.

Наряду с предикатами используется также более общее понятие — **формула**. Формула, как и предикат, является записью отношения между объектами с помощью предикатов и логических символов. Иначе говоря, формула — это логически выстроенное описание отношения.

Подчеркнем, что предикат иногда может быть определен довольно сложной формулой, т.е. не все предикаты в рамках данного языка имеют элементарный вид с точки зрения определения.

Забегая вперед, воспользуемся некоторыми логическими и математическими спецсимволами: $\forall$ — «для любого», $K \cap M$ — пересечение двух множеств, $k \in K$ — k есть элемент множества K .

Предположим, что нашей предметной областью являются животные, т.е. мы выступаем в роли зоологов. Пусть K — множество всех крокодилов, M — множество всех животных в Москве-реке. Любое животное имеет вполне конкретную характеристику — цвет (чтобы не усложнять модель, будем считать, что имеется в виду некоторый один главный цвет). Пусть предикат $\mathcal{A}(k, c)$ принимает истинное значение тогда и только тогда, когда животное k имеет цвет c . Пользуясь такими обозначениями запишем следующее утверждение: «Все крокодилы в Москве-реке красные». Получим:

$$\forall k \in K \cap M : \mathcal{A}(k, \text{«красный»}). \quad (\text{A1.2})$$

Разберем, как это получилось. Во-первых, мы решили ограничить предметную область только животными. Поэтому любая переменная (в нашем случае k) автоматически обозначает у нас произвольное животное. Во-вторых, мы выделили в генеральной совокупности животных два подмножества: «живущие в Москве-реке» и «крокодилы», при этом мы никак не определили эти классы явным образом, так что на самом деле неважно, какие слова для их обозначения мы используем. В-третьих, мы ввели еще один сорт объектов, который называли цветами (опять же, совершенно неважно, что на

⁴ От лат. *praedicatum* — заявленное, упомянутое, сказанное.

самом деле скрывается за этим словом). Наконец, мы добавили в теорию обозначение $\mathcal{A}$ для формулы, которая умеет связывать два вида объектов: животных и цвета. Семантика этого обозначения такова, что мы понимаем под ним соответствие животного и его цвета (в логике это называется *предикатом*, а в разговорном языке роль предиката обычно выполняет глагол). После этого мы написали формулу (A1.2), которая буквально расшифровывается так:

для любого k , который одновременно принадлежит множеству K и множеству M , выполняется предикат $\mathcal{A}(k, \text{«красный»})$.

Одновременная принадлежность двум множествам у нас реализована формулой $k \in K \cap M$, которая означает, что k входит в пересечение множеств K и M , т.е. является элементом и того, и другого множества одновременно. Иначе эту же формулу можно было бы записать так: $(k \in K) \wedge (k \in M)$, т.е. вместо операции пересечения множеств воспользоваться логической операцией *конъюнкции* ($\wedge$).

Попутно пронаблюдаем, как в формуле $k \in K \cap M$ реализуется «принцип матрешки». Сначала мы взяли алфавитный символ пересечения $\cap$, который имеет два подстановочных места (слева и справа), куда разрешено подставлять только множества, и подставили алфавитные символы K и M , получили более сложную лексему $K \cap M$. Затем мы взяли символ принадлежности $\in$, который также имеет два подстановочных места, и подставили, соответственно, переменную k и только что собранную лексему $K \cap M$. Таким образом, лексема $k \in K \cap M$ представляет собой двухуровневую матрешку из двух символов, имеющих подстановочные места (операторные символы), и трех символов без таковых мест (переменные).

Наконец, вспомним про «столы, стулья и пивные кружки». Пусть теперь мы имеем дело не с животными, а со звездами всей Вселенной. Пусть также K есть совокупность всех нейтронных звезд, M — совокупность всех звезд нашей галактики «Млечный путь», C — множество названий цветов, а выражение $\mathcal{A}(k, c)$ говорит нам, что звезда k имеет спектральный класс с преобладающим цветом c . В таком случае та же самая формула (A1.2) на сей раз будет иметь следующую семантику:

всякая нейтронная звезда галактики «Млечный путь» имеет красный спектр

Как видим, мы полностью изменили семантику нашей небольшой теории, но формула (A1.2) при этом не претерпела никаких изменений: формализм остался прежним.

Стоит отметить, что формальная теория становится теорией только тогда, когда помимо языка мы также предъявляем механизм определения того, какие формулы, написанные на этом языке, истинные, а какие ложные. Для этого теория снабжается *аксиомами*, из которых по общим логическим правилам выводятся истинные (относительно данных аксиом) утверждения, называемые *теоремами*.

Добавление аксиом может существенно сузить спектр применения формальной теории, поскольку теория будет иметь ценность только в том случае, когда при наделении ее определенной семантикой из заданной предметной области аксиомы теории будут истинными в данной семантике. Иначе говоря, если мы рассматриваем геометрию Эвклида про точки, прямые и плоскости со всеми пятью постулатами, а потом заменяем их на столы, стулья и пивные кружки, то мы ожидаем, что переписанные в этом новом языке постулаты Эвклида окажутся истинными. В этом случае и все теоремы геометрии будут истинными для столов, стульев и кружек, и теория будет полезной в данной предметной области. Любое же мельчайшее несоответствие аксиом или теорем свойствам

реальных объектов, к которым мы пытаемся применить формальную теорию, означает невозможность ее использования в данном случае.

Подводя промежуточный итог, отметим, что язык математики достаточно гибок для того, чтобы описывать на нем самые разные предметные области. Кроме того, он достаточно абстрактен для того, чтобы имелась возможность один и тот же язык использовать в совершенно различных ситуациях, т.е. математический язык, будучи порожденным какой-то конкретной семантикой, почти мгновенно отрывается от нее, следуя только своей внутренней архитектуре и правилам, никак не связанным с предметной областью.

Эта универсальность языка дает гарантию надежности всех результатов, полученных в нем: если язык удалось адекватно сопоставить с какой-то областью, и в этой области верны некоторые аксиомы, то в ней верны также и все теоремы, доказанные чисто логически из данных аксиом. Такое свойство математических теорий позволяет рассматривать их сами по себе, в отрыве от конкретных реализаций, и получать универсальные абсолютно точные теоремы.

Нужно отметить также, что и внутри самой математики существуют различные уровни абстракции и соответствующие им формальные языки и теории. Так, язык логики первого порядка (исчисление предикатов) является наиболее абстрактным и работает практически во всей математике без ограничений. Он и является тем самым рафинированным остовом аристотелевской логики, который мы называем формальной логикой.

Язык теории множеств является «овеществлением» логики в терминах множеств и элементов, он позволяет строить абсолютно все математические конструкции в виде конкретных множеств (или их классов). Однако сама по себе теория множеств страдает неполнотой в отличие от логики предикатов, что приводит к различным интересным «парадоксам». Следующей теорией по силе абстракции, видимо, следует считать теорию групп и колец, которая, как мы уже отмечали, имеет массу конкретных реализаций в самых разных областях исследований как в самой математике, так и за ее пределами.

С другой стороны, применение теорий в конкретных ситуациях следует производить с осторожностью: не путать, не менять и не дополнять введенные понятия, проверять истинность используемых аксиом в данной предметной области, не вводить в язык новые понятия, имеющие какой-либо внешний контекст, без строгого их определения в рамках самой формальной теории.

Наконец, нужно отметить, что в силе математических теорий заложена и некоторая их слабость, а именно: все результаты теории являются обезличенными, они не могут «почувствовать» тонкости, отличающие одну реализацию теории от другой. Неформально говоря, математическая теория «не видит» разницы между теми своими реализациями, которые являются корректными.

A1.4. Рефлексия

Очень важным свойством языка **Math**, как, в общем-то, любого достаточно развитого разговорного языка, является способность к *рефлексии*, т.е. возможность на этом языке описывать и анализировать внутренние закономерности самого себя.

Для этого обычно в языке выделяется специальный корпус графем, заточенный под описание элементов и конструкций самого себя. Эта специальная часть языка становится самостоятельным языком со своими особенностями, но в целом следует правилам

более общего, изучаемого ею языка. Обычно язык, используемый для описания других языков, называют **метаязыком**. В языке, способном к рефлексии, метаязык является его собственной частью.

Отметим, что инкапсуляция метаязыка в исходный язык **Math** позволяет, на самом деле, и метаязыку описывать самого себя, как часть языка **Math**. И здесь мы можем привести иллюстрацию из разговорного языка. В любом развитом языке, в частности, в русском, есть прилагательные, которые качественно характеризуют какие-либо предметы и явления. В частности, мы можем рассмотреть прилагательные, которые описывают свойства прилагательных. Например, любое прилагательное можно охарактеризовать как сложное или простое в зависимости от того, как оно устроено (скажем, «красный» — простое прилагательное, а «сложносоставной» — сложное, т.к. включает два образующих корня). Таким образом, среди всех прилагательных разговорного языка можно выделить класс прилагательных, которые могут быть использованы для описания всех прилагательных. Это и будет простейший метаязык внутри разговорного (русского) языка.

На рефлексии основаны различные «неудобные» логические парадоксы. Во-первых, это всем известный парадокс лжеца (или брадобрея), который мы не будем здесь повторять. Во-вторых, с прилагательными тоже можно придумать парадокс. Коль скоро прилагательные умеют что-то говорить про самих себя, разделим их на два класса: *самоприменимые* и *несамоприменимые*. Например, прилагательное «сложносоставной» является самоприменимым, т.к. оно действительно является сложносоставным (происходит от двух основ), а прилагательное «несложносоставной» явно не является самоприменимым.

Теперь зададим вопрос: а к какому классу относится прилагательное «несамоприменимое»? С одной стороны, если оно самприменимо, то оно является несоприменимым (ибо оно нам об этом говорит), с другой стороны, если оно несоприменимое, то оно именно таким свойством и обладает, т.е. является самоприменимым. Получается неразрешимый логический парадокс.

Чтобы не впадать в подобные ловушки, обычно требуют ограничить *самореферирующие* определения, т.е. определения, ссылающиеся на самих себя, а то и вовсе их запретить. Так что метаязык обычно обособляется от остального языка и сам про себя ничего не говорит.

Аналогичным трюком воспользовался Гёдель для получения своей самой знаменитой теоремы о неполноте формальной арифметики. Дело в том, что арифметика способна сама себя анализировать, если определенным способом научиться нумеровать арифметические формулы (т.е. лексемы языка арифметики), и в ней тоже можно найти утверждение, которое будет заявлять нечто о самом себе. Но в арифметике этот парадокс разрешается тем, что мы отделяем понятие истинности от доказуемости, выводя тем самым «нехорошие» высказывания в серую зону понятия доказуемости (они становятся ни доказуемыми, ни опровержимыми). С одной стороны, это спасает математику от противоречий, с другой стороны, дает повод усомниться в ее всеисильности.

Что касается языка **Math**, которым мы здесь занимаемся, то, как уже было сказано, мы просто не будем применять метаязык к анализу самого себя, считая его понятия и лексемы некой надстройкой более высокого порядка. С одной стороны, мы избегаем логических парадоксов, с другой стороны — получаем в руки более-менее стандартные языковые средства, позволяющие рассуждать в целом о языке **Math** некоторым формализованным образом, по возможности избегая средств разговорного языка (на котором до сих пор излагается эта книга).

В целом, нужно отметить, что такой дуализм языка в математике скорее правило, чем исключение. Математик всегда работает одновременно на двух уровнях языка: конкретном и абстрактном. Конкретный язык описывает исследуемые предметы и явления формально (например, линейные операторы и их свойства), абстрактный язык обслуживает процесс оперирования конкретным языком и наполнением его смыслом (например, для введения определений и пояснения целей формальных выкладок).

Наиболее отчетливо такой дуализм наблюдается в математической логике, где мы исследуем какую-то формальную систему (например, арифметику Пеано) с помощью более абстрактных инструментов логики и теории множеств. Более того, над этой метатеорией множеств мы также имеем в виду и мета-метатеорию, которая описывает общую канву исследований данной формальной системы.

На заметку: у любого художественного произведения тоже есть несколько смысловых уровней: то, что автор описал буквально; то, что он имел в виду (скрытый смысл, управляемый автором); а также непроизвольное проявление личности самого автора в данном произведении.

A1.5. Термины

Связь языка **Math** с разговорным (в нашем случае — русским) языком остается весьма прочной прежде всего через такое понятие как **тэрмин** (не путать с *термом*!). Дело в том, что лексемы сами по себе, как синтаксические единицы, не наделены смыслом, но для того, чтобы придать им определенную семантику (как тот самый робот из рассказа Азимова), мы должны так или иначе закрепить за лексемами какие-то понятия. Например, лексема $\sin$ служит для обозначения функции синуса. В этом есть определенный смысл выбранного обозначения: во-первых, график синуса похож на вогнутый берег, который называется заливом, во-вторых, по латыни залив — *sinus*, откуда и взялось обозначение. Таким образом придумываются в науке большинство (если не все) словесных обозначений понятий. Но ведь слово разговорного языка, записанное на бумаге, («sinus» или «синус») — это лексема данного слова в данном разговорном языке.

Отсюда возникает определение: **термином** мы называем лексему разговорного языка, закрепленную за данным понятием и имеющую соответствующую лексему в языке **Math**.

Итак, есть понятие функции синуса, за которым закреплен термин «синус» и лексема $\sin$ (языка **Math**).

Термины очень удобны, когда мы хотим определить какое-то понятие (локально в данной статье/тексте или глобально для всей науки), для которого нет устоявшейся лексемы, а само определение достаточно длинное, чтобы его каждый раз выписывать. Например, термин «функция» выделяет строго определенное понятие в математике, имеющее нетривиальное определение. Более того, сам по себе разговорный термин несет в себе определенную коннотацию, которая (так же, как в случае с заливом и синусом) перебрасывает мостик между смысловым наполнением разговорного термина и строго определенным формальным математическим понятием. Это практично, экономично и органично.

Таким образом, наличие терминов, взятых из разговорного языка и сопровождающих понятия и лексемы языка **Math**, обеспечивает нам некоторую базовую семантику, позволяющую а) лучше запоминать лексику языка **Math**, б) легче читать тексты на

языке **Math** и в) быстрее усваивать новые понятия и связи между ними, что является немаловажным гуманитарным фактором при работе с языком **Math**.

Ведь, как говорили древние греки, мерой всех вещей является человек!

При этом не стоит забывать, что семантика разговорного языка, проявляющаяся в терминах, является вспомогательной, иллюстративной (как картинки и графики), она не может служить доказательной базой внутри самого языка **Math**.

A1.6. Переменные и свертки

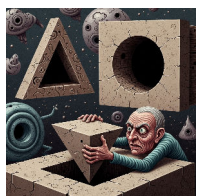
Проводя аналогию с разговорным языком, под **переменными** мы можем подразумевать достаточно простые лексемы, которым отведена роль обозначать произвольный элемент некоторого класса. Например, в разговорном языке под словом «человек» понимается какой-то человек (не какой-то конкретный человек и не все люди в целом, а просто произвольный представитель множества людей). Так и в языке **Math**, когда мы хотим что-либо сообщить о произвольном объекте данного рода, мы его обозначаем некоторой переменной, например, x или y , и далее пользуемся этой переменной в формулах. По сути переменная — это тоже временное обозначение, но не для конкретной формулы, а для произвольного объекта из того класса объектов, которые мы описываем в нашем языке.

Нужно отметить, что несмотря на достаточно простое определение переменной, само по себе это понятие — достаточно сложное с точки зрения семантики. Если вспомнить языки программирования (которые можно считать реализациями языка **Math**), то переменные бывают нескольких типов, например, строковые или числовые. Это значит, что за переменной закрепляется некоторый *тип*, который ограничивает возможности использования этой переменной. С одной стороны, это заставляет нас производить согласование типов, когда мы составляем какие-либо сложные лексемы, чтобы, например, случайно не подставить буквы вместо числовой переменной, с другой стороны, это сильно упрощает синтаксис, поскольку не нужно на протяжении всего текста (например, статьи по математике или листинга программы) явно указывать, какие значения может принимать та или иная переменная. Достаточно при определении переменной сказать что-то вроде «пусть x — произвольное натуральное число».

Вторая проблема, связанная с понятием переменной, — это, собственно, определение данного понятия. Чисто синтаксически понятно, что если есть переменная некоторого типа, то вместо нее можно подставлять любую лексему того же типа. Но это как минимум означает, что мы должны распределить все лексемы по типам, т.е. навязать им некоторую классификацию несинтаксического свойства. В формальных математических теориях так и поступают. При определении того фрагмента языка **Math**, который будет использоваться в конкретной теории, мы оговариваем сразу, как должны выглядеть лексемы того или иного типа, либо же постулируем отсутствие типов. В последнем случае все намного проще, поскольку переменные могут пробегать все возможные объекты, описываемые этой теорией, без каких-либо ограничений.

Неформально можно представлять себе переменную как некоторую ячейку определенной формы, в которую можно положить только совместимые по форме предметы (в треугольную ячейку — треугольные призмы, в круглую — цилиндры, и т.д.). Подходящие по форме предметы — это реализации данной переменной внутри своего типа. В разговорном языке аналогами переменных являются имена нарицательные. Например,

слово «дом» в каждом конкретном контексте может обозначать какой-то конкретный дом или разновидность домов, но не может применяться к человеку или животному.



Tun

В математике выделение классов объектов из общего разнообразия достигается написанием определяющей *формулы* и введением *термина*. При этом, как уже говорилось, сам по себе термин не относится к языку **Math**, он просто подменяет собой определение, выраженное на языке **Math**, каким-то говорящим словом. Например, «функция», «целое число», «квадратичная форма», «замкнутое множество» — это термины, за которыми не закреплены обозначения переменных. С другой стороны, есть термин «ординал», за которым иногда закрепляются переменные из

начала греческого алфавита (если это текст из оснований математики), чтобы всегда под α , β , γ понимать произвольные ординалы. При этом мы часто определяем обозначение для класса, которое можно считать обозначением константы, т.е. именем собственным. Например, класс всех ординалов обозначается Ord , класс всех целых чисел — $\mathbb{Z}$ и т.д. В разговорном языке классы однородных предметов обычно именуются словами во множественном числе, например, «дома», «люди» и т.д. Это означает, что «дом» есть элемент класса «дома». Сравните: α есть элемент класса Ord (ординал есть элемент класса ординалов).

Однако большинство имен нарицательных в разговорном языке — это обозначение довольно узких классов вещей. Тот же «дом» — это очень специальный класс предметов, имеющий сравнительно небольшой объем по сравнению с объемом вообще всех вещей, описываемых языком.

В то же время, в математике (и программировании) переменная обычно пробегает достаточно общий класс (по модулю рассматриваемой реализации языка **Math**). А значит, сравнение переменных только с именами нарицательными из обычного языка не вполне корректно. И тут нам на помощь приходят местоимения. Ведь такие слова, как «он», «она», «оно», «этот», «эта» и т.д., применимы в разговорном языке к очень широким классам объектов, т.к. разделяют их только по родовому признаку. Поэтому переменные можно сравнивать как с именами нарицательными, так и с местоимениями (но помнить, что любая аналогия неполна).

Как видим, и само понятие переменной, и ее соотнесение с понятиями из разговорного языка вызывает определенные затруднения и не является однозначным. Это значит, что параллели между математическим и разговорным языками нужно проводить с осторожностью.

Переменные позволяют нам выделить один особенный класс лексем языка **Math** под общим названием **свертки**. Под сверткой мы будем понимать такие лексемы, в которых переменные (одна или больше) играют роль итератора, по которому производится некоторое действие или берется предикат, в результате чего эта лексема обозначает объект, не зависящий от переменных, по которым производится итерирование. Эти итерируемые переменные называются *связанными* (данной сверткой), и их можно заменять на другие переменные того же типа, но нельзя заменять никакими другими лексемами. Свертка уменьшает число свободных параметров исходного выражения, которым она определяется.

Приведем примеры лексем-сверток (отделены друг от друга запятыми):

$$\int_0^1 e^{-x} dx, \quad \sum_{n=1}^{100} 2^n, \quad \{x \mid x^2 > 2\}, \quad \forall x (x = x), \quad \exists x (x > 0).$$

Здесь переменные x и n — те самые связанные переменные, по которым производится итерирование некоторой операции (в случае интеграла и суммы — суммирование, в случае множества и кванторов — проверка истинности соответствующего предиката). В конкретной математической семантике у каждой из этих лексем-сверток мы получаем конкретное значение, которое не зависит от x и n .

Часто лексема-свертка содержит переменные, которые она не связывает. Такие переменные называются *свободными переменными* или *параметрами*. Покажем это на примерах.

► **Свертка функций.** Свертка $(f * g)(t)$ обычно определяется интегралом:⁵

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{+\infty} f(\tau)g(t - \tau)d\tau.$$

Здесь τ — связанная переменная (итератор), которая «поглощается» интегралом. В отличие от нее, t — это свободная переменная (параметр) внутри связывающей лексемы. Результирующая лексема является записью функции от t , т.к. ее значение меняется при варьировании параметра t .

► **Свертывание множества с параметром.** Рассмотрим множество всех вещественных чисел, строго меньших некоторого числа c :

$$M_c \stackrel{\text{def}}{=} \{x \mid (x \in \mathbb{R}) \wedge (x < c)\}.$$

Переменная x связана фигурными скобками (нотацией построения множества) и служит лишь для описания элементов. Переменная c является параметром; изменение c меняет само множество (например, M_0 — это множество отрицательных чисел, тогда как M_{10} — более широкое множество).

► **Кванторы с параметрами.** Формула $\exists x (x^2 = a)$ содержит связанную переменную x и параметр a . Эта формула утверждает, что a имеет квадратный корень. Истинность этого утверждения полностью зависит от значения параметра a (в области вещественных чисел оно истинно для $a \geq 0$ и ложно для $a < 0$).

Таким образом, свободные переменные действуют как «входной интерфейс» лексемы, связывая ее с внешним контекстом, тогда как связанные переменные обеспечивают внутреннюю работу механизма свертывания.

Является ли эта концепция свертки чисто математическим изобретением? Вовсе нет. На самом деле, естественный язык активно использует очень похожий механизм, хотя он и менее строго формализован. Когда мы используем обобщающие конструкции, мы, по сути, создаем свертку. Хороший пример — знаменитая фраза, которая, по легенде, была начертана над входом в Академию Платона:

«Тот, кто не знает геометрии, да не войдет сюда.»

Синтаксически это явная структура связывания через квантор. Фраза «Тот, кто» по существу играет роль квантора всеобщности ($\forall$) в сочетании со связанной переменной. Она не относится к конкретному человеку; скорее, она создает временный заполнитель

⁵ Здесь слово «свертка» относится не только к синтаксическому объекту (лексема-свертка), но и непосредственно к математическому объекту — функции. Это — термин из функционального анализа.

(placeholder) для произвольного человека, назовем его x . Если этот произвольный человек x подходит под описание, то запрет распространяется на x .

В этой конкретной фразе слова «геометрия» и «сюда» выступают как **константы** (конкретные значения). Они определяют конкретный экземпляр правила. Однако мы легко можем представить общий логический *шаблон*, стоящий за этой фразой:

«*Тот, кто не знает Предмета, да не войдет в Место.*»

В этом шаблоне «Предмет» и «Место» становятся **параметрами** (свободными переменными). Присваивая этим параметрам различные значения, мы меняем область действия правила, не меняя его внутренней логики связывания. Например, если мы положим Предмет $\stackrel{\text{def}}{=} \text{«пароль»}$, а Место $\stackrel{\text{def}}{=} \text{«серверная»}$, мы получим стандартный протокол безопасности. Это полностью аналогично **свертке с параметрами**, такой как выражение $\exists x(x^2 = a)$, которое мы обсуждали ранее. Логическая структура (связывание переменной x) остается прежней, но конкретное утверждение зависит от значения, присвоенного свободному параметру a .

Математика просто берет эту естественную языковую способность к обобщению и накладывает строгие правила на *область действия* переменной (где она начинается и заканчивается) и ее *тип*, предотвращая двусмысленности, часто встречающиеся в обычной речи.

A1.7. Особенности математического языка

Настало время подвести некоторый промежуточный итог нашим изысканиям. В этом разделе мы отметим основные черты языка **Math**, которые в совокупности делают его, с одной стороны, уникальным с точки зрения наших знаний о языках вообще, а с другой стороны, подчеркивают и некоторую общность с ними:

1. Горизонтальная направленность текста.
2. Принцип матрешки и однозначность разбора лексем.
3. Синонимы и редукции.
4. Финитность конструкций.
5. Строгая детерминированность переменных.
6. Изъявительная форма повествования.
7. Отсутствие обычных спряжений и склонений.

Горизонтальную направленность текста вряд ли можно назвать отличительной чертой языка **Math**, поскольку она свойственна и разговорным языкам. Однако и в математике, и в языках программирования общее правило чтения лексем слева направо часто встраивается в некоторый структурный порядок текста, где, с одной стороны, внутри лексем допускаются самые разные направления (вспомните хотя бы обозначение определенного интеграла или матрицы) и порядок чтения зависит от семантики конкретного шаблона, а с другой стороны, общая логика текста подразумевает скорее вертикальную ориентацию сверху вниз, что в общем тоже согласуется с разговорными

языками, но лишь отчасти. Особенно это заметно в структуре листинга программ, когда переход на новую строку определяется не столько шириной носителя текста, сколько правилами обработки (чтения) данного текста. То же самое можно заметить и в отношении отступов в начале строки: часто они имеют значение не только для красоты и удобства чтения вложенных блоков программ, но и несут определенный функциональный смысл (как, например, в языке Python).

Принцип матрешки — пожалуй, главное отличительное свойство языка **Math**, поскольку он является определяющим для понятия лексемы нашего языка. Этот принцип, с одной стороны, позволяет потенциально бесконечно генерировать новые лексемы, с другой стороны, появляется возможность разобрать каждую лексему на составляющие части (построить дерево разбора) и понять, как она была сконструирована. Причем, как правило, такой разбор производится однозначно (даже если существуют альтернативные варианты разбора, то они в некотором смысле эквивалентны).

Данная особенность языка **Math** резко выделяет его на фоне разговорных языков. Конечно, некоторую рекурсивность или структурность можно усмотреть и в разговорных языках: все мы с детства помним, что предложение можно разложить на подлежащее, сказуемое и второстепенные члены, которые, в свою очередь, можно представить различными видами частей речи (существительное, глагол, прилагательное, наречие и т.д.), те уже представляются конкретными словами, а слова — морфемами и, в конце концов, буквами. Однако мы сразу же видим, во-первых, ограниченность такой иерархии, поскольку сложность структуры речи мы не можем наращивать неограниченно, а во-вторых, изначально заложенную типизацию элементов на каждом уровне «сборки» текста (например, в качестве сказуемого нужно всегда использовать глагол или, реже, составное (именное или глагольное) сказуемое). Это тот самый случай типизации, который мы обсуждали выше в связи с переменными, однако в разговорном языке типизация больше привязана к структуре речи, чем к семантике, в то время как в математике типизация прямо указывает на семантику объектов, описываемых лексемами того или иного типа.

Справедливости ради, стоит привести пример исключения, которое лишь подтверждает правило: в разговорном языке можно придумывать сколь угодно много уровней вложенности, если рассматривать цепочку придаточных предложений:

Вот два петуха,
Которые будят того пастуха,
Который бранится с коровницей строгою,
Которая доит корову безрогую,
Лягнувшую старого пса без хвоста,
Который за шиворот треплет кота,
Который пугает и ловит синицу,
Которая часто ворует пшеницу,
Которая в тёмном чулане хранится
В доме, который построил Джек.

И здесь мы можем видеть что-то вроде башни степеней из арифметики. Примечательно, что хотя обе конструкции (как в данном стихе, так и в случае башни степеней) никак не ограничивают нас в творчестве, чрезмерное увлечение ими приводит к полной нечитабельности текста.

Кроме того, большинство разговорных языков не являются аналитическими, т.е. те лексемы, которые используются при построении текста разговорного языка, могут

иметь т.н. начальную форму и различные вариации, образованные с помощью зависимых морфем (окончаний, суффиксов, приставок и т.п.), которые видоизменяются в зависимости от контекста. Язык **Math**, если не относить к нему термины, взятые из разговорных языков, наоборот, предельно аналитичен именно в силу природы его лексем. Действительно, трудно представить себе математические лексемы, части которых обладали бы склонностью к изменениям в зависимости от окружающих их графем. Обычно такого рода изменения реализуются функциями и условными операторами, которые сами по себе также предельно аналитичны.

Собственно говоря, аналитичность языка и возможность построить дерево разбора чисто синтаксическими методами (например, анализируя скобки в тексте) — две стороны одной медали.

На полях отметим, что несмотря на всю сложность и неаналитичность разговорных языков, математика предлагает ряд методов их изучения с помощью неклассических логик, например, к таковым относятся субструктурные логики Ламбека, формализующие определенным образом (через некоммутативные алгебры) синтаксическую структуру предложений разговорного языка.

В языке **Math**, как и в разговорных языках можно обнаружить такие явления как **синонимы** и **редукции**.

Редукции в математике распространены не менее щедро, чем в разговорных языках, хотя, как правило, они имеют строго регламентированную природу. Прежде всего это касается скобок. Например, по правилам построения арифметических лексем мы должны записывать выражение $a + bc$ как $(a + (bc))$, однако такое обилие скобок вызывает сложности в восприятии текста, и поэтому существуют стандартные приоритеты операций (как в математике, так и в программировании), позволяющие произвести однозначный синтаксический разбор данной лексемы, предварительно восстановив редуцированные скобки.

В разговорных языках редукции также имеют место, например, типичное французское сокращение *du* для сочетания *de le*, а также немецкое *im* для *in dem*. В отличие от математических редукций разговорные не имеют какой-то общей системы правил и просто сами по себе являются уникальными правилами. Впрочем, и в математике бывают такие частные редукции, которые вводятся скорее как переобозначения «на лету» для удобства локально производимых выкладок, с тем чтобы по завершении этих выкладок забыть о введенном переобозначении.

Термин *синоним* требует отдельного уточнения. Обычно под синонимами понимают два слова, обозначающие одно и то же понятие, либо очень близкие понятия. Например, «кавалерия» и «конница» суть синонимы. Однако, когда мы обсуждаем язык **Math**, мы еще не в состоянии оперировать термином «равенство» и уж тем более термином «близкий» или «похожий», поскольку для этого нам нужно научиться давать определения, а также ввести определенную аксиоматику (хотя бы для равенства). Единственное равенство, которое мы в состоянии предъявить на уровне лексем — это посимвольное тождество лексем. Но в таком случае синонимами у нас могут быть лишь полностью совпадающие лексемы, т.е. синонимов как таковых и нет. Тем не менее, если вспомнить о редукциях и сопутствующих им соглашениях о приоритетах, то мы уже можем построить целый класс синонимичных лексем, считая синонимами те лексемы, которые получаются друг из друга редукцией скобок.

Более того, если привлечь на помощь семантику языка, то можно дать косвенное определение синонимов как *таких лексем, которые имеют одинаковую семантику в рамках конкретной реализации языка Math*. Например, если речь идет о языке арифме-

тики, то синонимами могут быть лексемы 10 и 010, поскольку ведущий ноль в числах обычно игнорируется, но в ряде случаев может быть полезен для синтаксических разборов. Или, например, буквы E и M могут использоваться как синонимы для обозначения математического ожидания, а дробь как правило имеет два равнозначных представления: a/b и $\frac{a}{b}$. Однако такого рода синонимы все-таки редки в математике, а «настоящее» проявление синонимов возможно только после введения равенства в язык, когда принципиально разные по виду лексемы оказываются эквивалентными в силу каких-то логических свойств рассматриваемого языка.

Немного хуже дело обстоит с многозначностью лексем, т.е. тех случаев, когда одна и та же лексема в зависимости от контекста может иметь различную семантику. Достаточно вспомнить букву ω , которая может обозначать и первый бесконечный ординал, и период маятника, и число Эйзенштейна $(-1 + i\sqrt{3})/2$; другой пример — буква i , которая часто выступает как индекс суммирования и как мнимая единица. Здесь мы наблюдаем полную аналогию с разговорными языками, где одно слово может иметь множество смысловых оттенков в зависимости от контекста, а нередко и просто обозначать совершенно разные вещи (например, омографы «зámок» и «замóк», омоформы «мой» (местоимение) и «мой» (глагол повелительного наклонения) и т.д.).

Заметим, что в программировании такие вещи обычно пресекаются интерпретатором, поскольку в данном случае интерпретатором является машина, в то время как в разговорных языках и в математике интерпретатором является человек, которому свойственно экономить на обозначениях в угоду вариациям контекста.

Мы в дальнейшем постараемся придерживаться *контекстно-свободных* лексем, чтобы не запутать читателя еще больше.

Говоря о **финитности конструкций**, мы подразумеваем то обстоятельство, что все лексемы языка **Math** конечны по своей природе. Иначе говоря, каждая лексема изготавливается из конечного набора графем за конечное число шагов. В разговорном языке, разумеется, все конструкции также финитные. И хотя они используют такие неопределенные термины как «и т.д.», «и т.п.», это отсылка к потенциально бесконечному процессу, описываемому данным конечным текстом.⁶

Здесь мы имеем полную аналогию с математическим языком. Тем не менее, такую финитность нашего языка следует, по-видимому, считать, особенностью, поскольку сами математические объекты, которые этим языком описываются, часто имеют актуально бесконечную природу, причем разнообразие этих бесконечностей поражает воображение. И все же, именно в силу конечности самого языка на практике мы оперируем лишь конечным числом объектов или их типов, среди которых есть и самые разные бесконечности. Такой вот парадокс или, лучше сказать, антиномия.

Еще одна особенность языка **Math**, которую мы здесь отметим, — это **строгая детерминированность переменных**.

Под этим словосочетанием следует понимать, прежде всего, свойство «изотропности» переменных, т.е. если данной переменной присвоено какое-то значение (одно конкретное или тип), то во всех вхождениях ниже по тексту эта переменная имеет ровно такое же значение, пока она не будет явным образом переопределена. Иначе говоря, если мы где-то выполнили присвоение $a \stackrel{\text{def}}{=} 2$, то все вхождения a далее имеют значе-

⁶ В математике вместо и т.д. мы используем многоточие, но это скорее элемент метаязыка, чем какого-либо формализма. В формальных системах иногда допускаются бесконечные формулы или секвенции, например, ω -правило в языке PA_ω , но все такие бесконечности имеют строго регулярную, конечно описываемую конструкцию.

ние 2, сколько бы раз эта переменная ни появилась в тексте вплоть до следующего переопределения (отметим, что здесь важную роль играет направленность текста). Это свойство переменных мы будем подразумевать в дальнейшем, записывая подстановки вида $[a/x]$ (это омограф дроби, а не дробь!), которые означают, что в данной формуле требуется все вхождения переменной a (как подстроки) заменить на x .

Заметим, что в разговорных языках самые общие аналоги переменных, т.е. местоимения, не подчиняются такому свойству. Например, в тексте

«Она должна ехать на юг. Река здесь есть? — Она за лесом»

значение слова *она* меняется прямо «на лету», при этом читатель однозначно может восстановить это значение из контекста. В математике и программировании такие фокусы недопустимы.

Тем не менее, кое-какой аналог контекстного переопределения переменной имеется и в языке **Math**. Речь идет о **лексемах-свертках**. В свертке, во-первых, всегда присутствует переменная (хотя бы одна), а во-вторых, эта переменная имеет строго ограниченную область действия. Например, в лексеме $\forall x \in A \ (x = x^2)$ мы принудительно ограничиваем значения переменной множеством A и для таких значений далее заявляем некое равенство, стоящее в скобках. При этом мы можем следом написать еще одну лексему вроде $\exists x > 0 \ (x^2 < 0)$, и будет понятно, что теперь область значений переменной x сменилась с множества A на множество положительных чисел, и эти области значений переменной x актуальны только для формулы, стоящей следом в скобках. В программировании такое ограничение области действия переменной наблюдается, например, в циклах и функциях, где переменная имеет строго локальный контекст. Тем не менее, как видим, переопределение переменной «на лету» строго формализовано и не допускает разночтений.

Таким образом, свертка позволяет нам синтаксически обособить некоторый фрагмент текста и для него задать свои допустимые значения переменных. Более того, можно встретить иногда вложенные свертки, использующие одну и ту же переменную, например, такие:

$$\sum_{k=1}^{\infty} k^2 \left(\sum_{k=-10}^{10} k a_k \right)^3 + k^3 \left(\sum_{k=-100}^{100} k^2 b_k \right)^5.$$

В принципе, понятно, что «внешнее» k здесь не имеет никакого отношения к обоим «внутренним» k , которые тоже никак не связаны друг с другом, т.е. мы имеем дело со свертками внутри свертки, в каждой из которых переменная k принимает разные, вообще говоря, значения. В этом можно усмотреть аналог той текстовой фразы с местоимением «она», которую мы приводили выше. Но нужно отметить, что лексемы такого вида хоть и законны, но как правило не используются. Дело в том, что они будут работать ровно до тех пор, пока не потребуются внешнюю переменную k использовать совместно с внутренней в одной формуле. В самом деле, рассмотрим такую лексему:

$$\sum_k \int_0^{\infty} e^{itk-t^2} dt.$$

Здесь сверточные переменные k и t встречаются в общем арифметическом выражении $itk - t^2$. Что случится, если мы их обозначим одинаково? Получится такая лексема:

$$\sum_x \int_0^{\infty} e^{ixx-x^2} dx.$$

И здесь мы перестаем уже понимать, по какой части формулы надо интегрировать, а что является параметром суммы.

Поэтому общий подход к вложенным сверткам заключается в том, *чтобы все сверточные переменные были разными*. Более того, в современной логике вообще принято разделять переменные на два класса: *свободные* и *связанные*. Связанные переменные используются для формирования свертки, а свободные переменные обозначают внешние параметры. Такой подход позволяет избегать описанных выше коллизий с названиями переменных.

Наконец, отметим также, что в языке **Math** все повествование происходит в **изъявительном наклонении**, т.е. в математике мы описываем вещи как они есть. В тексте доказательств мы можем прибегать к использованию условного наклонения, например, при доказательстве методом от противного, когда нам нужно что-то предположить, а затем опровергнуть, но на формальном уровне это никак не проявляется.⁷ Тем более в математике не используется повелительное наклонение иначе как при присваивании значений переменных. Мы можем сказать «пусть $x = 2$ », и это выглядит как повелительное наклонение, однако на формальном уровне это записывается как посылка импликации.

Последнее, что стоит отметить как особенность математического языка, — это отсутствие изменений имен объектов (аналоги существительных) и операций (аналоги глаголов) по лицам, числам, родам, падежам, временам, видам. Здесь уместно вспомнить, что и для разговорных языков не все эти атрибуты обязательны, например, в английском языке нет родов и падежей, а также видов глаголов, зато есть большое число времен. В математике же таких категорий изменений объектов и суждений нет. Например, операция пересечения множеств ($\cap$) не может со временем состариться и перестать работать так, как мы ее определили. Или теорема Пифагора не может устареть в том смысле, что вчера она была верна, а завтра перестанет быть верной. Есть, конечно, изменения символики с течением времени (как, например, устаревшее обозначение импликации $\supset$), но это скорее эволюция языка, чем временные формы его единиц.

Однако же отсутствие времен и падежей — не повод облегченно вздохнуть, поскольку в математике имеются свои собственные способы изменения имен и операций.

Чаще всего это связано с тем, что мы экономим на количестве используемых символов, чтобы текст не выглядел громоздким, и поэтому многие символы определяем в ходе изложения материала. Например, знак $\cong$ часто используется для обозначения изоморфизма, но какого конкретно — определяется в тексте. Символ ω может обозначать как первый бесконечный ординал, так и число Эйзенштейна, либо использоваться как обозначение дифференциальной формы и т.п. Все такие определения обозначений могут зависеть как от области математики, в которой мы работаем, так и от конкретной статьи или книги. Ну а такие ходовые буквы как a, b, c, x, y, z переопределяются в ходе повествования регулярно. То же самое можно сказать и о таких обозначениях операторов, как штрих производной, «крышка», возведение в степень и т.д., смысл которых варьируется от книги к книге, от одной области математики к другой. И единых правил (вроде таблицы спряжений) для этих изменений нет.

В этом заключается абсолютная свобода математики, и часто это же служит препятствием для нематематиков к пониманию математических текстов. С другой стороны, любая статья или книга по математике не состоит из одного лишь формального текста, обычно не менее половины ее объема составляют лексемы разговорного языка, часть

⁷ Точнее, в формальной логике мы используем правило контрапозиции: $((\mathcal{A} \rightarrow \mathcal{B}) \wedge \neg \mathcal{B}) \rightarrow \neg \mathcal{A}$.

из которых являются математическими терминами, а подавляющее большинство — обычными речевыми конструкциями разговорного языка, призванными пояснить то, что написано в формальной части. Это позволяет воспринимать чисто формальный материал через неформальные образы и понятия.

A1.8. Вопросы для самоконтроля

- Q1** Что такое «принцип матрешки»? [стр. 20]
- Q2** Перечислите как минимум три признака естественных языков, которыми обладает математический язык (*Math*). [стр. 18]
- Q3** В чем заключается основное отличие математического языка от естественных языков? [стр. 18]
- Q4** Что такое графемы и жесткие графемы в математическом языке? Приведите примеры. [19]
- Q5** Объясните на примере формулы Кардано (A1.1), как используется «принцип матрешки» для упрощения сложных выражений.
- Q6** Что такое дерево разбора и какую роль оно играет в понимании структуры математических формул? [стр. 23]
- Q7** В чем разница между синтаксисом и семантикой математического языка? [стр. 25]
- Q8** Что такое предикаты и термы в математическом языке? Приведите примеры из текста. [стр. 26]
- Q9** Что такое лексема-свертка? [стр. 32]

A1.9. Упражнения

Ex1. Постройте дерево разбора для формулы корней квадратного уравнения:

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

Укажите иерархию используемых шаблонов (дробь, сложение, вычитание, квадратный корень, возведение в степень, умножение).

Ex2. В математике скобки часто опускаются на основе приоритета операций. Восстановите все опущенные скобки в следующих выражениях, чтобы сделать структуру однозначной:

1. $a \cdot b + c \cdot d$;
2. $x^2 + 2xy + y^2$;
3. $a/b/c$ (предполагая стандартную левоассоциативность);
4. e^{x^2+y} .

Ex3. Нарисуйте (или опишите) дерево разбора для тождества Эйлера, определив иерархию используемых шаблонов:

$$e^{i\pi} + 1 = 0$$

Ex4. Декомпозируйте следующее вложенное выражение по слоям. Что является самым внешним шаблоном, и каковы внутренние компоненты?

$$\sqrt{1 + \sqrt{1 + \sqrt{1 + x}}}$$

Ex5. В выражении ниже определите, какие символы выступают как объекты (существительные/переменные/константы), а какие как действия/отношения (глаголы/операторы):

$$\frac{d}{dx}(x^3 + 3x) = 3x^2 + 3$$

Ex6. Переведите следующие предложения естественного языка на язык логики предикатов, используя кванторы ($\forall, \exists$), логические связки ($\wedge, \vee, \rightarrow, \neg$) и подходящие предикаты. Пример: «Все французы говорят по-французски». Универсум: Все люди. Предикаты: $Fr(x)$ — « x француз», $Speak(x, French)$ — « x говорит по-французски». Формула: $\forall x(Fr(x) \rightarrow Speak(x, French))$.

1. «Некоторые французы говорят по-английски».
2. «Ни один француз не говорит по-китайски». (Подсказка: перефразируйте как «Для любого человека, если он француз, то он не говорит по-китайски»).
3. «Есть физики, которые любят математику, но не любят квантовую механику».
4. «Любой, кто изучает математическую логику, должен знать теорию множеств».
5. «Существует физик, который предложил теорию, которую понимает каждый математик».
6. «У каждого человека есть ровно один биологический отец». (Подсказка: «существует ровно один» $\exists!$ можно записать как $\exists x(P(x) \wedge \forall y(P(y) \rightarrow y = x))$).
7. «Не все то золото, что блестит».
8. «Если физик не знает математики, он не поймет ни Общей теории относительности, ни Квантовой механики».
9. «Не существует теории, которая объясняет абсолютно всё».

Ex7. Переведите следующие формальные предложения на естественный русский язык (старайтесь, чтобы это звучало естественно, а не механически):

1. $A \subseteq B \wedge B \subseteq A \rightarrow A = B$;
2. $\forall x \in \mathbb{R} : x^2 \geq 0$;
3. $n, m \in \mathbb{Z} \wedge n \neq 0 \rightarrow \exists q, r : (m = nq + r \wedge 0 \leq r < |n|)$.

Ex8. Переведите следующие утверждения на язык **Math**:

1. Квадрат суммы a и b равен сумме их квадратов плюс удвоенное произведение.

2. Произведение любого числа на ноль равно нулю.
3. Для каждого положительного числа ε существует положительное число δ .

Ex9. Определите, какие из следующих строк являются синтаксически корректными математическими выражениями, а какие — «абракадаброй». Объясните почему.

1. $5 + (3 \cdot 2)$;
2. $5 + \cdot (3 \ 2)$;
3. $\sin(x) = \cos$;
4. $\int_a^b f(x)dx$.

Ex10. В следующих выражениях определите *связанную* переменную (итератор) и *свободные* переменные (параметры):

1. $\sum_{k=1}^n k^3$;
2. $\lim_{x \rightarrow a} (x^2 - 3)$;
3. $\bigcup_{i \in I} A_i$.

Ex11. Рассмотрим выражение $\sum_{i=1}^n (i + \sum_{j=1}^i j)$. Почему запись $\sum_{i=1}^n (i + \sum_{i=1}^i i)$ была бы сбивающей с толку или некорректной?

Ex12. Выполните синтаксическую подстановку $[x/(a+b)]$ (замените x на лексему $a+b$) в следующих выражениях. Будьте внимательны со скобками!

1. $3x$;
2. x^2 ;
3. $\sin(x)$.

Ex13. В формуле площади круга $S = \pi r^2$:

1. Какие символы являются константами?
2. Какие символы являются переменными?

Ex14. Восстановите скобки в выражении 2^{3^4} согласно стандартному математическому соглашению для степеней (сверху вниз / правоассоциативность). Вычислите значения $(2^3)^2$ и $2^{(3^2)}$, чтобы продемонстрировать разницу при расстановке скобок.

Ex15. Посмотрите на выражение $f^{-1}(x)$. В зависимости от контекста, синтаксически оно может означать две совершенно разные вещи. Что это за вещи? (Подсказка: обращение функции vs возведение в степень.)

Ex16. Классифицируйте следующие операторы как унарные (1 аргумент) или бинарные (2 аргумента):

1. $-$ в выражении -5 ;
2. $-$ в выражении $a - b$;
3. $!$ в выражении $n!$;
4. $\cup$ в выражении $A \cup B$.

У чертога логики

Но тебе придется примириться с этим, — возразил Воланд, и усмешка искривила его рот, — не успел ты появиться на крыше, как уже сразу отвесил нелепость, и я тебе скажу, в чем она, — в твоих интонациях. Ты произнес свои слова так, как будто ты не признаешь теней, а также и зла. Не будешь ли ты так добр подумать над вопросом: что бы делало твое добро, если бы не существовало зла, и как бы выглядела земля, если бы с нее исчезли тени? Ведь тени получаются от предметов и людей. Вот тень от моей шпаги. Но бывают тени от деревьев и от живых существ. Не хочешь ли ты ободрать весь земной шар, снеся с него прочь все деревья и все живое из-за твоей фантазии наслаждаться голым светом?

— М. Булгаков, «Мастер и Маргарита»

Приведенная цитата из классика показывает нам важность дифференциации чего бы то ни было хотя бы на два класса. При этом мы не заостряем внимание на том, как именно называются эти классы, имеют ли эти названия какой-то смысловой подтекст или являются абракадаброй. Важно то, что если совокупность всего того, что мы в принципе могли бы описывать языком **Math** (или любым другим языком), нельзя было бы разделить на категории, то мы принципиально не смогли бы различать предметы, события, явления, а также видеть какие-либо отношения между ними. Вся вселенная у нас схлопнулась бы в одну точку.

По сути **логика** как таковая представляет собой комплекс правил и методов, позволяющих некоторым регулярным способом отличать «свет» от «тьмы», обнаруживать сходства и различия, отделять существенное от несущественного, находить связи и структуры в том объеме знаний, к которому мы применяем язык **Math** как описательное средство. В этом и следующем разделах мы поговорим о логике в общем виде, понимая ее как некоторое неотъемлемое свойство нашего разума, позволяющее выстраивать правильные суждения.

Первый же вопрос, который может возникнуть у читателя: зачем производить какое-то деление на классы, если они и так уже есть? Действительно, ведь при определении языка **Math** мы научились строить синтаксически правильные лексемы. Более того, мы предъявили вполне конкретный алгоритм, как это делать, отправляясь от некоторых предопределенных подстановочных шаблонов и алфавитных символов. Наконец, по любому тексту мы даже можем за конечное время распознать, является ли он правильно построенной лексемой или нет. Давайте тогда считать лексемы «истинными» текстами, а все остальные комбинации символов — «ложными» текстами.

Такой догматический подход таит в себе ряд проблем. Во-первых, мы тем самым построили не более чем логику лексем. То есть мы с помощью этой логики умеем опре-



ОЗНАКОМИТЕЛЬНЫЙ ФРАГМЕНТ

Выше приведены первые 43 страницы книги: содержание, ключевые теоремы, предисловие и уровень A1 «Выживание» целиком.

Далее следует фрагмент из конца книги — *Список обозначений* (с. 345–348), сводка всей нотации, которая используется в тексте.

СПИСОК ОБОЗНАЧЕНИЙ

Math	Неформальный язык математики (предмет этой книги)
$\neg \varphi$	Отрицание формулы φ
$\varphi \wedge \psi$	Конъюнкция («И»)
$\varphi \vee \psi$	Дизъюнкция («ИЛИ»)
$\varphi \rightarrow \psi$	Импликация («ЕСЛИ..., ТО...»)
$\varphi \leftrightarrow \psi$	Эквиваленция («тогда и только тогда»)
$\varphi \equiv \psi$	Формулы φ и ψ равносильны
$\vdash$	Синтаксическая выводимость (доказуемость)
$\models$	Семантическое (логическое) следование
$\models$	Отношение истинности в модели
$\top$	Тождественная истина
$\perp$	Тождественная ложь
$\stackrel{\text{def}}{=}$	Равенство по определению
$\stackrel{\text{def}}{\leftrightarrow}$	Равносильность по определению
$\text{Th}(\mathcal{M})$	Полная (элементарная) теория модели $\mathcal{M}$
$\text{Pr}(n, m)$	Предикат доказуемости: формула с кодом n имеет доказательство с кодом m
$\text{Con}(T)$	Формула, выражающая непротиворечивость теории T
$\ulcorner \gamma \urcorner$	Геделев номер выражения γ
PA	Арифметика Пеано
Q	Арифметика Робинсона
ZF	Теория множеств Цермело–Френкеля
ZFC	Теория множеств Цермело–Френкеля с аксиомой выбора
HF	Теория наследственно-конечных множеств
AC	Аксиома выбора (общая)

AC_{ω}	Аксиома счетного выбора
AC_{ω}^{fin}	Аксиома счетного выбора из конечных множеств
ZT	Теорема Цермело
ZL	Лемма Цорна
CH	Континуум-гипотеза Кантора
GB	Теория множеств Гёделя–Бернайса
$\forall x \varphi(x)$	Квантор всеобщности: для любого x истинно $\varphi(x)$
$\exists x \varphi(x)$	Квантор существования: существует x такой, что $\varphi(x)$
$\exists! x \varphi(x)$	Существует единственный x такой, что $\varphi(x)$
$\forall x \in A : \varphi(x)$	Ограниченный квантор всеобщности в теории множеств
$\exists x \in A : \varphi(x)$	Ограниченный квантор существования в теории множеств
$\forall x < k : \varphi(x)$	Ограниченный квантор всеобщности в арифметике
$\exists x < k : \varphi(x)$	Ограниченный квантор существования в арифметике
$\forall \alpha : \varphi(\alpha)$	Для любого ординала α истинно $\varphi(\alpha)$
$\exists \alpha : \varphi(\alpha)$	Существует ординал α такой, что $\varphi(\alpha)$
$\emptyset, \{\}$	Пустое множество
$a \in b$	a является элементом множества b
$a \notin b$	Отрицание $a \in b$
$A \subseteq B$	Множество A является подмножеством B
$A \subsetneq B$	Множество A является собственным подмножеством B
$\{a, b, \dots, c\}$	Множество, состоящее из элементов $a, b, \dots, c$
$\{x \mid \varphi(x)\}$	Множество всех x , удовлетворяющих свойству $\varphi(x)$
$A \cup B$	Объединение множеств A и B
$A \cap B$	Пересечение множеств A и B
$\bigcup A$	Объединение всех элементов множества A
$\bigcap A$	Пересечение всех элементов множества A
$A \setminus B$	Разность множеств A и B
$A \Delta B$	Симметрическая разность множеств A и B
$\mathcal{P}(A)$	Множество всех подмножеств A (булеан)

$A \sqcup B$	Дизъюнктное объединение множеств A и B
$\langle a, b \rangle$	Упорядоченная пара
$A \times B$	Прямое (декартово) произведение множеств A и B
$\mathfrak{U}$	Универсум фон Неймана, класс всех множеств
$TC(A)$	Транзитивное замыкание множества A
$\text{rank}(x)$	Ранг множества x
V_α	Универсум фон Неймана ранга α
$S(x)$	Операция взятия последователя (в PA и ZF)
$\omega, \aleph_0$	Первый бесконечный ординал (кардинал)
$ A , \#A$	Мощность (кардинальное число) множества A
$\text{Card}(A)$	Кардинальность: класс всех множеств, равномоощных A
$\mathfrak{c}, 2^{\aleph_0}$	Мощность континуума
$A \simeq B$	Множества A и B равномоощны (существует биекция)
$A \leqslant B$	Мощность A не больше мощности B (существует инъекция)
$A < B$	Мощность A строго меньше мощности B
Ord	Класс всех ординалов
Succ, Lim	Классы последующих и предельных ординалов
Card	Класс всех кардиналов
$\text{ot}(X, <)$	Порядковый тип вполне упорядоченного множества
$\mathfrak{h}(A)$	Число Хартогса для множества A
$f : A \rightarrow B$	Функция f из множества A в множество B
$f : A \leftrightarrow B$	Биекция между множествами A и B
$\hookrightarrow$	Вложение (инъекция) с сохранением порядка
$\text{dom}(f)$	Область определения функции f
$\text{ran}(f)$	Область значений функции f
$f \upharpoonright D$	Сужение функции f на множество D
f^{-1}	Обратная функция
R^{-1}	Обратное отношение
χ_A	Индикаторная функция множества A

$\text{supp}(f)$	Носитель функции f
$F^{fin}(Y, X)$	Множество функций из Y в X с конечным носителем
$\text{pr}_A(p)$	Проекция на компоненту A
aRb	$\langle a, b \rangle \in R$ (элементы a, b находятся в отношении R)
$[a]_R$	Класс эквивалентности элемента a по отношению R
$A/_R$	Фактор-множество множества A по отношению эквивалентности R
$\mathfrak{A} \cong \mathfrak{B}$	Структуры $\mathfrak{A}$ и $\mathfrak{B}$ изоморфны
$\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$	Стандартные числовые системы
$a = b$	Равенство
$a < b, a > b$	Строгое неравенство
$a \leq b, a \geq b$	Нестрогое неравенство
$a + b$	Сложение
$a \cdot b, ab$	Умножение
$a \div b$	Монус (усеченное вычитание)
$a \mid b$	a делит b
$a \equiv b \pmod{m}$	a сравнимо с b по модулю m
$\underline{n}$	Нумерал, терм вида $S \dots S0$, представляющий число n
$\beta(c, d, i)$	β -функция Гёделя для кодирования последовательностей
$\text{Code}(S)$	Функция кодирования множества натуральным числом
Λ	Пустая строка (в грамматиках)
$\max A, \min A$	Максимальный/минимальный элемент множества A
$\sup A, \inf A$	Точная верхняя/нижняя грань множества A

МАТЕМАТИКА КАК ИНОСТРАННЫЙ

или архитектура математической мысли

Н. И. Казимиров

Полная версия — 354 страницы: пять уровней от A1 до C1,
доказательная база и около 190 упражнений с решениями.

СКОРО НА ПЛОЩАДКАХ

ЛитРес · Ozon · Wildberries

АНГЛИЙСКОЕ ИЗДАНИЕ

Springer, 19 сентября 2026

link.springer.com/book/9783032307200

СТРАНИЦА КНИГИ

mathem.at/Materials/maglish_index.html