

А. Н. Колмогоров, А. Г. Драгалин

МАТЕМАТИЧЕСКАЯ ЛОГИКА

ДОПОЛНИТЕЛЬНЫЕ ГЛАВЫ

Допущено Министерством высшего и среднего специального образования СССР в качестве учебного пособия для студентов вузов, обучающихся по специальности «Математика»

ИЗДАТЕЛЬСТВО
МОСКОВСКОГО УНИВЕРСИТЕТА
1984

Колмогоров А. Н., Драгалин А. Г. Математическая логика. Дополнительные главы: Учеб. пособие. — М.: Изд-во Моск. ун-та, 1984. — 120 с

Книга представляет собой вторую часть учебного пособия авторов «Введение в математическую логику» (Изд-во Моск. ун-та, 1982 г.), но может изучаться и самостоятельно. Излагаются фундаментальные факты математической логики: начала аксиоматической теории множеств, теория алгоритмов, теорема о полноте исчисления предикатов, теорема Гёделя о неполноте. Обсуждается программа Гильберта обоснования математики

Рецензенты:

кафедра математического анализа МГПИ имени В. И. Ленина;
кандидат физ.-мат. наук В. Е. Плиско

ОГЛАВЛЕНИЕ

Предисловие	4
Введение	6
Глава I. ТЕОРИЯ МНОЖЕСТВ	15
§ 1. Язык наивной теории множеств, парадоксы наивной теории множеств	15
§ 2. Язык теории множеств Цермело — Френкеля	24
§ 3. Отношения и функции в языке теории множеств	26
§ 4. Натуральные числа в теории множеств. Запись математических утверждений в языке теории множеств	34
§ 5. О континуум-гипотезе и аксиоме выбора	42
§ 6. Аксиоматическая теория множеств Цермело — Френкеля	44
Глава II. ЭЛЕМЕНТЫ ТЕОРИИ АЛГОРИФМОВ	56
§ 1. Машины Тьюринга	56
§ 2. Тезис Чёрча	64
§ 3. Рекурсивные и рекурсивно-перечислимые множества и предикаты	65
§ 4. Прimitивно-рекурсивные функции, геделева нумерация, арифметика с примитивно-рекурсивными термами	73
§ 5. Некоторые теоремы общей теории алгорифмов	81
Глава III. ЭЛЕМЕНТЫ ТЕОРИИ ДОКАЗАТЕЛЬСТВ	89
§ 1. Неполнота и неразрешимость аксиоматических теорий	89
§ 2. Теорема Геделя о полноте исчисления предикатов	97
§ 3. Теорема об устранении сечения	104
§ 4. О программе Гильберта обоснования математики	112
Литература	119

ПРЕДИСЛОВИЕ

Книга представляет собой вторую часть первоначального курса математической логики [1], но может читаться и независимо, если читатель имеет некоторое предварительное знакомство с логико-математическими языками и теорией логического вывода. Необходимые предварительные сведения можно получить как в нашей книге [1], так и в первых главах любого из более подробных курсов математической логики, например [2], [3], [4]. Необходимый минимум сведений и терминологию мы напоминаем также во введении.

Книга возникла в результате обработки конспектов лекций семестрового курса математической логики для студентов первого курса механико-математического факультета Московского университета, читавшегося обоими авторами. В первой книге мы стремились познакомить читателя с основными понятиями математической логики, правильным обращением с логической символикой, логическими законами, техникой логического вывода, что составляет, на наш взгляд, минимум сведений, полезных в работе математика любой специальности. В настоящей второй части большее внимание уделяется изложению некоторых фундаментальных результатов математической логики, представляющих общематематический интерес. Предполагается, что при построении курса лектор может выбрать из предложенного материала ту или иную тему в зависимости от потребностей учебного плана или аудитории.

В первой главе излагается теория множеств в стиле аксиоматической системы Цермело — Френкеля. Мы стремились показать, как основные математические понятия и структуры могут быть введены на базе точного логико-математического языка теории множеств. Здесь же обсуждаются традиционные вопросы, относящиеся к основаниям теории множеств: парадоксы теории множеств, непротиворечивость системы Цермело — Френкеля, парадокс Сколема, содержательный аксиоматический метод и формальный аксиоматический метод в математике.

Во второй главе излагаются элементы теории алгорифмов. Здесь даны точные определения, касающиеся вычислимости по Тьюрингу, обсуждаются тезис Чёрча и понятия рекурсивного и рекурсивно-перечислимого множества. Эта часть главы может рассматриваться как обязательный минимум по теории алгорифмов. Далее приводятся основные теоремы общей теории

алгорифмов относительно существования неразрешимых множеств и предикатов и излагается подготовительный материал по геделевой нумерации конструктивных объектов и выводам свойств конструктивных объектов в формальной арифметике. Этот материал может рассматриваться как факультативный по отношению к обязательному курсу логики, лектор может использовать его выборочно или перенести часть материала на семинарские занятия.

Третья глава посвящена теории вывода. Здесь доказываются теорема Геделя о полноте исчисления предикатов, теорема Левенгейма — Сколема, теорема о неполноте и неразрешимости всякой достаточно выразительной формальной аксиоматической теории. Доказываются также знаменитая вторая теорема Геделя о невозможности доказательства непротиворечивости достаточно мощной формальной аксиоматической теории средствами самой этой теории и теорема Генцена об устранении сечения. Завершается глава обсуждением программы Гильберта обоснования математики. Материал третьей главы также может рассматриваться как факультативный. Например, лектор может ограничиться лишь формулировками некоторых фундаментальных теорем.

Сложность изложения многих важных результатов математической логики состоит в том, что они часто требуют для своего доказательства большого подготовительного аппарата для аккуратного проведения деталей доказательств. Так, необходимо убеждаться, что те или иные предикаты действительно вычислимы по Тьюрингу или примитивно-рекурсивны, что рассматриваемые формулы действительно выводимы в тех или иных формальных аксиоматических теориях. Такого рода утверждения доказываются обычно путем громоздкого, но в принципе нетрудного непосредственного построения соответствующих машин Тьюринга, примитивно-рекурсивных описаний, формальных выводов и т. п. Разумеется, в коротком курсе нецелесообразно тратить время на такие построения. Читателя, интересующегося деталями построений, мы в таких случаях отсылаем к более подробным руководствам, но все же стремимся к тщательному изложению всех принципиальных моментов доказательств.

Курс логики предполагает выполнение упражнений на семинарских занятиях. С этой целью следует использовать специальные задачки, например известный сборник [19]. Упражнения в тексте не могут заменить такого задачника. Все упражнения в тексте легкие, обязательны для выполнения и предназначены для самоконтроля.

Знак \triangleright в тексте отмечает начало доказательства, а знак \square — его окончание. Знаки \Rightarrow , \Rightarrow , \Leftrightarrow заменяют словесные обороты «есть по определению», «если..., то», «тогда и только тогда, когда» соответственно.

Авторы

Напомним некоторые определения и понятия, нужные для дальнейшего. Подробное изложение имеется, например, в нашей книге [1], но может быть почерпнуто и из начальных глав более подробных учебников, предназначенных для лиц, специализирующихся по математической логике (см. [2], [3], [4]).

Математические суждения в логике записываются в виде формул в точно определенных логико-математических языках. Основными объектами данного логико-математического языка являются его *выражения*. Выражения подразделяются на *формулы* и *термы*. Язык Ω содержит несколько сортов *переменных*. Переменные каждого сорта рассматриваются как пробегающие некоторую область — множество объектов данного сорта. Вхождения переменных в выражения языка делятся на *свободные* и *связанные*. Переменные, входящие в выражение свободно, называются его *параметрами*. Выражение, не содержащее параметров, называется *замкнутым*.

Выражения логико-математического языка представляют собой строчки символов-букв из алфавита языка и строятся по некоторым строго определенным правилам. Так, каждая формула S данного языка Ω может иметь один и только один из следующих семи видов:

- $(A \wedge B)$ — читается « A и B »,
- $(A \vee B)$ — читается « A или B »,
- $(A \supset B)$ — «из A следует B »,
- $\neg A$ — «не A »,
- $\forall x A$ — «для всякого x имеет место A »,
- $\exists x A$ — «существует x такое, что A »,

атомарная формула.

Символы \wedge , \vee , \supset , \neg называются *логическими связками* и имеют соответственно наименования: *конъюнкция*, *дизъюнкция*, *импликация* и *отрицание*. Символы \forall , \exists называются *кванторами* (квантор общности и квантор существования). Как обычно, считается, что кванторные приставки $\forall x$, $\exists x$ связывают переменную x в последующей формуле, так что в формулу, начинающуюся с такой приставки, переменная x не входит свободно.

Строение атомарных формул зависит от рассматриваемого языка. Особенно просто устроены атомарные формулы у языков первого порядка, которые и составляют основной объект изучения в математической логике. А именно атомарная формула языка первого порядка имеет вид $P(t_1, \dots, t_n)$, где P — предикатная буква языка, а t_1, \dots, t_n — термы языка. Здесь возможен и случай $n=0$, тогда 0-местная предикатная буква P сама по себе является атомарной формулой и называется в этом случае пропозициональной буквой, или пропозициональной переменной. n -местная предикатная буква изображает в языке предикат от n переменных, в иной терминологии, параметрическое суждение, высказывательную форму от n переменных.

Термы также в разных языках устроены по-разному. И здесь самым простым и важным случаем является случай языков первого порядка. В языках первого порядка термы строятся из переменных и констант языка с помощью функциональных символов, т. е. каждый терм есть либо переменная, либо константа языка, либо имеет вид $f(t_1, \dots, t_n)$, где f — функциональный символ, обозначающий в языке n -местную операцию. В языках первого порядка все переменные, входящие в терм, являются свободными, т. е. являются параметрами этого терма.

При практическом написании формул и термов мы экономим скобки, пользуясь хорошо известными соглашениями. Так, обычно опускаются внешние скобки. Кроме того, мы располагаем связки и кванторы в определенном порядке, считая, что те символы, которые в этом порядке находятся правее, «связывают сильнее», т. е. что их следует выполнять в первую очередь. Для конкретности мы придерживаемся следующего порядка выполнения операций:

$$\equiv \supset \vee \wedge \neg \forall \exists$$

Здесь \equiv — производная логическая связка, $(A \equiv B)$ есть сокращенное обозначение формулы $((A \supset B) \wedge (B \supset A))$.

Например, формулу

$$(\forall x(P(x, y) \supset (\forall zQ(z) \wedge R)) \vee Q(x))$$

можно записать в виде

$$\forall x(P(x, y) \supset \forall zQ(z) \wedge R) \vee Q(x).$$

Иногда для дальнейшей экономии скобок мы используем точку. Если внутри скобок выполняется несколько однородных (по силе связывания) логических символов, то точкой внизу у символа мы отмечаем главный логический символ, т. е. тот, который выполняется в последнюю очередь. Например, формулу

$$P \supset (Q \vee R \equiv (\neg R \equiv \neg P))$$

можно записать в виде

$$P \supset (Q \vee R \equiv \neg R \equiv \neg P).$$

Если T — выражение языка, x_1, \dots, x_n — различные переменные и t_1, \dots, t_n — термы соответствующих сортов, то через $(T(x_1, \dots, x_n \parallel t_1, \dots, t_n))$ мы обозначим результат *правильной подстановки* термов t_1, \dots, t_n вместо свободных вхождений переменных x_1, \dots, x_n в T . Для получения выражения $(T(x_1, \dots, x_n \parallel t_1, \dots, t_n))$ следует заместить одновременно все свободные вхождения переменных x_1, \dots, x_n на термы t_1, \dots, t_n соответственно. При этом, если необходимо, следует переименовать некоторые связанные переменные выражения T . А именно, если некоторая переменная x некоторого терма t_i , будучи свободной в t_i , оказывается связанной после подстановки (в таких случаях говорят, что терм t_i *не свободен* для подстановки, или что происходит *коллизия* переменных), то следует в T предварительно переименовать некоторые связанные переменные на новые, чтобы при подстановке не происходило коллизий. Например, результатом правильной подстановки

$$(\exists x P(x, y) (y \parallel f(x, y)))$$

является формула

$$\exists u P(u, f(x, y))$$

с новой связанной переменной u .

Выражение $(T(x_1, \dots, x_n \parallel t_1, \dots, t_n))$ сокращаем до $T(x_1, \dots, x_n \parallel t_1, \dots, t_n)$ или даже до $T(t_1, \dots, t_n)$, если упоминание о переменных x_1, \dots, x_n несущественно. Следует помнить, что в записях вида $A(t, r)$ всегда имеется в виду именно правильная подстановка с необходимым переименованием связанных переменных.

Мы предполагаем, что читатель знаком с обычными языками первого порядка, такими как язык арифметики, язык теории групп, язык логики предикатов. Впрочем, язык логики предикатов и язык арифметики мы напомним ниже.

Выражения языка суть просто строчки символов специального вида и сами по себе не имеют никакого смысла. Для придания смысла выражениям языка Ω необходимо задать *интерпретацию для языка Ω* (в другой терминологии, *структуру для языка Ω , модель для языка Ω*). Задать интерпретацию M для языка Ω означает, в частности, задать *область пробега* для каждого сорта переменных, фигурирующих в языке. Если в произвольном выражении языка заместить все его параметры объектами из соответствующих областей пробега, то получится то, что называется *оцененным выражением* языка (точнее, выражением, оцененным в данной модели M). В частности, замкнутое выражение само по себе является оцененным в любой модели.

Каждая интерпретация M языка позволяет по естественным правилам подразделить все оцененные формулы на *истинные* в данной интерпретации и на *ложные* в данной интерпретации. Если A — формула, оцененная в M , то запись $M \models A$ будет означать, что A истинна в интерпретации M .

Понятие истинности оцененной формулы в структуре M согласовано со строением формулы. Например, $M \models A \wedge B$ тогда и только тогда, когда $M \models A$ и $M \models B$. Далее, $M \models A \vee B$ тогда и только тогда, когда $M \models A$ или $M \models B$. $M \models \neg A$ равносильно тому, что неверно $M \models A$. Утверждение $M \models A \supset B$ ложно лишь в одном случае, а именно когда $M \models A$ и $M \models \neg B$. При остальных комбинациях истинностных значений утверждение $M \models A \supset B$ считается истинным (так понимаемую логическую связку \supset часто называют *материальной импликацией*). Далее, $M \models \forall x A$ тогда и только тогда, когда для всякого объекта a из области пробегания переменной x имеем $M \models A(x||a)$. Аналогично $M \models \exists x A$ равносильно тому, что существует объект a из нужной области, такой, что $M \models A(x||a)$. Коротко можно сказать, что мы придерживаемся классической семантики, где логические связи трактуются по законам двузначной булевой алгебры.

Подобным образом интерпретация M позволяет приписать значение $|t|_M$ каждому оцененному терму языка. Значением $|t|_M$ является объект, который выражает терм t при данной оценке своих параметров. Значение терма зависит только от значений составляющих его подтермов. Это означает, что, если r — терм, в котором оценены все параметры, за исключением, может быть, x , и t — оцененный терм того же сорта, что и x , то $|r(x||t)|_M$ совпадает с $|r(x|||t|_M)|_M$.

Аналогично, $M \models A(x||t)$ тогда и только тогда, когда $M \models A(x|||t|_M)$.

Таким образом, при фиксированной интерпретации M языка Ω формула A этого языка рассматривается как выражающая некоторое параметрическое суждение, зависящее от оценки параметров. Как говорят, формула задает в Ω *высказывательную форму*. В частности, если формула является предложением Ω , т. е. вовсе не содержит параметров, то в M такая формула определяет конкретное истинное или ложное высказывание. Аналогично терм t задает в M операцию от своих параметров. Замкнутый терм задаст в интерпретации M некоторый конкретный объект.

С точки зрения логики особенно интересны формулы языка, истинные в любой интерпретации для Ω при любой оценке своих параметров. Такие формулы называются *логическими законами* языка (в другой терминологии — *общезначимыми формулами*, *тавтологиями* языка). Понятие логического закона является некоторым математическим уточнением идеи формулы, истинной «лишь в силу своей формы, независимо от содержания».

Далее, мы говорим, что формула A логически эквивалентна формуле B , и пишем $A \sim B$, если формула $A \equiv B$ является логическим законом. Читатель, несомненно, знаком с некоторыми логическими законами и логическими эквивалентностями, такими как выражение одних логических связей через другие, пронесение кванторов через логические связки и т. п. Логические законы позволяют преобразовывать формулы к нужному нам виду, например, всякую формулу, как известно, можно заменить логически эквивалентной формулой, имеющей уже *предваренный вид*, т. е. такой, у которой все кванторы находятся впереди формулы, а далее идет бескванторная формула.

Для каждого логико-математического языка Ω определяется *исчисление предикатов* в языке Ω . При описании исчисления предикатов задаются его аксиомы и правила вывода, при этом в различных учебниках можно найти несколько отличающиеся друг от друга формулировки (эти отличия, конечно, несущественны для развиваемой далее теории). Мы остановимся на следующей формулировке. *Аксиомами исчисления предикатов* в языке Ω называются формулы этого языка, имеющие один из следующих видов:

- 1) $A \supset B \supset A$;
- 2) $(A \supset B \supset C) \supset (A \supset B) \supset (A \supset C)$;
- 3) $A \supset B \supset A \wedge B$;
- 4) $A \wedge B \supset A$;
- 5) $A \wedge B \supset B$;
- 6) $(A \supset C) \supset (B \supset C) \supset (A \vee B \supset C)$;
- 7) $A \supset A \vee B$;
- 8) $B \supset A \vee B$;
- 9) $(A \supset B) \supset (A \supset \neg B) \supset \neg A$;
- 10) $\neg \neg A \supset A$;
- 11) $\forall x A \supset A(x|t)$;
- 12) $\forall x (C \supset A(x)) \supset C \supset \forall x A(x)$;
- 13) $A(x|t) \supset \exists x A$;
- 14) $\forall x (A(x) \supset C) \supset \exists x A(x) \supset C$.

Здесь A, B, C — произвольные формулы Ω . В схемах 12), 14) формула C не содержит свободно переменной x . Нетрудно убедиться, что все аксиомы исчисления предикатов суть логические законы.

Правилами вывода исчисления предикатов являются фигуры следующих видов:

$$\frac{A, A \supset B}{B}, \frac{A}{\forall x A}.$$

Здесь A, B — произвольные формулы Ω , x — произвольная переменная. Первое правило носит традиционное латинское название — *модус поненс*, второе правило называется *правилом обобщения*.

Дерево формул есть по определению двумерная фигура, составленная из формул языка по следующим правилам:

- 1) каждая формула A сама является деревом формул, нижней формулой этого дерева называется сама формула A ;
- 2) если D_1 и D_2 суть деревья формул с нижними формулами вида A и $A \supset B$ соответственно, то фигура

$$\frac{D_1, D_2}{B}$$

есть дерево формул; мы говорим, что формула B получена в этом дереве из A и $A \supset B$ по правилу модус поненс; нижней формулой результирующего дерева формул является по определению формула B ;

- 3) если D_1 — дерево формул с нижней формулой A и x — переменная, то формула

$$\frac{D_1}{\forall x A}$$

есть также дерево формул; мы говорим, что нижняя формула $\forall x A$ этого дерева формул получена из A по правилу обобщения.

Коротко говоря, дерево формул есть конечное дерево, в вершинах которого расставлены формулы и переходы сверху вниз в котором совершаются по правилам вывода исчисления предикатов.

Верхние, начальные формулы в дереве формул, которые не имеют вида аксиом исчисления предикатов, называются *гипотезами*, или *открытыми посылками* рассматриваемого дерева формул. Ветвью дерева формул D мы назовем последовательность A_1, \dots, A_n вхождений формул в D такую, что для всякого $i < n$ формула A_{i+1} находится в D непосредственно выше формулы A_i и A_1 — нижняя формула D , а A_n — верхняя формула D . Пусть A и B — различные вхождения формул в D . Мы говорим, что формула B расположена выше формулы A в D , если существует ветвь A_1, \dots, A_n дерева формул D такая, что A есть A_i и B есть A_j , причем $j > i$.

Деревом вывода, или просто *выводом*, в исчислении предикатов назовем дерево формул, удовлетворяющее следующему *структурному требованию*: если формула вида $\forall x A$ получена в выводе из формулы A по правилу обобщения, то переменная x не входит свободно в гипотезы, расположенные выше рассматриваемого вхождения формулы $\forall x A$.

Если формула вида $\forall x A$ получена по правилу обобщения из формулы A и формула B находится выше рассматриваемого вхождения $\forall x A$ и содержит свободно x , то говорят, что переменная x *варьируется* в формуле B . Наше структурное требование означает, таким образом, что в выводе параметры гипотез не *варьируются*, остаются *фиксированными*.

Если Γ — конечный список формул и A — формула языка Ω , то будем говорить, что формула A *выводима* в исчислении предикатов из списка формул Γ , и писать $\Gamma \vdash A$, если может быть построен вывод D с нижней формулой A такой, что всякая гипотеза D является членом списка Γ . Разумеется, некоторые формулы Γ могут при этом и не быть гипотезами D . Мы говорим, что вывод D формулы A не зависит от таких членов Γ .

Если список Γ пуст, то $\Gamma \vdash A$ означает, что существует вывод формулы A вообще без гипотез. В этом случае мы говорим, что формула A *выводима* в исчислении предикатов, и пишем $\vdash A$.

Саму фигуру $\Gamma \vdash A$ назовем *выводимостью*, или *секвенцией*. Таким образом, чтобы установить секвенцию $\Gamma \vdash A$, следует построить вывод в исчислении предикатов с нижней формулой A , все гипотезы которого находятся среди членов Γ .

Теорема (о дедукции). $\Gamma, A \vdash B$ тогда и только тогда, когда $\Gamma \vdash A \supset B$.

Если $\vdash A$, то A является логическим законом. Таким образом, на исчисление предикатов можно смотреть как на некоторый формальный механизм для получения логических законов.

Для вывода формул в исчислении предикатов используется специально подобранная удобная система производных правил, называемая *техникой естественного вывода*. Читатель может познакомиться с этими правилами по нашей книге [1] или, например, по книге [3]. Мы будем предполагать некоторое умение выводить формулы в исчислении предикатов и иногда будем опускать доказательства того, что та или иная формула выводима. Начинаящему при этом полезно по крайней мере убеждаться, что рассматриваемая формула является логическим законом.

Формальная аксиоматическая теория T задается двумя объектами: логико-математическим языком Ω и множеством Y предложений этого языка. Ω называется языком теории T , а элементы множества Y называются *нелогическими аксиомами T* . Подчеркнем, что все нелогические аксиомы суть предложения, т. е. замкнутые формулы языка Ω .

Мы говорим, что формула A языка Ω *выводима в теории T* , и пишем $T \vdash A$, если существует конечный список Γ членов Y такой, что $\Gamma \vdash A$ в исчислении предикатов.

Интерпретация M для языка Ω называется *моделью теории T* , если $M \models A$ для всякого предложения $A \in Y$, т. е. если всякая нелогическая аксиома теории T истинна в M . Если M — модель теории T и $T \vdash B$, где B — предложение, то $M \models B$. Таким образом, выводимое в T предложение оказывается истинным во всякой модели теории T .

Опишем кратко некоторые формальные аксиоматические теории.

Элементарная арифметика Ag .

Язык этой теории содержит один сорт переменных x, y, z, \dots ,

константу 0, один одноместный функциональный символ Sx и два двухместных функциональных символа $x+y$, $x \cdot y$. Атомарные формулы Ag имеют вид $(t=r)$, где t, r — произвольные термы языка.

Нелогические аксиомы Ag делятся на три группы: аксиомы равенства, аксиомы Пеано, определяющие аксиомы для сложения и умножения. При формулировке нелогических аксиом ниже мы будем опускать кванторы общности спереди. Предполагается, конечно, что каждая аксиома является предложением.

Аксиомы равенства:

- 1) $x=x$;
- 2) $x=y \wedge x=z \supset y=z$.

Аксиомы Пеано:

- 3) $Sx \neq 0$;
- 4) $(Sx = Sy) \equiv x = y$;
- 5) $A(0) \wedge \forall x (A(x) \supset A(Sx)) \supset \forall x A(x)$ (принцип полной математической индукции).

Определяющие аксиомы для сложения и умножения:

- 6) $x+0=x$;
- 7) $x+Sy=S(x+y)$;
- 8) $x \cdot 0=0$;
- 9) $x \cdot Sy=x \cdot y+x$.

Аксиоматика теории Ag подобрана таким образом, чтобы все обычные факты, выражимые в языке теории, были бы в ней выводимы. Стандартной моделью теории Ag является модель ω . В этой модели переменные рассматриваются как пробегающие множество всех натуральных чисел; это множество мы также обозначаем через ω , константа изображает натуральное число нуль, функциональные символы интерпретируются очевидным образом. Как мы увидим далее, существуют и другие модели теории Ag , не изоморфные ω . В то же время теория Ag является неполной: можно указать предложение, истинное в модели ω , но не выводимое в Ag ; мы докажем это в гл. III, § 1.

Арифметика второго порядка $Ag2$.

Язык этой теории содержит два сорта переменных. Переменные x, y, z, \dots рассматриваются как пробегающие натуральные числа, переменные X, Y, Z, \dots пробегают произвольные подмножества множества всех натуральных чисел. Константы и функциональные символы $Ag2$ те же, что и в языке Ag . Таким образом, язык $Ag2$ содержит термы двух сортов: термы для натуральных чисел и термы для множеств натуральных чисел, причем термами для множеств натуральных чисел являются лишь переменные X, Y, Z, \dots . Атомарные формулы $Ag2$ имеют один из следующих двух видов: формальные равенства $(t=r)$, где t и r — термы для натуральных чисел, и $(t \in X)$, где t — терм для натуральных чисел и X — переменная для множеств натуральных чисел. Сложные формулы строятся

обычным образом с помощью логических связок и кванторов, причем кванторы используются по обоим сортам переменных.

Нелогические аксиомы Ag_2 имеют тот же вид, что и аксиомы Ag , но в схеме индукции 5) в качестве формулы $A(x)$ можно использовать произвольную формулу языка Ag_2 . Кроме того, добавляется новая схема аксиом, *аксиома свертывания*:

10) $\exists X \forall x (x \in X \equiv A(x))$,
где $A(x)$ — произвольная формула Ag_2 , не содержащая свободно переменной X .

Стандартной моделью Ag_2 является модель, которую мы, так же как и в случае Ag , будем называть моделью ω . В этой модели переменные x, y, z, \dots пробегают множество ω всех натуральных чисел, а переменные X, Y, Z, \dots пробегают множество P_ω всех подмножеств натуральных чисел. Эта теория также имеет много других интересных моделей, неизоморфных ω .

Исчисление предикатов в языке Ω также можно рассматривать как формальную аксиоматическую теорию в языке Ω с пустым множеством нелогических аксиом. Всякая интерпретация языка Ω является моделью этой теории.

Чистое исчисление предикатов есть по определению исчисление предикатов в специальном языке с одним сортом переменных, без констант и функциональных символов, со счетным набором предикатных символов для каждого числа аргументных мест.

§ 1. ЯЗЫК НАИВНОЙ ТЕОРИИ МНОЖЕСТВ. ПАРАДОКСЫ НАИВНОЙ ТЕОРИИ МНОЖЕСТВ

1. Теперь мы перейдем к изучению некоторого конкретного языка M^+ , предназначенного для описания свойств множеств. Мы исходим из гипотезы, что имеется некоторая математическая структура — семейство множеств. Для каждого двух множеств a и b определено, когда $a \in b$ — a принадлежит b , а когда это неверно. Каковы дальнейшие свойства этой структуры \mathcal{M} и в какой мере законно ее рассмотрение — мы обсудим позднее.

Язык M^+ содержит один сорт переменных x, y, z, \dots , которые рассматриваются как пробегающие множества — элементы структуры \mathcal{M} .

Понятие формулы и термина языка M^+ формулируется в виде одновременного индуктивного определения. Пункт 1) — базис этого определения, а 2)–5) — индуктивные шаги.

- 1) переменная есть терм;
- 2) если t и r суть термы, то $(t \in r)$ есть формула;
- 3) если φ и ψ суть формулы, то

$$(\varphi \wedge \psi), (\varphi \vee \psi), (\varphi \supset \psi), \neg \varphi$$

также суть формулы;

- 4) если x — переменная, а φ — формула, то $\forall x\varphi, \exists x\varphi$ суть формулы;

- 5) если x — переменная, φ — формула, то $\{x|\varphi\}$ есть терм.

Определение закончено.

Формула M^+ может быть, например, такой

$$\{x|x \in y \supset x \in z\} \in \{y|\neg y \in y\}.$$

Терм $\{x|\varphi\}$ читается так: «множество всех x , для которых $\varphi(x)$ ». Мы говорим, что этот терм определяет множество *свертыванием*, а сам терм называем *сверткой* по формуле φ . Формула φ может быть совершенно произвольной формулой языка. Говорят, что в языке M^+ имеется *неограниченное свертывание*.

Заметим, что язык M^+ не является языком первого порядка, именно в силу пункта 5) определения, позволяющего определять новые термы с помощью формул. Такого эффекта нет в языках первого порядка, где термы определяются отдельно.

Следует иметь в виду, что в терме $\{x|\varphi\}$ переменная x — связанная, так что $\{x|\}$ играет роль квантора. В языке M^+ термы могут содержать связанные переменные, так что теория переименования связанных переменных, коллизии переменных и т. п. должна теперь относиться и к термам.

2. Введем сокращенные обозначения:

$$\begin{array}{l} x \text{ включено в } y \\ x \subseteq y \Leftrightarrow \forall z (z \in x \supset z \in y); \end{array}$$

$$\begin{array}{l} x \text{ равно } y \\ x = y \Leftrightarrow (x \subseteq y) \wedge (y \subseteq x); \end{array}$$

$$\begin{array}{l} x \text{ не равно } y \\ x \neq y \Leftrightarrow \neg (x = y); \end{array}$$

$$\begin{array}{l} x \text{ не принадлежит } y \\ x \notin y \Leftrightarrow \neg (x \in y); \end{array}$$

$$\begin{array}{l} x \text{ строго включено в } y \\ x \subset y \Leftrightarrow (x \subseteq y) \wedge (x \neq y). \end{array}$$

Разумеется, эти обозначения можно использовать не только с переменными x и y , но и с другими термами. При этом $(t \subseteq r)$ следует трактовать, конечно, как $(x \subseteq y) (x, y \parallel t, r)$. Аналогичное замечание относится и к дальнейшим обозначениям.

Пользуясь законами логики, можно установить, что в \mathcal{M} имеют место некоторые факты. Например, в \mathcal{M} при любой оценке $x \subseteq x$, т. е. $x \subseteq x$ есть закон теории множеств. В самом деле, $x \subseteq x$ имеет вид $\forall z (z \in x \supset z \in x)$ и вытекает из логического закона тождества $z \in x \supset z \in x$.

Мы будем писать

$$. x \subseteq x,$$

где точка слева заменяет словесный оборот «следующая формула является законом теории множеств», т. е. истинна в \mathcal{M} при любой оценке свободных переменных.

Упражнение. Докажите:

$$. x = x;$$

$$. x = y \supset y = x;$$

$$. x = y \wedge y = z \supset x = z.$$

3. Так как структура \mathcal{M} должна отражать интуитивные свойства множеств, мы считаем, что в ней законами являются следующие два вида формул.

1) Аксиома объемности (экстенциональности)

$$. x = y \wedge x \in z \supset y \in z;$$

2) Аксиома свертывания

$$. z \in \{y | \varphi(y)\} \Leftrightarrow \varphi(z).$$

Здесь $\varphi(z)$ есть, разумеется, $\varphi(y \parallel z)$.

Аксиома объемности имеет следствием, что определенное нами равенство $x=y$ действительно обладает свойствами равенства. А именно выполняются свойства замены равного на равное:

$$.x=y \supset (\varphi(x) \equiv \varphi(y));$$

$$.x=y \supset (t(x) = t(y)).$$

Нетрудно доказать эти два закона, исходя из аксиомы объемности, индукцией по определению п. 1, но мы не будем этим заниматься и свободно используем эти два последних закона в дальнейшем.

Аксиома свертывания описывает свойства выражения $\{x| \varphi\}$.

4. Введем понятие *неупорядоченной пары*:

$$\{x, y\} \equiv \{z | z=x \vee z=y\}.$$

Основные свойства этого выражения:

$$1) .u \in \{x, y\} \equiv (u=x \vee u=y);$$

$$2) .x \in \{x, y\};$$

$$3) .y \in \{x, y\};$$

$$4) .\{x, y\} = \{y, x\}.$$

▷ Первое следует из аксиомы свертывания. Докажем второе. Из первого утверждения, в частности,

$$.x \in \{x, y\} \equiv (x=x \vee x=y),$$

но

$$.x=x \vee x=y,$$

так как $.x=x$ (см. п. 2). Следовательно, верна правая часть эквивалентности, а значит, и левая.

Докажем четвертое утверждение. Необходимо показать, что для всякого z

$$z \in \{x, y\} \equiv z \in \{y, x\}.$$

Но ввиду первого свойства

$$z \in \{x, y\} \equiv (z=x \vee z=y);$$

$$z \in \{y, x\} \equiv (z=y \vee z=x),$$

и правые части логически эквивалентны. □

5. Понятие *одноэлементного множества* (синглтона):

$$\{x\} \equiv \{z | z=x\}.$$

Основные свойства:

$$1) .u \in \{x\} \equiv (u=x);$$

$$2) .x \in \{x\}.$$

Доказательство предоставляется читателю в качестве упражнения.

Лемма. 1) $\{x\} = \{y\} \equiv (x = y)$;

2) $\{x, y\} = \{u, v\} \equiv ((x = u \wedge y = v) \vee (x = v \wedge y = u))$;

3) $\{x, y\} = \{u\} \equiv (x = u \wedge y = u)$.

▷ Докажем 2). Импликация справа налево вытекает просто из общих законов равенства п. 3. Допустим $\{x, y\} = \{u, v\}$. По определению равенства это означает

$$\forall z (z \in \{x, y\} \equiv z \in \{u, v\}).$$

Имеем $x \in \{x, y\}$ и, значит, $x \in \{u, v\}$. Отсюда (п. 4) $x = u \vee x = v$. Разберем два случая и покажем, что в каждом из них верно

$$(x = u \wedge y = v) \vee (y = u \wedge x = v).$$

1) Пусть $x = u$.

Ввиду $v \in \{u, v\}$ имеем $v \in \{x, y\}$, т. е. $v = x \vee v = y$. Если $v = y$, то $(x = u \wedge y = v)$, что и требовалось. Если же $v = x$, то ввиду $x = u$ имеем $u = v$. Далее $y \in \{x, y\}$ и, значит, $y \in \{u, v\}$, т. е. $y = u \vee y = v$. Ввиду $u = v$ всегда $y = v$. Таким образом, вновь имеем $(x = u \wedge y = v)$.

2) Пусть $x = v$. Рассматривается симметрично. \square

6. Понятие *пустого множества*:

$$\emptyset \equiv \{x \mid x \neq x\}.$$

Основные свойства:

1) $\forall z (z \notin \emptyset)$;

2) $\emptyset \neq \{\emptyset\}$;

3) $\{\emptyset\} \neq \{\emptyset, \{\emptyset\}\}$.

▷ 1) $z \in \emptyset \equiv (z \neq z)$, но правая часть ложна в силу п. 2. Значит, $z \notin \emptyset$.

2) Имеем $\emptyset \notin \emptyset$ и $\emptyset \in \{\emptyset\}$, значит,

$$\neg \forall z (z \in \emptyset \equiv z \in \{\emptyset\}),$$

т. е. $\emptyset \neq \{\emptyset\}$. \square

Упражнение. Докажите $\emptyset \subseteq u$ и, в частности, $\emptyset \subseteq \emptyset$.

Приведите примеры конкретных множеств A, B, C, D (множества описываются замкнутыми термами M^+) такие, что

$$A \subseteq B, \text{ но } A \not\subseteq B \text{ и}$$

$$C \subseteq D, \neg (C \subseteq D).$$

7. Пересечение, объединение и разность двух множеств

$$x \cap y \equiv \{z \mid z \in x \wedge z \in y\};$$

$$x \cup y \equiv \{z \mid z \in x \vee z \in y\};$$

$$x \setminus y \equiv \{z \mid z \in x \wedge z \notin y\};$$

1) $u \in x \cap y \equiv (u \in x \wedge u \in y)$;

2) $u \in x \cup y \equiv (u \in x \vee u \in y)$;

3) $u \in x \setminus y \equiv (u \in x \wedge u \notin y)$.

▷ Эти утверждения вытекают непосредственно из аксиомы свертывания. □

Упражнение. Докажите:

$$1) . z \setminus (x \cap y) = (z \setminus x) \cup (z \setminus y);$$

$$2) . (x \cup y) \cap z = (x \cap z) \cup (y \cap z).$$

8. Определим теперь *объединение* и *пересечение* семейства множеств.

$$\cup x \Leftrightarrow \{z \mid \exists u (u \in x \wedge z \in u)\};$$

$$\cap x \Leftrightarrow \{z \mid \forall u (u \in x \supset z \in u)\}.$$

Для формулировки свойств этих термов введем полезное *общее* определение *ограниченных кванторов*

$$(\forall x \in y) \varphi(x) \Leftrightarrow \forall x (x \in y \supset \varphi(x));$$

$$(\exists x \in y) \varphi(x) \Leftrightarrow \exists x (x \in y \wedge \varphi(x)).$$

Первое читается как «для всех x , принадлежащих y , имеет место $\varphi(x)$ », а второе — как «существует x , принадлежащее y , такое, что имеет место $\varphi(x)$ ».

Обратите внимание, что для общности используется импликация, а для существования — конъюнкция. При таком определении законы де Моргана верны и для ограниченных кванторов.

Точнее,

$$. \neg (\forall x \in y) \varphi(x) \equiv (\exists x \in y) \neg \varphi(x);$$

$$. \neg (\exists x \in y) \varphi(x) \equiv (\forall x \in y) \neg \varphi(x).$$

В самом деле, $\neg (\forall x \in y) \varphi(x)$ означает $\neg \forall x (x \in y \supset \varphi(x))$ и по логическим законам эквивалентно $\exists x \neg (x \in y \supset \varphi(x))$ и, далее, $\exists x (x \in y \wedge \neg \varphi(x))$.

Подобным образом можно определить

$$(\forall x \subseteq y) \varphi(x) \Leftrightarrow \forall x (x \subseteq y \supset \varphi(x));$$

$$(\exists x \subseteq y) \varphi(x) \Leftrightarrow \exists x (x \subseteq y \wedge \varphi(x))$$

с выполнением законов

$$. \neg (\forall x \subseteq y) \varphi(x) \equiv (\exists x \subseteq y) \neg \varphi(x);$$

$$. \neg (\exists x \subseteq y) \varphi(x) \equiv (\forall x \subseteq y) \neg \varphi(x).$$

В языке арифметики можно определить

$$(\forall x \leq y) A(x) \Leftrightarrow \forall x (x \leq y \supset A(x));$$

$$(\exists x \leq y) A(x) \Leftrightarrow \exists x (x \leq y \wedge A(x)).$$

И вновь будем иметь, например,

$$\omega \models \neg (\exists x \leq y) A(x) \equiv (\forall x \leq y) \neg A(x).$$

Заметьте, что знак неравенства в законах де Моргана «не переворачивается»! Студент иногда склонен ошибочно утверждать

$$\omega \models \neg (\exists x \leq y) A(x) \equiv (\forall x > y) \neg A(x)?$$

Основные свойства объединения и пересечения:

$$1) . z \in \bigcup x \equiv (\exists u \in x) (z \in u);$$

$$2) . z \in \bigcap x \equiv (\forall u \in x) (z \in u).$$

▷ Эти свойства имеют место по аксиоме свертывания и в силу определения. □

9. Универсальное множество (универсум, множество всех множеств) определяется как

$$V = \{z \mid z = z\}.$$

Имеем: $\forall z (z \in V)$.

▷ По определению $z \in V \equiv (z = z)$, но правая часть верна (см. п. 2). □

Общее дополнение определяется следующим образом:

$$\bar{x} \equiv V \setminus x.$$

Упражнение.

$$\overline{x \cup y} = \bar{x} \cap \bar{y};$$

$$x \setminus y = x \cap \bar{y}.$$

Л е м м а.

$$\bigcup \emptyset = \emptyset;$$

$$\bigcap \emptyset = V.$$

▷ $z \in \bigcup \emptyset \equiv (\exists u \in \emptyset) (z \in u)$ и правая часть ложна, так как $\forall u (u \neq \emptyset)$. Следовательно, $z \notin \bigcup \emptyset$ для всякого z и, значит, $\bigcup \emptyset = \emptyset$. $z \in \bigcap \emptyset \equiv (\forall u \in \emptyset) (z \in u)$ и правая часть истинна (в силу ложности посылки импликации). Следовательно, $z \in \bigcap \emptyset$ для всякого z и, значит, $\bigcap \emptyset = V$. □

10. Множество всех подмножеств (множество — степень)

$$Px \equiv \{z \mid z \subseteq x\};$$

$$. u \in Px \equiv (u \subseteq x).$$

11. Дадим определение некоторого стандартного бесконечного множества.

Введем теоретико-множественную операцию следования:

$$Sx \equiv x \cup \{x\}.$$

Назовем множество x прогрессивным, если оно содержит \emptyset и замкнуто относительно S . Формально

$$\text{Prog}(x) \equiv (\emptyset \in x) \wedge \forall z (z \in x \supset Sz \in x).$$

Теперь в качестве стандартного бесконечного множества возьмем пересечение семейства всех прогрессивных множеств:

$$\omega \equiv \bigcap \{x \mid \text{Prog}(x)\}.$$

Интуитивно ω состоит из последовательности элементов:

$$\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}, \dots$$

Каждый член этой последовательности получается операцией S из предыдущего члена. Все члены различны, так что ω бесконечно. Член последовательности с номером n (если считать с нуля) является n -элементным множеством. Эти интуитивные идеи можно подтвердить в виде точных фактов:

- 1) $\emptyset \in \omega$;
- 2) $z \in \omega \supset Sz \in \omega$;
- 3) $\text{Prog}(x) \supset (\omega \subseteq x)$.

▷ По определению $u \in \omega \equiv \forall x (\text{Prog}(x) \supset u \in x)$. Если сюда вместо u подставить \emptyset , то правая часть верна (по определению $\text{Prog}(x)$). Следовательно, $\emptyset \in \omega$. Далее, если $z \in \omega$, то $z \in x$ для всех x , $\text{Prog}(x)$. Но тогда и $Sz \in x$ для всех x , $\text{Prog}(x)$, т. е. $Sz \in \omega$. Наконец, докажем 3). Пусть $\text{Prog}(x)$, необходимо установить $\omega \subseteq x$, т. е. $z \in \omega \supset z \in x$ для всех z . Но если $z \in \omega$, то, конечно, $z \in x$ ввиду $\text{Prog}(x)$. \square

12. Введем теперь два специальных обозначения для некоторых термов вида свертки.

Определение множества выделением:

$$\{x \in y \mid \varphi(x)\} \equiv \{x \mid x \in y \wedge \varphi(x)\}.$$

Выражение слева читается как «множество всех x , принадлежащих y , такое, что $\varphi(x)$ ». Здесь переменная x — связанная, а y — свободная. Мы считаем, что x отлично от y . На общих основаниях вместо y можно, разумеется, правильно подставлять термы (переименовывая связанное x , если нужно).

Основное свойство этого обозначения:

$$u \in \{x \in y \mid \varphi(x)\} \equiv (u \in y \wedge \varphi(u)).$$

Определение множества подстановкой:

$$\{y \mid x \in u, (x \mapsto y), \varphi(x, y)\} \equiv \\ \{y \mid (\exists x \in u) (\varphi(x, y) \wedge \forall z (\varphi(x, z) \supset y = z))\}.$$

Здесь x, y, u — различные переменные, причем x и y — связанные, а u — свободная.

Таким образом, слева обозначено некоторое множество Q такое, что $y \in Q \Leftrightarrow$ найдется $x \in u$ такое, что для этого x верно $\varphi(x, y)$, причем указанное y единственно. Иными словами, будем просматривать различные $x \in u$. Для некоторых $x \in u$ не найдется вовсе y такого, что $\varphi(x, y)$. Для некоторых $x \in u$ существует много y таких, что $\varphi(x, y)$. А вот для некоторых $x \in u$ существует ровно одно y такое, что $\varphi(x, y)$. В точности все такие y мы и зачисляем в множество Q .

Эти интуитивные идеи можно оформить в виде точных фактов.

Пусть далее Q обозначает терм

$$\{y \mid x \in u, (x \mapsto y), \varphi(x, y)\}, \\ z \in Q \equiv (\exists x \in u) (\varphi(x, z) \wedge \forall v (\varphi(x, v) \supset z = v))$$

▷ Утверждение следует непосредственно из определения \square

Введем сокращение (читается «существует и только одно x такое, что $\varphi(x)$ »):

$$\exists ! x \varphi(x) \Leftrightarrow \exists x \varphi(x) \wedge \forall yz (\varphi(y) \wedge \varphi(z) \supset y = z).$$

Упражнение. $x \in u \wedge \exists ! v \varphi(x, v) \wedge \varphi(x, y) \supset y \in Q$.

13. Рассмотрим множество Рассела

$$R \Rightarrow \{x | x \notin x\}.$$

Это — конкретное множество, описанное замкнутым термом языка M^+ .

По аксиоме свертывания для любого u имеем

$$. u \in R \equiv u \notin u.$$

В частности, подставляя вместо u множество R , получим

$$. R \in R \equiv R \notin R.$$

С другой стороны, очевидно, имеем $\neg (R \in R \equiv R \notin R)$.

Мы пришли к противоречию.

Это короткое рассуждение и является *парадоксом Рассела* в нашем языке.

Обсудим кратко природу парадоксов, к которым относится и парадокс Рассела.

Рассмотрим следующее высказывание:

«Высказывание, написанное в этой строке, ложно».

Истинно или ложно высказывание, написанное в кавычках? Исходя из его смысла, можно заключить, что оно истинно тогда и только тогда, когда оно ложно, и мы приходим к противоречию.

Причину парадокса можно усматривать в структуре высказывания, написанного в кавычках; оно ссылается само на себя. Здесь проявляется *абстракция отчуждения*, в силу которой исследователь сам процесс своего исследования, свои мысли, делает объектом исследования. Мы видим, что если это делать неосторожно, то получают внутренне противоречивые высказывания.

В несколько иной форме этот парадокс известен как парадокс Эвбулида (четвертый век до н. э.). Пусть некто говорит. «Я лгу». Истинно или ложно это его высказывание?

Конечно, можно устранить парадоксы, считая приведенные высказывания неосмысленными (или неправильно построенными). Но возникает трудная проблема, а какие высказывания осмыслены? Не окажется ли, что, казалось бы, надежные утверждения в математике или физике внутренне противоречивы? Каковы критерии, отличающие осмысленные высказывания от неосмысленных?

Внимательный анализ показывает, что в теории множеств

возникает аналогичная ситуация при использовании аксиомы свертывания (п. 3). Сначала исследователь формулирует свойство $\varphi(y)$, а затем образует объект исследования — множество $\{y|\varphi(y)\}$. Свойство $\varphi(y)$ может сообщать нечто обо всех множествах, в том числе и о вновь образуемом $\{y|\varphi(y)\}$. Так происходит ссылка на себя. Например, $\varphi(y)$ может содержать кванторы по переменным, которые, естественно, мыслятся как пробегающие все множества, в том числе и множество $\{y|\varphi(y)\}$. В определенном смысле множество $\{y|\varphi(y)\}$ нельзя считать вновь образованным: о нем уже идет речь в формулировке свойства $\varphi(y)$! Это явление называется *непредикативностью* теории множеств.

Полный отказ от непредикативных определений требует значительной перестройки существующей математики, и поэтому обычно используется компромиссный подход. В необходимых случаях используется непредикативная аксиома свертывания, но ее применение ограничивается, чтобы избежать возникновения парадоксов.

Рассмотрим теперь еще одну парадоксальную ситуацию. Деревенский парикмахер бреет тех и только тех в своей деревне, кто сам не бреется. Бреет ли он сам себя? Несложное рассуждение показывает, что он бреет сам себя тогда и только тогда, когда не бреет сам себя.

Как следует реагировать на такую ситуацию? Очень просто. Такого парикмахера просто не существует. Условие, которому должен подчиняться наш гипотетический парикмахер, внутренне противоречиво, его нельзя выполнить (хотя это сразу и незаметно). Конечно, не следует полагать, что тем самым снимаются все вопросы, которые вызывает вышеприведенный очень тонкий пример. Главный вопрос — какие же условия в таком случае внутренне противоречивы, а какие — нет? Можно ли уверенно выделить широкий класс заведомо непротиворечивых условий?

Аналогично естественно считать, что множество R не существует. Однако в таком случае необходимо подвергнуть пересмотру наши представления о структуре \mathcal{M} теории множеств и изменить строение языка M^+ . До сих пор мы полагали, что *всякий замкнутый терм M^+ определяет множество*, т. е. мы считали, что любое условие $\varphi(x)$ определяет объект $\{x|\varphi(x)\}$. Оказалось, однако, что такая точка зрения внутренне противоречива. Нужно выделить класс условий, которые заведомо определяют множества, и таких множеств должно быть достаточно для обслуживания обычных математических рассуждений. В то же время следует признать, что некоторые условия множеств не определяют.

Современная математическая логика еще очень далека от полного решения этой задачи. Имеется несколько практически удобных решений. Из них самым популярным, по-видимому, является подход, предложенный Цермело и усовершенствованный

Френкелем. Основная его идея состоит в том, что мы отказываемся от «слишком обширных» множеств неопределенной мощности, таких, как например, универсальное множество.

§ 2. ЯЗЫК ТЕОРИИ МНОЖЕСТВ ЦЕРМЕЛО — ФРЕНКЕЛЯ

1. Мы вновь исходим из гипотезы, что имеется математическая структура \mathcal{M}_0 — семейство множеств с отношением $a \in b$. Ее свойства определяются *семантическими соглашениями*, которые мы (несколько неточно) будем называть аксиомами теории множеств Цермело — Френкеля.

Язык ZF^+ содержит один сорт переменных x, y, z, \dots , которые рассматриваются как пробегающие множества.

Понятие *формулы* и *терма* ZF^+ определяется одновременной индукцией:

- 1) переменная есть терм;
- 2) если t, r — термы, то $(t \in r)$ есть формула;
- 3) если φ и ψ суть формулы, то

$$(\varphi \wedge \psi), (\varphi \vee \psi), (\varphi \supset \psi), \neg \varphi$$

также суть формулы;

4) если x — переменная, а φ — формула, то $\forall x \varphi, \exists x \varphi$ суть формулы;

5) символы \emptyset и ω суть термы;

6) если t, r — термы, то $\{t, r\}$ есть терм;

7) если t — терм, то Pt и $\bigcup t$ суть термы;

8) если φ — формула, t — терм, не содержащий свободно переменной x , то $\{x \in t \mid \varphi\}$ есть терм;

9) если φ — формула, x, y — различные переменные и t — терм, не содержащий свободно переменных x и y , то

$$\{y \mid x \in t, (x \mapsto y), \varphi\}$$

есть терм.

Определение формулы и терма закончено.

Идея языка ZF^+ состоит в том, что в него мы включаем лишь часть конструкций из M^+ , избегая известных парадоксов. Теперь $\emptyset, \omega, \bigcup, P$ уже не определяются, а являются самостоятельными значками. Вместо неограниченной аксиомы свертывания мы имеем лишь ее частные случаи, в том числе определение выделением и определение подстановкой.

2. Мы считаем, что законами структуры \mathcal{M}_0 являются следующие формулы:

1) *Аксиома объемности*:

$$. x = y \wedge x \in z \supset y \in z.$$

Существенным для нас является то, что из этой аксиомы (с помощью остальных аксиом) выводятся свойства равенства:

$$. x = y \supset (\varphi(x) \equiv \varphi(y));$$

$$. x = y \supset (t(x) = t(y))$$

(ср. п. 3 § 1). Здесь $x \subseteq y$ и $x = y$ определяются, как и раньше (п. 2 § 1).

2) Аксиома пустого множества:

$$\cdot \forall z (z \notin \emptyset);$$

в M^+ это свойство доказывается, а знак \emptyset определяется особым образом. В языке ZF^+ знак \emptyset является самостоятельным, и его свойства следует специально определить. Это замечание относится и к другим конструкциям ZF^+ .

3) Аксиома пары:

$$\cdot \forall z (z \in \{x, y\} \equiv (z = x \vee z = y)).$$

Свойства пары п. 4 и п. 5 § 1 доказываются и в ZF^+ , так как все они выводятся только из п. 4 § 1, 1), а это свойство пары выполняется и в ZF^+ . Далее, определим теперь

$$\{x\} \Rightarrow \{x, x\},$$

тогда основное свойство п. 5 § 1, 1) $\{x\}$ также выполняется в ZF^+ .

4) Аксиома суммы:

$$\cdot \forall z (z \in \bigcup x \equiv (\exists u \in x) (z \in u)).$$

Основное свойство объединения п. 8 § 1, 1) тогда выполняется и в ZF^+ .

5) Аксиома множества подмножеств:

$$\cdot \forall z (z \in Px \equiv (z \subseteq x))$$

(ср. п. 10 § 1).

6) Аксиомы стандартного бесконечного множества:

$$\cdot \emptyset \in \omega;$$

$$\cdot (\forall z \in \omega) (Sz \in \omega);$$

$$\cdot \forall x (Prog(x) \supset (\omega \subseteq x))$$

(ср. п. 11 § 1).

7) Аксиома выделения:

$$\cdot \forall z (z \in \{x \in t \mid \varphi(x)\} \equiv (z \in t \wedge \varphi(z)))$$

(ср. п. 12 § 1).

8) Аксиома подстановки:

$$\begin{aligned} &\cdot \forall z (z \in \{y \mid x \in u, (x \mapsto y), \varphi(x, y)\} \equiv \\ &(\exists x \in u) (\varphi(x, z) \wedge \forall v (\varphi(x, v) \supset z = v))) \end{aligned}$$

(ср. п. 12 § 1).

На этом формулировка семантических соглашений ZF^+ заканчивается. Заметьте, что в естественном смысле ZF^+ есть часть языка наивной теории множеств M^+ .

В языке ZF^+ уже нет никакой возможности образовать универсальное множество $V = \{x \mid x = x\}$ или множество Рассела

$R = \{x | x \neq x\}$, для этого просто нет подходящих языковых средств. Мы считаем, что эти множества отсутствуют в структуре \mathcal{M}_0 .

3. Далее нам следует убедиться, что обычные множества, употребляемые в математике, выражаются в ZF^+ . Мы начнем с рассмотрения тех конструкций, которые уже встречались в § 1.

Пересечение, объединение и дополнение двух множеств можно определить в ZF^+ :

$$\begin{aligned}x \cup y &\equiv \cup \{x, y\}; \\x \cap y &\equiv \{z \in (x \cup y) | z \in x \wedge z \in y\}; \\x \setminus y &\equiv \{z \in x | z \notin y\}.\end{aligned}$$

Упражнение. Убедитесь, что свойства п. 7 § 1 этих понятий выполняются и в новом определении.

Чуть сложнее обстоит дело с определением пересечения семейства множеств. Определить $\cap x$ так, чтобы выполнялось определяющее свойство п. 8 § 1

$$z \in \cap x \equiv (\forall u \in x) (z \in u),$$

не представляется возможным, так как тогда согласно лемме п. 9 § 1 было бы $\cap \emptyset = V$. Однако в ZF^+ можно определить очень похожее понятие:

$$\cap x \equiv \{z \in \cup x | (\forall u \in x) (z \in u)\}.$$

Л е м м а. *Имеем:*

$$1) \ x \neq \emptyset \supset \forall z (z \in \cap x \equiv (\forall u \in x) (z \in u));$$

$$2) \ \cap \emptyset = \emptyset.$$

▷ Если $x \neq \emptyset$, то существует $u_0 \in x$. Тогда из $(\forall u \in x) (z \in u)$ следует $z \in u_0$ и, значит, $z \in \cup x$, так что ограничение в аксиоме выделения становится фиктивным. Далее, $\cap \emptyset = \emptyset$, так как $\cup \emptyset = \emptyset$. □

Разумеется, в ZF^+ нам придется отказаться от общего дополнения, так как тогда было бы $\bar{\emptyset} = V$. Множества V такого, что $\forall z (z \in V)$, существовать не может, так как иначе мы выделением могли бы определить и множество Рассела

$$R = \{z \in V | z \notin z\}.$$

§ 3. ОТНОШЕНИЯ И ФУНКЦИИ В ЯЗЫКЕ ТЕОРИИ МНОЖЕСТВ

1. *Упорядоченная пара* (по Куратовскому) определяется следующим образом:

$$\langle x, y \rangle \equiv \{\{x\}, \{x, y\}\}.$$

Основное свойство этого множества выражается следующей леммой:

$$\text{Л е м м а. } \langle x, y \rangle = \langle u, v \rangle \equiv (x = u \wedge y = v).$$

▷ Импликация справа налево следует из свойств равенства. Допустим $\langle x, y \rangle = \langle u, v \rangle$ и докажем $x = u \wedge y = v$.

По допущению $\{\{x\}, \{x, y\}\} = \{\{u\}, \{u, v\}\}$. Согласно лемме п. 5 § 2, 2) тогда

$$\{x\} = \{u\} \wedge \{x, y\} = \{u, v\}$$

или

$$\{x\} = \{u, v\} \wedge \{x, y\} = \{u\}.$$

Соответственно разберем два случая.

$$a) \{x\} = \{u, v\}, \{x, y\} = \{u\}.$$

Ввиду п. 5 § 2 тогда $x = y = u = v$, так что, очевидно, $x = u \wedge y = v$.

$$b) \{x\} = \{u\}, \{x, y\} = \{u, v\}.$$

Ввиду п. 5 § 2 тогда $x = u$.

Далее, имеем два случая:

$$x = u \wedge y = v \text{ или } x = v \wedge y = u.$$

В первом случае мы уже имеем искомос. Во втором же случае ввиду $x = u$ получаем $x = u = y = v$, т. е. вновь $x = u \wedge y = v$. □

Далее можно определить *упорядоченные тройки, четверки и т. п.*

$$\langle x \rangle \rightleftharpoons x;$$

$$\langle x_1, \dots, x_n, x_{n+1} \rangle \rightleftharpoons \langle x_1, \dots, x_n \rangle, x_{n+1} \rangle.$$

Из леммы будет следовать и более общее утверждение

$$\begin{aligned} \cdot \langle x_1, \dots, x_n \rangle &= \langle u_1, \dots, u_n \rangle \equiv \\ & (x_1 = u_1 \wedge \dots \wedge x_n = u_n). \end{aligned}$$

Заметим, что упорядоченная n -ка при $n > 0$ в нашем определении есть одновременно и упорядоченная пара.

2. *Отношение* есть по определению множество, каждый элемент которого есть упорядоченная пара. На нашем языке «быть отношением» записывается как

$$\text{Rel}(x) \rightleftharpoons (\forall z \in x) \exists uv (z = \langle u, v \rangle).$$

Мы говорим, что u и v находятся в отношении x , если $\langle u, v \rangle \in x$. Последнее записывается также в виде uvx .

Поле отношения x называется множество всех u таких, что u входит в x в качестве левого члена некоторой пары или в качестве правого члена некоторой пары.

Оказывается, что в нашем языке поле отношения может быть определено следующим образом:

$$\text{Fld}(x) \rightleftharpoons \bigcup \bigcup x.$$

То, что это действительно подходящее определение, видно из следующей леммы:

Л е м м а. 1) $\langle u, v \rangle \in x \supset u \in \text{Fld}(x) \wedge v \in \text{Fld}(x)$;

2) $\text{Rel}(x) \wedge u \in \text{Fld}(x) \supset$

$\exists v (\langle u, v \rangle \in x \vee \langle v, u \rangle \in x)$.

▷ 1) Имеем $\{u, v\} \in \langle u, v \rangle \in x$, что означает $\{u, v\} \in \bigcup x$. Далее, $u \in \{u, v\} \in \bigcup x$, что влечет $u \in \bigcup \bigcup x$, т. е. $u \in \text{Fld}(x)$. Симметрично $v \in \text{Fld}(x)$.

2) Пусть $\text{Rel}(x)$, $u \in \text{Fld}(x)$. Таким образом, найдется v , $u \in v \in \bigcup x$. А значит, найдется z такое, что $u \in v \in z \in x$. Так как $\text{Rel}(x)$, то найдутся $v_1, v_2, z = \langle v_1, v_2 \rangle$. Ввиду $v \in \langle v_1, v_2 \rangle$ имеем $v = \{v_1\}$ или $v = \{v_1, v_2\}$. Так как $u \in v$, то необходимо $u = v_1$ или $u = v_2$. □

Область определения отношения x есть множество всех левых членов пар из x .

$\text{dom}(x) \Leftarrow \{u \in \text{Fld}(x) \mid \exists v (\langle u, v \rangle \in x)\}$.

Это типичное определение в ZF^+ выделением.

Заметим, что ввиду леммы ограничение $u \in \text{Fld}(x)$ в определении несущественно:

$\exists v (\langle u, v \rangle \in x) \supset u \in \text{Fld}(x)$.

В обыденной математике дают более простое определение:

$\text{dom}(x) = \{u \mid \exists v (\langle u, v \rangle \in x)\}$.

Но в нашем языке последняя конструкция просто невозможна и нам приходится предложить некоторое, хотя бы и несущественное, ограничение.

Такое *фиктивное ограничение* в аксиоме выделения — особенность ZF^+ — есть плата за устранение парадоксов.

Основные свойства области определения:

1) $\langle u, v \rangle \in x \supset u \in \text{dom}(x)$;

2) $\text{Rel}(x) \wedge u \in \text{dom}(x) \supset \exists v (\langle u, v \rangle \in x)$.

▷ 2) Пусть $\text{Rel}(x)$ и $u \in \text{dom}(x)$. Тогда по аксиоме выделения непосредственно $\exists v (\langle u, v \rangle \in x)$. □

$\text{Rel}(x) \supset \text{dom}(x) \subseteq \text{Fld}(x)$.

Область значений отношения x есть множество всех правых членов пар из x .

$\text{rng}(x) \Leftarrow \{v \in \text{Fld}(x) \mid \exists u (\langle u, v \rangle \in x)\}$.

1) $\langle u, v \rangle \in x \supset v \in \text{rng}(x)$;

2) $\text{Rel}(x) \wedge v \in \text{rng}(x) \supset \exists u (\langle u, v \rangle \in x)$;

3) $\text{Rel}(x) \supset \text{rng}(x) \cup \text{dom}(x) = \text{Fld}(x)$.

▷ Для доказательства используйте лемму. □

3. *Декартово произведение двух множеств* x и y есть некоторое отношение. А именно это есть множество всех пар $\langle u, v \rangle$ таких, что $u \in x, v \in y$.

В языке M^+ декартово произведение можно было бы определить следующим образом:

$x \times y \Leftarrow \{z \mid \exists uv (z = \langle u, v \rangle \wedge u \in x \wedge v \in y)\}$.

В языке ZF^+ нам придется подобрать некоторое (фиктивное) ограничивающее условие для определения выделением. Оказывается, что можно определить

$$x \times y \Leftrightarrow \{z \in PP(x \cup y) \mid \exists uv (z = \langle u, v \rangle \wedge u \in x \wedge v \in y)\}.$$

В самом деле, покажем, что из условия справа от черты следует $z \in PP(x \cup y)$. Итак, пусть для некоторых u, v имеем $z = \langle u, v \rangle$, $u \in x$, $v \in y$.

Тогда $u \in x \cup y$ и $v \in x \cup y$. Отсюда

$$\{u\} \subseteq x \cup y, \{v\} \subseteq x \cup y, \{u, v\} \subseteq x \cup y.$$

Это дает $\{u\} \in P(x \cup y)$, $\{u, v\} \in P(x \cup y)$. В свою очередь, отсюда

$$\{\{u\}, \{u, v\}\} \subseteq P(x \cup y),$$

т. е. $\langle u, v \rangle \subseteq P(x \cup y)$ и, значит, $\langle u, v \rangle \in PP(x \cup y)$, т. е. $z \in PP(x \cup y)$.

Итак, в аксиоме выделения условие слева от черты выполняется, если выполняется условие справа от черты.

Основные свойства декартова произведения предлагается доказать в качестве упражнения:

- 1) $u \in x \wedge v \in y \supset \langle u, v \rangle \in x \times y$;
- 2) $z \in x \times y \supset \exists uv (z = \langle u, v \rangle \wedge u \in x \wedge v \in y)$.

Мы говорим, что отношение R задано на множестве z , если $R \subseteq z \times z$.

Следующая лемма указывает, откуда берется множество z .

Л е м м а. $\text{Rel}(R) \supset R \subseteq \text{dom}(R) \times \text{rng}(R)$.

Отсюда $R \subseteq z \times z$, где $z = \text{dom}(R) \cup \text{rng}(R)$.

▷ Пусть $\text{Rel}(R)$ и $y \in R$. Тогда $y = \langle u, v \rangle$ и из определений $u \in \text{dom}(R)$, $v \in \text{rng}(R)$, что ввиду 1) дает $y = \langle u, v \rangle \in \text{dom}(R) \times \text{rng}(R)$. □

4. Для каждого отношения R может быть определено обратное отношение, состоящее из пар, расположенных в обратном порядке. Оно обозначается R^{-1} , \bar{R} или $\text{Conv}(R)$. Точное определение таково

$$\text{Conv}(R) \Leftrightarrow \{z \in \text{rng}(R) \times \text{dom}(R) \mid$$

$$\exists uv (z = \langle u, v \rangle \wedge \langle v, u \rangle \in R)\}.$$

Заметим, что ограничение здесь — фиктивное.

- 1) $\text{Rel}(\text{Conv}(R))$;
- 2) $\langle u, v \rangle \in \text{Conv}(R) \equiv \langle v, u \rangle \in R$.

5. Если даны два отношения, то может быть определена их композиция

$$R \circ S \Leftrightarrow \{z \in \text{dom}(S) \times \text{rng}(R) \mid$$

$$\exists uvw (z = \langle u, w \rangle \wedge \langle u, v \rangle \in S \wedge \langle v, w \rangle \in R)\}.$$

Основные свойства композиции:

- 1) $\text{Rel}(R \circ S)$;
- 2) $\langle u, v \rangle \in S \wedge \langle v, w \rangle \in R \supset \langle u, w \rangle \in R \circ S$;
- 3) $\langle u, w \rangle \in R \circ S \supset \exists v (\langle u, v \rangle \in S \wedge \langle v, w \rangle \in R)$.

▷ Докажем 3), доказательства остальных свойств предоставим читателю.

Если $\langle u, w \rangle \in R \circ S$, то по определению $\langle u, w \rangle \in \text{dom}(S) \times \text{rng}(R)$ и, кроме того, существуют u', v', w' , $\langle u, w \rangle = \langle u', w' \rangle$. $\langle u', v' \rangle \in S$, $\langle v', w' \rangle \in R$. Ввиду п. 1 тогда $u = u'$, $w = w'$. Если определить $v = v'$, то мы видим, что $\langle u, v \rangle \in S$, $\langle v, w \rangle \in R$. □

6. Введем понятие *рефлексивного, симметричного и транзитивного отношений*.

Напомним, что $xRy \Leftrightarrow \langle x, y \rangle \in R$.

$$\text{Ref}(R) \Leftrightarrow (\forall x \in \text{Fld}(R)) (xRx);$$

$$\text{Sym}(R) \Leftrightarrow \forall xy (xRy \Rightarrow yRx);$$

$$\text{Trans}(R) \Leftrightarrow \forall xyz (xRy \wedge yRz \Rightarrow xRz).$$

Отношение называется *отношением эквивалентности*, если оно одновременно рефлексивно, симметрично и транзитивно. Точное определение:

$$\text{Eq}(R) \Leftrightarrow \text{Ref}(R) \wedge \text{Sym}(R) \wedge \text{Trans}(R).$$

Упражнение. Докажите:

$$\text{Ref}(R) \wedge \text{Ref}(R) \supset \text{Fld}(R) = \text{dom}(R) = \text{rng}(R).$$

7. *Ограничение отношения R множеством x* есть по определению множество таких пар $\langle u, v \rangle$ из R , что $u \in x$. Точное определение: $R \upharpoonright x \Leftrightarrow R \cap (x \times \text{rng}(R))$.

$$\langle u, v \rangle \in (R \upharpoonright x) \Leftrightarrow (\langle u, v \rangle \in R \wedge u \in x).$$

Образ множества x по отношению к R — это множество всех тех v , для которых найдется u , принадлежащее x и $\langle u, v \rangle \in R$.

$$R''x \Leftrightarrow R[x] \Leftrightarrow \{v \in \text{rng}(R) \mid (\exists u \in x) (\langle u, v \rangle \in R)\}.$$

Упражнение. Докажите:

$$R''x = \text{rng}(R \upharpoonright x).$$

Симметричное понятие — *прообраз множества x по отношению к R* — это множество всех тех u , для которых найдется v , принадлежащее x , и такое, что $\langle u, v \rangle \in R$. Точное определение:

$$R_{-1}x \Leftrightarrow \{u \in \text{dom}(R) \mid (\exists v \in x) (\langle u, v \rangle \in R)\}.$$

Упражнение.

$$\text{Ref}(R) \supset R_{-1}x = (R^{-1})''x.$$

Подобным образом можно было бы развивать и теорию *многочестных отношений*. Например, *трехместное отношение* есть по определению множество, все элементы которого суть упорядоченные тройки.

8. *Функция* (от одной переменной) есть по определению отношение, которое удовлетворяет следующему условию: если

$\langle u, v_1 \rangle \in R$ и $\langle u, v_2 \rangle \in R$, то $v_1 = v_2$, т. е. для каждого u из области определения R существует только одно v , находящееся с этим u в отношении R . Точное определение:

$$\text{Fnc}(R) \equiv \text{Rel}(R) \wedge \forall uv_1v_2 (uRv_1 \wedge uRv_2 \supset v_1 = v_2).$$

Второй конъюнктивный член здесь называют иногда *условием униформности* (по второй координате). Таким образом, функция есть отношение, униформное по второй координате.

Мы говорим, что функция f отображает множество X в множество Y , и пишем

$$f: X \rightarrow Y,$$

если область определения f есть X , а область значений f включена в Y :

$$f: X \rightarrow Y \equiv \text{Fnc}(f) \wedge \text{dom}(f) = X \wedge \text{rng}(f) \subseteq Y.$$

Мы говорим, что функция f переводит x в y , если $\langle x, y \rangle \in f$:

$$f: x \mapsto y \equiv \text{Fnc}(f) \wedge \langle x, y \rangle \in f.$$

Если $x \in \text{dom}(f)$ и f — функция, то существует и только одно y такое что $\langle x, y \rangle \in f$. Это y называется *значением функции f на x* . Можно дать и явное определение значения функции в нашем языке:

$$f'x \equiv f(x) \equiv \bigcup \{y \in \text{rng}(f) \mid \langle x, y \rangle \in f\}.$$

Основное свойство этого обозначения:

$$\text{Fnc}(f) \wedge x \in \text{dom}(f) \supset ((z = f'x) \equiv (\langle x, z \rangle \in f)).$$

▷ Пусть $\text{Fnc}(f)$ и $x \in \text{dom}(f)$. Так как $x \in \text{dom}(f)$, то найдется u такое, что $\langle x, u \rangle \in f$. Пусть $Q = \{y \in \text{rng}(f) \mid \langle x, y \rangle \in f\}$. По определению $y \in Q \equiv \langle x, y \rangle \in f$. Но по условию униформности $\langle x, y \rangle \in f \wedge \langle x, u \rangle \in f \supset y = u$. Таким образом, $y \in Q \equiv (y = u)$. Отсюда следует $Q = \{u\}$. А тогда $\bigcup Q = \bigcup \{u\} = u$. Но $\bigcup Q$ есть $f'x$, так что $f'x = u$. Если $z = f'x$, то $z = u$ и, значит, $\langle x, z \rangle \in f$. Обратное, если $\langle x, z \rangle \in f$, то ввиду $\langle x, u \rangle \in f$ и условия униформности $z = u$, т. е. $z = f'x$. □

Не следует путать два обозначения, имеющие совсем разный смысл:

$f(x) \equiv f'x$ — значение функции f в x .

$f[x] \equiv f''x$ — образ множества x по отношению к функции f .

Аналогично, различный смысл имеют и формулы:

$f: X \rightarrow Y$ — функция f отображает множество X в множество Y .

$f: x \mapsto y$ — функция f переводит x в y .

Заметим, что всякая функция есть в то же время и отношение, так что все определения, предназначенные для изучения отношений, могут быть использованы и для функций. Таким образом, задаются область определения функций, область значений, ограничение функции множеством, композиция функций.

Л е м м а. *Композиция двух функций есть функция:*

$$. \text{Fnc}(f) \wedge \text{Fnc}(g) \supset \text{Fnc}(f \circ g).$$

▷ Пусть $\text{Fnc}(f)$, $\text{Fnc}(g)$, проверим условие равномерности для $f \circ g$. Пусть $\langle u, v_1 \rangle \in f \circ g$, $\langle u, v_2 \rangle \in f \circ g$, покажем $v_1 = v_2$. По определению композиции найдутся w_1 и w_2 такие, что

$$\langle u, w_1 \rangle \in g, \quad \langle w_1, v_1 \rangle \in f$$

и

$$\langle u, w_2 \rangle \in g, \quad \langle w_2, v_2 \rangle \in f.$$

Из условия равномерности для g имеем $w_1 = w_2$. Но тогда $v_1 = v_2$ из условия равномерности для f . \square

Л е м м а. *Значение композиции равно последовательному вычислению значений составляющих функций:*

$$. \text{Fnc}(f) \wedge \text{Fnc}(g) \wedge x \in \text{dom}(g) \wedge$$

$$(g'x) \in \text{dom}(f) \supset (f \circ g)'x = f'(g'x).$$

Обратите внимание, что сначала вычисляется значение функции g (правой в композиции).

▷ Пусть $v = (f \circ g)'x$. Тогда $\langle x, v \rangle \in (f \circ g)$ и, значит, найдется w , $\langle x, w \rangle \in g$, $\langle w, v \rangle \in f$ по определению композиции. Но тогда $w = g'x$ и $v = f'w$, т. е. $v = f'(g'x)$. \square

Для данной функции f обратное отношение $f^{-1} = \text{Conv}(f)$ может и не быть функцией. Функция f называется *взаимно-однозначной функцией*, или *биекцией*, если обратное отношение также является функцией:

$$(1-1) (f) \Leftrightarrow \text{Fnc}(f) \wedge \text{Fnc}(\text{Conv}(f)).$$

Множество *всех функций* из множества X в множество Y обозначим через $(X \rightarrow Y)$. Таким образом,

$$. f \in (X \rightarrow Y) \equiv f : X \rightarrow Y.$$

Точное определение выделением:

$$(X \rightarrow Y) \Leftrightarrow \{f \in P(X \times Y) \mid f : X \rightarrow Y\}.$$

Упражнение. Убедитесь, что ограничение $f \in P(X \times Y)$ фиктивно и следует из условия $f : X \rightarrow Y$.

Можно было бы естественно развивать и теорию функций нескольких переменных. Так, функция от двух переменных есть по определению трехместное отношение, равномерное по последней координате.

9. Пусть $t(i)$ — терм языка ZF^+ с выделенной переменной i . Мы будем писать иногда t_i вместо $t(i)$, чтобы подчеркнуть особую роль переменной i . Пусть I — произвольный терм, не содержащий свободно переменную i .

Можно ввести обозначение

$$\{t_i \mid i \in I\}$$

таким образом, что

$$u \in \{t_i \mid i \in I\} \equiv (\exists i \in I) (u = t_i).$$

Это обозначение мы читаем так: «семейство всех t_i для $i \in I$ ».

Аналогично для терма $t(i, j)$ с двумя выделенными переменными определяется семейство $\{t_{ij} \mid i \in I, j \in J\}$, так что

$$u \in \{t_{ij} \mid i \in I, j \in J\} \equiv (\exists i \in I) (\exists j \in J) (u = t_{ij}).$$

Подобным образом можно определить семейства множеств по термам с большим количеством выделенных переменных.

▷ Эти обозначения вводятся с помощью определения подстановкой. Пусть, например, $t(i, j)$ терм, i, j — две различные переменные, а I, J — произвольные термы, не содержащие свободно переменных i и j . Определим

$$\begin{aligned} \{t_{ij} \mid i \in I, j \in J\} &\Leftarrow \{y \mid x \in I \times J, (x \mapsto y), \\ &\exists ij (x = \langle i, j \rangle \wedge y = t_{ij})\}. \end{aligned}$$

Тогда согласно п. 12 § 1 $u \in \{t_{ij} \mid i \in I, j \in J\} \Leftrightarrow$ существует $x \in I \times J$ такое, что в точности для одного u имеем

$$\exists ij (x = \langle i, j \rangle \wedge u = t_{ij}).$$

Но для всякого $x \in I \times J$ такое u всегда существует, и единственно. В самом деле, по $x \in I \times J$ однозначно определяются i и j , $x = \langle i, j \rangle$, а тогда u однозначно определяется из равенства $u = t_{ij}$. Таким образом, $u \in \{t_{ij} \mid i \in I, j \in J\}$ тогда и только тогда, когда существует $i \in I, j \in J, u = t_{ij}$. □

Эти обозначения позволяют компактно и наглядно выразить в языке ZF^+ многие популярные конструкции обыденной математики.

Например,

$$U \times V = \{\langle u, v \rangle \mid u \in U, v \in V\},$$

здесь u и v — выделенные переменные.

Заметьте, что выделенные переменные связаны в нашем обозначении.

Определим *объединение семейства множеств*

$$\bigcup_{i \in I} t_i \Leftarrow \bigcup \{t_i \mid i \in I\}.$$

Основное свойство:

$$\begin{aligned} z \in \bigcup_{i \in I} t_i &\equiv (\exists i \in I) (z \in t_i). \\ \triangleright z \in \bigcup_{i \in I} t_i &\equiv z \in \bigcup \{t_i \mid i \in I\} \equiv \\ &\equiv (\exists u \in \{t_i \mid i \in I\}) (z \in u) \equiv \exists u (\exists i \in I) \\ &(u = t_i \wedge z \in u) \equiv (\exists i \in I) (z \in t_i). \quad \square \end{aligned}$$

Определим пересечение семейства множеств

$$\bigcap_{i \in I} t_i \equiv \bigcap \{t_i \mid i \in I\}.$$

$$1). I \neq \emptyset \supset z \in \bigcap_{i \in I} t_i \equiv (\forall i \in I) (z \in t_i);$$

$$2). \bigcap_{i \in \emptyset} t_i = \emptyset.$$

▷ См. п. 3 § 2. □

Декартово произведение семейства множеств определяется следующим образом:

$$\prod_{i \in I} t_i \equiv \{f \in (I \rightarrow \bigcup_{i \in I} t_i) \mid (\forall i \in I) ((f' i) \in t_i)\}.$$

10. В обычной математике под функцией иногда понимают аналитическое выражение, зависящее от переменной. Например, говорят о «функции» $x^2 + x + 1$. В точном языке такого рода выражениям соответствуют не функции, а термы. Терм описывает не функцию, а ее значения.

Если по терму $t(x)$ мы желаем образовать функцию с областью определения z , то она (функция) будет определяться уже другим термом

$$f = \{ \langle x, t(x) \rangle \mid x \in z \}.$$

Для всех $x \in z$ будет

$$.f : x \mapsto t(x).$$

§ 4. НАТУРАЛЬНЫЕ ЧИСЛА В ТЕОРИИ МНОЖЕСТВ. ЗАПИСЬ МАТЕМАТИЧЕСКИХ УТВЕРЖДЕНИЙ В ЯЗЫКЕ ТЕОРИИ МНОЖЕСТВ

1. Рассмотрим последовательность множеств:

$$\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}, \dots$$

Эта последовательность получается, если начиная с пустого множества последовательно применять операцию $Sx = x \cup \{x\}$:

$$\{\emptyset\} = S(\emptyset), \{\emptyset, \{\emptyset\}\} = S(\{\emptyset\}) \text{ и т. д.}$$

Первый член нашей последовательности вовсе не содержит элементов, второй содержит один элемент, третий — два и т. д.

Обратите внимание на важную особенность нашей последовательности: каждый член в ней равен в точности множеству всех предыдущих членов последовательности. Например, члену $\{\emptyset, \{\emptyset\}\}$ предшествуют в точности члены \emptyset и $\{\emptyset\}$. Если упорядочить члены нашей последовательности, считая, что $a < b \Leftrightarrow a$ появилось в последовательности раньше, чем b , то для членов нашей последовательности

$$a < b \Leftrightarrow a \in b.$$

Таким образом, отношение принадлежности задает линейный порядок на нашей последовательности.

Фон Нейман предложил определить *натуральные числа* в теории множеств как члены вышеуказанной последовательности:

$$0 \Leftarrow \emptyset, 1 \Leftarrow \{\emptyset\}, 2 \Leftarrow \{\emptyset, \{\emptyset\}\}, \dots$$

и вообще, если задано натуральное число n , то следующее число есть S_n .

Аксиомы бесконечности (п. 2 § 2) как раз и утверждают, что ω есть в точности множество всех натуральных чисел по фон Нейману. В самом деле, первые два утверждения

$$\emptyset \in \omega, (\forall x \in \omega) (Sx \in \omega)$$

гласят, что ω во всяком случае содержит все натуральные числа, а третье утверждение

$$\forall x (Prog(x) \supset (\omega \subseteq x))$$

гласит, что ω — «самое маленькое» из множеств, содержащих все натуральные числа, т. е., интуитивно говоря, состоит только из натуральных чисел.

Основные свойства натуральных чисел выражаются следующими тремя утверждениями:

$$1) . (\forall x \in \omega) (Sx \neq 0);$$

$$2) . (\forall xy \in \omega) (Sx = Sy \equiv x = y);$$

3) для всякой формулы $\varphi(x)$ выполняется следующее утверждение, называемое *принципом полной математической индукции*:

$$. \varphi(0) \wedge (\forall x \in \omega) (\varphi(x) \supset \varphi(Sx)) \supset (\forall x \in \omega) \varphi(x).$$

▷ 1) Очевидно, $x \in Sx = x \cup \{x\}$. Если было бы $Sx = 0$, то $x \in \emptyset$, что невозможно.

3) Допустим $\varphi(0)$ и

$$(\forall x \in \omega) (\varphi(x) \supset \varphi(Sx))$$

и докажем $(\forall x \in \omega) \varphi(x)$. С этой целью рассмотрим множество $u = \{x \in \omega \mid \varphi(x)\}$. Из допущений следует, что $Prog(u)$. А тогда по аксиомам бесконечности имеем $\omega \subseteq u$, что и означает $(\forall x \in \omega) \varphi(x)$.

2) Для установления этого факта нам понадобится доказать два вспомогательных утверждения с помощью принципа полной математической индукции.

Первое утверждение:

$$(\forall x \in \omega) \forall yz (y \in z \wedge z \in x \supset y \in x).$$

Для $x = 0$ это утверждение

$$\forall yz (y \in z \wedge z \in \emptyset \supset y \in \emptyset)$$

тривиально в силу ложности посылки $z \in \emptyset$. Рассмотрим произвольно число $x \in \omega$ и, допустив

$$\forall yz (y \in z \wedge z \in x \supset y \in x),$$

докажем

$$\forall yz (y \in z \wedge z \in Sx \supset y \in Sx).$$

Итак, пусть $y \in z$, $z \in Sx$. Из $z \in Sx$ следует $z \in x$ или $z = x$. В первом случае $y \in z$, $z \in x$ и, значит, $y \in x$ по индуктивному предположению. Отсюда $y \in Sx$.

Во втором случае $y \in Sx$ ввиду $z \in Sx$.

Первое утверждение доказано.

Второе утверждение:

$$\forall x (x \notin x).$$

Очевидно, $\emptyset \notin \emptyset$. Допустим для $x \in \omega$, что $x \notin x$, и докажем $Sx \notin Sx$. Предположим противное, и пусть $Sx \in Sx$. Тогда $Sx \in x$ или $Sx = x$. В первом случае ввиду $x \in Sx$ и первого утверждения имеем $x \in x$, что противоречит индуктивному предположению. Во втором случае ввиду $x \in Sx$ непосредственно заключаем $x \in x$, что вновь противоречит индуктивному предположению. Второе утверждение доказано.

Установим теперь 2). Пусть $Sx = Sy$, докажем $x = y$ (обратная импликация следует из общих законов равенства). Так как $x \in Sx$, то $x \in Sy$ и, значит, $x \in y$ или $x = y$. Во втором случае утверждение доказано, так что достаточно показать, что случай $x \in y$ невозможен. Из $y \in Sy$ заключаем $y \in Sx$, т. е. $y \in x$ или $y = x$. Но если $x \in y$ и $y \in x$, то по первому вспомогательному утверждению тогда $x \in x$, что противоречит второму утверждению. \square

Оказывается, все, что требуется в математике от натуральных чисел, вытекает уже только из указанных трех утверждений, называемых *аксиомами Пеано*. Можно определять натуральные числа в теории множеств и по-иному, не обязательно по фон Нейману, важно лишь, чтобы выполнялись утверждения 1)–3).

Например, из аксиом Пеано средствами ZF^+ можно уже доказать существование функций сложения, умножения, возведения в степень для натуральных чисел и вывести все обычные свойства этих функций.

Более того, оказывается, что аксиомы Пеано фактически однозначно определяют натуральный ряд в ZF^+ . Точнее, если имеется некоторое множество ω^* , элемент $0^* \in \omega^*$ и функция, определенная на ω^* ; $S^*: \omega^* \rightarrow \omega^*$, причем так, что для 0^* , ω^* , S^* выполняются аксиомы Пеано, то существует естественный изоморфизм между ω и ω^* , т. е. существует взаимно-однозначная функция $f: \omega \rightarrow \omega^*$ на все множество ω^* такая, что $f'0 = 0^*$ и $f'(Sx) = S^*(f'x)$. Кроме того, такая функция f единственна. Как говорят, аксиомы Пеано *категоричны в теории множеств*, т. е. фактически однозначно определяют натуральный ряд, если позволить себе пользоваться средствами теории множеств.

Основные свойства натуральных чисел, действительных чисел изучаются в учебниках по числовым системам и по математи-

ческому анализу, и мы не будем здесь повторять этого изучения, но все же наметим коротко определение множества R всех действительных чисел в стиле Дедекинда.

Сначала определим множество Rat_0 канонических обозначений для рациональных чисел. А именно

$$x \in \text{Rat}_0 \equiv \exists u v w (x = \langle u, v, w \rangle \wedge (u=0 \vee u=1) \wedge v \in \omega \wedge w \in \omega \wedge w \neq 0).$$

Если $x = \langle u, v, w \rangle$, то u назовем знаком x , v — числителем x и w — знаменателем x . Тройку вида $\langle 0, v, w \rangle$ будем обозначать через v/w или $+v/w$, а тройку вида $\langle 1, v, w \rangle$ будем обозначать через $-v/w$.

Определим далее естественное отношение эквивалентности между каноническими обозначениями для рациональных чисел таким образом, что

$$\begin{aligned} \langle u, v, w \rangle =_r \langle u, v_1, w_1 \rangle &\Leftrightarrow v \cdot w_1 = v_1 \cdot w, \\ \langle 0, v, w \rangle =_r \langle 1, v_1, w_1 \rangle &\Leftrightarrow v = v_1 = 0, \end{aligned}$$

здесь умножение справа есть обычное умножение натуральных чисел.

Множество Rat всех рациональных чисел определяется как множество всех классов эквивалентности Rat_0 по этому отношению эквивалентности. Более формально

$$\begin{aligned} \text{Rat} \equiv \{y \in P(\text{Rat}_0) \mid (\exists z \in \text{Rat}_0) \forall x \\ (x \in y \equiv (x =_r z))\}. \end{aligned}$$

Далее, на множестве Rat обычным образом вводятся все основные действия над рациональными числами. Например, умножение рациональных чисел описывается следующим образом, отражающим известное школьное правило «умножения дроби на дробь»:

$$\begin{aligned} x_1 \cdot_r x_2 &\Leftrightarrow \{z \in \text{Rat}_0 \mid \exists u_1 v_1 w_1 u_2 v_2 w_2 \\ ((\langle u_1, v_1, w_1 \rangle \in x_1) \wedge (\langle u_2, v_2, w_2 \rangle \in x_2) \wedge \\ (z =_r \langle u, v_1 \cdot v_2, w_1 \cdot w_2 \rangle)) \wedge \\ (u=0 \vee u=1) \wedge (u=1 \equiv (u_1 \neq u_2))\}. \end{aligned}$$

Разумеется, следует проверить, что эта операция действительно по рациональным числам x_1 и x_2 выдает рациональное число и удовлетворяет обычным свойствам умножения рациональных чисел. При желании можно определить и константу, функцию от двух аргументов — умножение рациональных чисел:

$$\begin{aligned} \cdot_r &\Leftrightarrow \{z \in \text{Rat}^3 \mid (\exists x_1 x_2 x_3 \in \text{Rat}) \\ (z = \langle x_1, x_2, x_3 \rangle \wedge x_3 = x_1 \cdot_r x_2)\}. \end{aligned}$$

Тогда будем иметь

$$. (\forall x_1 x_2 \in \text{Rat}) ((\cdot_r ' \langle x_1, x_2 \rangle = x_1 \cdot_r x_2) .$$

В таком же стиле можно ввести операции сложения $x +_r y$, вычитания $x -_r y$, взятия модуля $|x|_r$ для рациональных чисел, а также отношение $x <_r y$ строгого неравенства между рациональными числами.

Сечением в области рациональных чисел называется, как известно, разбиение множества Rat рациональных чисел на два непустых подмножества X и Y таким образом, что X вместе с каждым числом содержит и все меньшие, а Y вместе с каждым числом содержит и все большие. Определение сечения в нашем языке:

$$\begin{aligned} \text{Ct}(x) \Leftarrow & (\exists X \subseteq \text{Rat}) (\exists Y \subseteq \text{Rat}) (x = \langle X, Y \rangle \wedge \\ & X \neq \emptyset \wedge Y \neq \emptyset \wedge X \cup Y = \text{Rat} \wedge X \cap Y = \emptyset \wedge \\ & (\forall uv \in \text{Rat}) (u <_r v \supset (v \in X \supset u \in X) \wedge \\ & (u \in Y \supset v \in Y))). \end{aligned}$$

Сечение $\langle X, Y \rangle$ назовем *правильным*, если в нижнем классе X этого сечения отсутствует наибольшее число. В нашем языке понятие правильного сечения можно записать в виде следующей формулы:

$$\begin{aligned} \text{Cor Ct}(x) \Leftarrow & \text{Ct}(x) \wedge \forall XY (x = \langle X, Y \rangle \\ & \supset (\forall u \in X) (\exists v \in X) (u <_r v)). \end{aligned}$$

Теперь, следуя знакомой схеме, мы отождествим множество R всех действительных чисел со множеством всех правильных сечений:

$$R \Leftarrow \{ x \in P(\text{Rat}) \times P(\text{Rat}) \mid \text{Cor Ct}(x) \}.$$

Это типичное определение по схеме выделения, где слева от черты стоит фиктивное ограничение.

Далее следует определить на множестве R все обычные действия над действительными числами. Отношение строгого неравенства, например, можно ввести следующим образом:

$$\langle X_1, Y_1 \rangle < \langle X_2, Y_2 \rangle \Leftarrow Y_1 \cap X_2 \neq \emptyset.$$

2. Мы не будем заниматься далее развитием теории действительных чисел и примем, что в ZF^+ могут быть определены множество R всех действительных чисел и все обычные операции над действительными числами: сложение, умножение, взятие модуля действительных чисел и т. п.

Напомним определение ограниченных кванторов (см. п. 8 § 1):

$$\begin{aligned} (\forall x \in y) \varphi(x) & \Leftarrow \forall x (x \in y \supset \varphi(x)); \\ (\exists x \in y) \varphi(x) & \Leftarrow \exists x (x \in y \wedge \varphi(x)). \end{aligned}$$

Ценность этого определения помимо его компактности состоит еще и в том, что ограниченные кванторы во многом ведут себя аналогично обычным, особенно если $y \neq \emptyset$. Так, для ограниченных кванторов имеет место аналог законов де Моргана. Далее,

$$.y \neq \emptyset \supset ((\forall x \in y) \varphi(x) \supset (\exists x \in y) \varphi(x)).$$

Можно пойти еще немного далее в употреблении сокращений. А именно фиксируем некоторое множество u , $u \neq \emptyset$. В языке ZF^+ это множество может изображаться, в частности, некоторым замкнутым термом. Фиксируем теперь некоторый набор F переменных: a, b, c, \dots , которые условимся считать *пробегающими элементами множества u* . Более формально это означает, что:

1) во всяком доказательстве, рассматривая элемент a , изображаемый переменной из набора F , мы автоматически считаем верным $.a \in u$;

2) во всех формулах кванторы

$$\forall a \varphi(a), \quad \exists a \varphi(a)$$

следует понимать как ограниченные

$$(\forall a \in u) \varphi(a), \quad (\exists a \in u) \varphi(a);$$

3) термы вида $\{a | \varphi(a)\}$ следует рассматривать как $\{a \in u | \varphi(a)\}$.

Коротко говоря, мы всюду опускаем ограничение $\in u$, если речь идет о переменных набора F . При аккуратной полной записи это ограничение следует, конечно, восстанавливать. Указанием на то, что следует добавлять ограничение, служит то обстоятельство, что переменная берется из набора F .

Такой способ употребления переменных называется *введением подчиненных переменных*. Мы говорим, что переменные из списка F *подчинены условию $\in u$* .

Например, мы согласимся употреблять буквы m, n, k, \dots , быть может, с индексами, для обозначения *натуральных чисел*, т. е. переменные набора m, n, k, \dots подчиним условию $\in \omega$.

В этих обозначениях принцип полной математической индукции запишется в виде

$$\varphi(0) \wedge \forall m (\varphi(m) \supset \varphi(Sm)) \supset \forall m \varphi(m).$$

Его более развернутая запись имеет вид

$$\varphi(0) \wedge (\forall m \in \omega) (\varphi(m) \supset \varphi(Sm)) \supset (\forall m \in \omega) \varphi(m).$$

Упражнение. Обдумайте следующие определения:

1) a есть последовательность действительных чисел:

$$(a : \omega \rightarrow R).$$

2) b есть подпоследовательность последовательности a :

$$(a : \omega \rightarrow R) \wedge (b : \omega \rightarrow R) \wedge$$

$$(\exists f \in (\omega \rightarrow \omega)) \quad \forall m ((f'm < f'Sm) \wedge (b'm = a'(f'm))).$$

Согласимся, далее, употреблять буквы $\alpha, \beta, \gamma, \delta, \varepsilon, \dots$ быть может с индексами, для обозначения действительных чисел, т. е. переменные этого набора подчиним условию $\in \mathbb{R}$.

Отрезок $[\alpha, \beta]$ действительных чисел:

$$[\alpha, \beta] \Leftarrow \{\gamma \mid \alpha \leq \gamma \wedge \gamma \leq \beta\}.$$

Функция f непрерывна в точке α отрезка $[0, 1]$:

$$(f : [0, 1] \rightarrow \mathbb{R}) \wedge \alpha \in [0, 1] \wedge (\forall \varepsilon > 0) (\exists \delta > 0) (\forall \beta \in [0, 1]) (|\alpha - \beta| < \delta \supset |f'\alpha - f'\beta| < \varepsilon).$$

Функция f непрерывна в каждой точке отрезка $[0, 1]$:

$$(f : [0, 1] \rightarrow \mathbb{R}) \wedge (\forall \varepsilon > 0) (\forall \alpha \in [0, 1]) (\exists \delta > 0) (\forall \beta \in [0, 1]) (|\alpha - \beta| < \delta \supset |f'\alpha - f'\beta| < \varepsilon).$$

Функция f равномерно непрерывна на отрезке $[0, 1]$:

$$(f : [0, 1] \rightarrow \mathbb{R}) \wedge \forall \varepsilon > 0 (\exists \delta > 0) (\forall \alpha \in [0, 1]) (\forall \beta \in [0, 1]) (|\alpha - \beta| < \delta \supset |f'\alpha - f'\beta| < \varepsilon).$$

Обратите внимание на различие в порядке кванторов по δ и по α в определении непрерывности и равномерной непрерывности. В этой перестановке кванторов — важное различие между непрерывностью и равномерной непрерывностью.

Знание логических законов позволяет теперь быстро преобразовать формулы к нужному нам виду. Сформулируем, например, утверждение, что функция f на отрезке $[0, 1]$ неравномерно непрерывна, причем так, чтобы отрицание не фигурировало в этом утверждении. С этой целью в определении равномерной непрерывности следует поставить отрицание перед вторым членом конъюнкции, а затем, применяя законы де Моргана и отрицание импликации, пронести отрицание внутрь. В результате получим:

$$(f : [0, 1] \rightarrow \mathbb{R}) \wedge (\exists \varepsilon > 0) (\forall \delta > 0) (\exists \alpha \in [0, 1]) (\exists \beta \in [0, 1]) (|\alpha - \beta| < \delta \wedge |f'\alpha - f'\beta| \geq \varepsilon).$$

3. У читателя должно возникнуть правильное впечатление, что практически любое математическое утверждение может быть записано формулой ZF^+ , а имя любого математического объекта исследования может быть записано в виде термина этого языка.

Приведем еще один пример. *Топологическим пространством* называется, как известно, непустое множество X , на котором определено семейство S его подмножеств, называемых открытыми множествами, причем: 1) \emptyset и X суть открытые множества;

2) пересечение любых двух открытых множеств вновь открыто;

3) объединение любого семейства открытых множеств открыто.

На нашем языке «быть топологическим пространством» может быть записано формулой:

$$\begin{aligned} \text{Top}(x) \Leftrightarrow \exists X S (x = \langle X, S \rangle \wedge X \neq \emptyset \wedge \\ S \subseteq P(X) \wedge \emptyset \in S \wedge X \in S \wedge \\ (\forall yz \in S) (y \cap z \in S) \wedge (\forall u \subseteq S) (\cup u \in S)). \end{aligned}$$

4. Язык ZF^+ , как мы уже отмечали, не является языком первого порядка ввиду особой структуры своих термов. Рассмотрим язык первого порядка ZF . Этот язык содержит один сорт переменных x, y, z, \dots для множеств и единственный атомарный предикат \in . Язык ZF не содержит ни констант, ни функциональных символов.

ZF составляет часть языка ZF^+ . Тем не менее, как мы увидим, всякая формула ZF^+ эквивалентна некоторой формуле ZF . В этом смысле термы не являются необходимой принадлежностью языка теории множеств и введены нами лишь для удобства записи математических утверждений. В математической логике чаще употребляется именно язык ZF . А именно, для всякой формулы φ языка ZF^+ мы определим формулу φ^0 языка ZF с теми же параметрами такую, что $\varphi \equiv \varphi^0$. Что касается термов языка ZF^+ , то для всякой формулы ZF^+ вида $y = t$, где t — терм и y — переменная, не входящая свободно в t , мы определим ZF -формулу $(y = t)^*$ так, что $(y = t) \equiv (y = t)^*$. Формулы φ^0 и $(y = t)^*$ определяются одновременной индукцией по определению формул и термов п. 1 § 2:

- 1) $(y = x)^* \Leftrightarrow (y = x)$;
- 2) $(t \in r)^0 \Leftrightarrow \exists y_1 y_2 ((y_1 = t)^* \wedge (y_2 = r)^* \wedge (y_1 \in y_2))$;
- 3) $(\varphi \wedge \psi)^0 \Leftrightarrow \varphi^0 \wedge \psi^0$;
 $(\varphi \vee \psi)^0 \Leftrightarrow \varphi^0 \vee \psi^0$;
 $(\varphi \supset \psi)^0 \Leftrightarrow \varphi^0 \supset \psi^0$;
 $(\neg \varphi)^0 \Leftrightarrow \neg \varphi^0$;
- 4) $(\forall x \varphi)^0 \Leftrightarrow \forall x \varphi^0$; $(\exists x \varphi)^0 \Leftrightarrow \exists x \varphi^0$;
- 5) $(y = \emptyset)^* \Leftrightarrow \forall x (x \notin y)$;
 $\text{Prog}^0(x) \Leftrightarrow \exists u (\forall v (v \notin u) \wedge u \in x) \wedge (\forall u \in x) (\exists v \in x) \forall z (z \in v \equiv (z \in u \vee z = u))$;
 $(y = \omega)^* \Leftrightarrow \text{Prog}^0(y) \wedge \forall x (\text{Prog}^0(x) \supset y \subseteq x)$;
- 6) $(y = \{t, r\})^* \Leftrightarrow \exists y_1 y_2 ((y_1 = t)^* \wedge (y_2 = r)^* \wedge \forall z (z \in y \equiv (z = y_1 \vee z = y_2)))$;
- 7) $(y = Pt)^* \Leftrightarrow \exists y_1 ((y_1 = t)^* \wedge \forall z (z \in y \equiv z \subseteq y_1))$;
 $(y = \cup t)^* \Leftrightarrow \exists y_1 ((y_1 = t)^* \wedge \forall z (z \in y \equiv (\exists v \in y_1) (z \in v)))$;

$$8) (y = \{x \in t \mid \varphi(x)\})^* \Leftrightarrow \exists y_1 ((y_1 = t)^* \wedge \forall z (z \in y \equiv (z \in y_1 \wedge \varphi^0(z))));$$

$$9) (y = \{v \mid u \in t, (u \mapsto v), \varphi(u, v)\})^* \Leftrightarrow \exists y_1 ((y_1 = t)^* \wedge \forall v (v \in y \equiv (\exists u \in y_1 (\varphi^0(u, v) \wedge \forall w (\varphi^0(u, w) \supset u = w))))).$$

Используя эту длинную индуктивную процедуру, можно по всякой ZF^+ -формуле φ получить эквивалентную ZF -формулу φ^0 . Доказательство эквивалентности проводится непосредственно индукцией и мы не будем на нем останавливаться.

§ 5. О КONTИНУУМ-ГИПОТЕЗЕ И АКСИОМЕ ВЫБОРА

1. Два множества назовем *равномощными*, если существует биекция, отображающая одно множество на другое:

$$x \simeq y \Leftrightarrow \exists f ((1-1)(f) \wedge \text{dom}(f) = x \wedge \text{rng}(f) = y).$$

Множество называется *конечным*, если оно равномощно некоторому натуральному числу (напомним, что в теории множеств каждое натуральное число отождествляется с множеством всех чисел, ему предшествующих):

$$\text{Fin}(x) \Leftrightarrow (\exists n \in \omega) (x \simeq n).$$

Множество называется *счетным*, если оно равномощно множеству всех натуральных чисел:

$$\text{Count}(x) \Leftrightarrow x \simeq \omega.$$

2. Теорема (Кантор).

$$\neg (x \simeq Px).$$

▷ Предположим противное, и пусть $x \simeq Px$. Тогда существует биекция f , $\text{dom } f = x$, $\text{rng } f = Px$. Рассмотрим множество v ,

$$v = \{z \in x \mid z \notin f^{-1}z\}.$$

Очевидно, $v \subseteq x$, т. е. $v \in Px$. Кроме того, по определению

$$(\forall z \in x) (z \in v \equiv z \notin f^{-1}z).$$

Так как f — биекция, то существует $z_0 \in x$, такое, что $f^{-1}z_0 = v$.

Подставляя в вышеописанную эквивалентность вместо z конкретное z_0 и замечая, что $f^{-1}z_0 = v$, получим

$$z_0 \in v \equiv z_0 \notin v,$$

что невозможно по законам логики.

Обратите внимание на аналогию между этим рассуждением и рассуждением в парадоксе Рассела. □

Следствие. Множество всех подмножеств натуральных чисел несчетно:

$$\neg \text{Count}(P\omega).$$

3. Знаменитую континуум-гипотезу Кантора можно сформулировать следующим образом: всякое семейство подмножеств натурального ряда либо конечно, либо счетно, либо равномощно множеству всех подмножеств натурального ряда. Точная формулировка:

$$(\forall y \subseteq P\omega) (\text{Fin}(y) \vee \text{Count}(y) \vee y \simeq P\omega).$$

Суть этого утверждения состоит в том, что не существует бесконечных множеств, имеющих мощность, строго промежуточную между мощностями множеств ω и $P\omega$.

Как показали Гедель и Коэн, континуум-гипотеза не зависит от остальных аксиом ZF, т. е. средствами логики ее нельзя ни доказать, ни опровергнуть с помощью остальных аксиом, даже привлекая аксиому выбора.

4. *Аксиома выбора* утверждает, что для всякого семейства множеств x существует функция f такая, что если $z \in x$ и $z \neq \emptyset$, то f выдает элемент z в качестве значения на z . Функцию f можно назвать выбирающей функцией, она «выбирает» по элементу $(f'z) \in z$ из каждого множества z , $z \in x$, $z \neq \emptyset$.

Точная формулировка аксиомы выбора такова:

$$\begin{aligned} \forall x \exists f (\text{Fnc}(f) \wedge \text{dom}(f) = x \wedge \\ (\forall z \in x) (z \neq \emptyset \supset (f'z) \in z)). \end{aligned}$$

Истинно ли это утверждение?

Особенность этого утверждения такова, что функция f никак не определяется по множеству x — утверждается лишь ее существование. Теоремы, полученные с использованием аксиомы выбора, часто имеют ту же особенность: доказываются существование множеств, обладающих теми или иными свойствами, и в то же время не указывается никакого индивидуального примера такого множества, никакого способа его определения. Типичный пример — доказательство существования неизмеримых множеств. На этом основании многие математики подвергли критике неограниченное использование аксиомы выбора.

Тем не менее в практических математических рассуждениях аксиома выбора довольно широко используется. Как показали Гедель и Коэн, аксиома выбора также не зависит от аксиом ZF.

В следующем параграфе мы еще уточним общую постановку задачи о независимости утверждений от теории.

5. Мы говорим, что множество x имеет мощность, меньшую или равную мощности множества y , если x можно взаимнооднозначно отобразить на подмножество y . Точнее:

$$x \preceq y \Leftrightarrow \exists f ((1-1)(f) \wedge \text{dom}(f) = x \wedge \text{rng}(f) \subseteq y).$$

Теорема (Кантор — Шредер — Бернштейн). Если $x \preceq y$ и $y \preceq x$, то $x \simeq y$.

▷ Достаточно показать, что если $a_1 \subseteq b \subseteq a$ и $a_1 \simeq a$, то $b \simeq a$. Пусть (1-1) (f) , $\text{dom}(f) = a$, $\text{rng}(f) = a_1$. Положим

$$a_0 = a, \quad a_1 = f''a_0, \quad a_2 = f''a_1, \dots$$

$$b_0 = b, \quad b_1 = f''b_0, \quad b_2 = f''b_1, \dots$$

Определим $g(x) = f(x)$, если $x \in a_n \setminus b_n$ для некоторого n . В противном случае $g(x) = x$. Тогда (1-1) (g) , $\text{dom}(g) = a$, $\text{rng}(g) = b$. □

Упражнение. Докажите:

$$x \subseteq y \wedge y \subseteq z \supset x \subseteq z.$$

6. Замечание. В практической математике помимо множеств употребляют иногда еще и *классы*.

Рассмотрим произвольную формулу $\varphi(x)$ языка ZF^+ , в которой выделена переменная x . Удобно бывает считать, что такая формула всегда определяет некоторый объект исследования $\{x | \varphi(x)\}$. Этот объект называется классом всех множеств x , удовлетворяющих условию $\varphi(x)$.

Для некоторых формул $\varphi(x)$ специального вида класс $\{x | \varphi(x)\}$ оказывается множеством, в общем же случае понятие класса оказывается шире, чем понятие множества. Так, $\{x | x \notin x\}$ есть класс, не являющийся множеством, *собственный класс*. Интуитивно говоря, собственные классы — это «очень большие» совокупности неопределенной мощности. Элементы класса всегда суть множества (а не собственные классы). По аналогии с множествами над классами также могут быть определены некоторые теоретико-множественные операции. Например, если даны классы

$$X = \{x | \varphi(x)\}, \quad Y = \{y | \psi(y)\},$$

то можно определить класс

$$X \cup Y = \{x | \varphi(x) \vee \psi(x)\}.$$

Можно образовать класс V всех множеств, это будет собственный класс, не множество, *универсум* теории множеств.

Семейство множеств образует столь мощную структуру, что в практической математике нет реальной необходимости использовать собственные классы. Упоминания о классах обычно можно избежать, рассматривая вместо классов условия, их определяющие. Тем не менее исследуются языки и теории, содержащие классы, такова, например, известная теория Геделя—Бернайса—Неймана.

§ 6. АКСИОМАТИЧЕСКАЯ ТЕОРИЯ МНОЖЕСТВ ЦЕРМЕЛО — ФРЕНКЕЛЯ

1. В первом параграфе мы ввели логико-математический язык наивной теории множеств и начали систематически развивать теорию множеств в виде серии утверждений в этом языке.

Доказательства утверждений при этом являлись содержательными, «наивными». Мы исходили из представления о некоей структуре \mathcal{M} , объекты которой называются множествами. В этой структуре выполняются простые и естественные законы, относящиеся к множествам. Утверждения мы доказывали, так сказать, путем непосредственного изучения этой воображаемой структуры, а точнее условий-аксиом, на нее налагаемых. Такой подход совершенно естествен для математика-неспециалиста по математической логике.

Однако более внимательный анализ показал, что предложенные аксиомы ведут к противоречиям — парадоксам. Пришлось признать, что структуры \mathcal{M} , удовлетворяющей предложенным законам наивной теории множеств, по-видимому, не существует. Выход из положения мы нашли в том, чтобы рассмотреть более ограничительную структуру \mathcal{M}_0 , удовлетворяющую уже некоторым более стеснительным и гораздо менее естественным условиям, выражаемым в некотором более сложном языке ZF^+ . Доказательства утверждений при этом оставались по-прежнему наивными.

Возникает вопрос, в какой мере законно рассмотрение структуры \mathcal{M}_0 , не придут ли наши рассуждения вновь к парадоксам, не придется ли признать, что и структуры \mathcal{M}_0 не существует? А если даже рассмотрение \mathcal{M}_0 законно, то верна ли в \mathcal{M}_0 , например, аксиома выбора? В современной теоретико-множественной математике накоплено большое количество проблем, которые упорно не поддаются решению обычными теоретико-множественными средствами, т. е. их не удастся ни доказать, ни опровергнуть. Происходит ли это просто в силу трудности проблемы и недостаточности приложенных усилий, или же рассматриваемую проблему принципиально нельзя ни доказать, ни опровергнуть?

Решение такого рода проблем упирается в трудность изучения семантики рассматриваемых теорий, т. е. в трудность изучения способа понимания формул теории. Семантика богатых математических теорий, таких, как математический анализ, теория множеств и др., по необходимости является недостаточно ясной, носит отчасти философский характер. Вместо описания конкретной модели теории в таких случаях часто приходится ограничиваться лишь формулировкой *аксиом-семантических соглашений*, которым должна удовлетворять наша теория. Для изучения богатых математических теорий со сложной семантикой с успехом может быть применен *метод формализации Гильберта*. Метод состоит в том, что на основании семантических соглашений содержательной теории \mathcal{T} строится формальная аксиоматическая теория T . При этом мы стремимся, чтобы аксиомы и правила вывода T были согласованы с семантическими требованиями \mathcal{T} . Тогда формулы, выводимые по формальным правилам теории T , оказываются содержательно истинными с точки зрения теории \mathcal{T} и отражают, таким обра-

зом, по крайней мере некоторый фрагмент содержательной теории \mathcal{T} . Желательно при этом, конечно, чтобы этот фрагмент был достаточно обширным и охватывал все интересующие нас черты теории \mathcal{T} . Формальную теорию T можно затем подвергнуть точному математическому исследованию и, таким образом, судить о семантике исходной неформальной теории.

Ключевым обстоятельством является здесь то, что для понимания отношения формальной выводимости $T \vdash A$ нет необходимости вникать в, может быть, очень сложную семантику теории \mathcal{T} : для установления $T \vdash A$ достаточно построить некоторое дерево вывода, т. е. описать простой синтаксический объект, составленный из строчек символов, расположенных по строго определенным правилам. Отношение $T \vdash A$ можно описать, как правило, уже в весьма элементарной теории, например в *формальной арифметике* Ag , теории, имеющей дело лишь с натуральными числами.

Формализация теорий позволяет уточнить наши семантические проблемы в виде некоторых уже синтаксических утверждений о формальных теориях.

Теория T в языке Ω называется *непротиворечивой*, если не существует предложения (т. е. формулы без параметров в языке Ω), такой, что $T \vdash A$ и $T \vdash \neg A$.

Упражнение. Покажите, что теория T непротиворечива тогда и только тогда, когда существует формула языка Ω , не выводимая в T .

Теория T называется *полной*, если для всякого предложения A в языке Ω имеем $T \vdash A$ или $T \vdash \neg A$.

Будем говорить, что предложение A *совместно с теорией T* , если из непротиворечивости теории T следует и непротиворечивость теории $T+A$, полученной добавлением к теории T формулы A в качестве новой нелогической аксиомы.

Упражнение. Докажите, что A совместно с T тогда и только тогда, когда из непротиворечивости теории T следует, что неверно $T \vdash \neg A$.

Предложение называется *независимыми от теории T* , если оба предложения A и $\neg A$ совместны с T .

Упражнение. Докажите, что предложение A независимо от теории T тогда и только тогда, когда из непротиворечивости теории T следует, что в T не выводимо ни предложение A , ни предложение $\neg A$.

2. Сформулируем теперь формальную аксиоматическую теорию Цермело—Френкеля. Это теория в языке первого порядка ZF , т. е. в языке без сложных термов. Саму аксиоматическую теорию мы также будем обозначать через ZF .

Опишем нелогические аксиомы ZF . Они формулируются параллельно семантическим соглашениям § 2, п. 2. Если в формулировке аксиомы ниже присутствуют параметры, то, как обычно, следует считать, что аксиомой является замыкание формулы кванторами общности по всем параметрам. Нелогиче-

ская аксиома теории есть всегда предложение, замкнутая формула этой теории.

1) Аксиома объемности (экстенциональности):

$$x = y \wedge x \in z \supset y \in z.$$

2) Аксиома пустого множества:

$$\exists u \neg z (z \in u).$$

3) Аксиома пары:

$$\exists u \forall z (z \in u \equiv (z = x \vee z = y)).$$

4) Аксиома суммы:

$$\exists u \forall z (z \in u \equiv (\exists v \in x) (z \in v)).$$

5) Аксиома множества подмножеств (аксиома степени):

$$\exists u \forall z (z \in u \equiv (z \subseteq x)).$$

6) Аксиома бесконечности:

$$\exists u (\forall z (\forall x (x \notin z) \supset z \in u) \wedge (\forall z \in u) (\forall v (\forall x (x \in v \equiv (x \in z \vee x = z)) \supset v \in u))).$$

7) Аксиома выделения:

$$\exists u \forall z (z \in u \equiv (z \in x \wedge \varphi(z))),$$

здесь $\varphi(z)$ — произвольная формула языка ZF, не содержащая свободно переменной u .

8) Аксиома подстановки (аксиома замены):

$$\exists u \forall z (z \in u \equiv (\exists x \in v) (\varphi(x, z) \wedge \forall w (\varphi(x, w) \supset z = w))),$$

здесь формула $\varphi(x, z)$ не содержит свободно переменных u и v .

9) Аксиома фундирования (аксиома регулярности):

$$\exists z (z \in x) \supset (\exists z \in x) \neg \exists u (u \in z \wedge u \in x).$$

Формулировка теории ZF закончена.

Аксиомы 2)–8) выражают существование тех множеств, которые непосредственно изображаются термами языка ZF^+ . Более точно, аксиома 6) говорит, что существует некоторое прогрессивное множество. Существование ω может быть уже затем доказано с помощью аксиомы выделения.

Нелогические аксиомы ZF выбраны таким образом, чтобы в полученной теории можно было вывести все формулы, содержательную истинность которых мы установили в предыдущих параграфах. Точнее, выводить следует, конечно, не сами формулы ZF^+ , а их переводы в язык ZF согласно § 4, п. 4: если в предыдущих параграфах для некоторой формулы φ языка ZF^+ мы утверждали φ , то теперь мы можем установить $ZF \vdash \varphi^0$. Проведение всех таких выводов является длинным, но

вполне тривиальным упражнением в применении правил техники естественного вывода исчисления предикатов. Мы не будем на этом останавливаться, но надеемся, что читатель приобрел самостоятельно некоторый опыт в формальных выводах.

Аксиома фундирования утверждает, что в некотором смысле все множества построены исходя из пустого множества. Она не используется при выводе математических утверждений, не использовали и мы ее, но она упрощает строение универсума множеств и обычно включается в состав нелогических аксиом ZF.

Следствием аксиомы фундирования является отсутствие «самопринадлежащих» множеств, а именно

не существует множеств таких, что

$$x \in x,$$

$$x \in y \wedge y \in x,$$

$$x \in y \wedge y \in z \wedge z \in x.$$

▷ Допустим, например, что существуют множества x и y такие, что $x \in y$ и $y \in x$. Рассмотрим множество $Q = \{x, y\}$. Q непусто и в то же время не существует элемента Q , не пересекающегося с Q , что противоречит аксиоме фундирования. □

Подобным образом можно установить с помощью аксиомы фундирования, что не существует последовательности множеств, «убывающих по принадлежности», т. е. функции f , $\text{dom } f = \omega$, такой, что $f(n+1) \in f(n)$, для всякого $n \in \omega$.

Следствием аксиомы фундирования является также следующий принцип *индукции по принадлежности*:

$$\forall x ((\forall y \in x) \varphi(y) \supset \varphi(x)) \supset \forall x \varphi(x).$$

▷ Напомним коротко доказательство этой схемы. Допустим

$$\forall x ((\forall y \in x) \varphi(y) \supset \varphi(x))$$

и предположим противное, т. е. что для некоторого x , $\neg \varphi(x)$. Рассмотрим функцию f , $\text{dom } f = \omega$, такую, что $f(0) = \{x\}$ и $f(n+1) = \bigcup_{n \in \omega} f(n)$.

Если $u \in z$, то $u \in f(n)$ для некоторого n , и тогда $u \in f(n+1)$ и, значит, $u \in z$. Таким образом, $(\forall u \in z) (u \in z)$ (это свойство называется *транзитивностью* множества z). Кроме того, очевидно, $x \in z$, так как $x \in f(0)$. Пусть $z' = \{u \in z \mid \neg \varphi(u)\}$. Множество z' не пусто, так как $x \in z'$. По аксиоме фундирования существует $x' \in z'$ такое, что $x' \cap z' = \emptyset$. Если $y \in x'$, то $y \notin z'$, т. е. по определению z' имеем $y \notin z \vee \varphi(y)$. Но первый член этой дизъюнкции не имеет места, так как $y \in x' \wedge x' \in z$, а множество z транзитивно. Таким образом, $\varphi(y)$. Мы установили $(\forall y \in x') \varphi(y)$. По допущению отсюда $\varphi(x')$. Но, с другой стороны, $x' \in z'$, что влечет $\neg \varphi(x')$, и мы приходим к противоречию. □

Часто к теории ZF добавляют еще и аксиому выбора. Точнее, если через AC обозначить точную формулировку аксиомы

выбора, приведенную в § 5, п. 4, то к ZF в качестве новой нелогической аксиомы следует присоединить формулу AC^0 . Полученную теорию обозначают через ZFC.

3. Теория ZFC исключительно богата по своим выразительным возможностям. Практически любая доказанная математическая теорема может быть записана на языке ZF и выведена в теории ZFC. В то же время известные выводы парадоксов теории множеств не проходят в ZFC.

В естественном смысле теория ZF содержит все формальные аксиоматические теории, рассмотренные нами в [1]. Уточним этот факт, например, по отношению к *формальной арифметике* Ag. Напомним, что язык Ag содержит один сорт переменных для натуральных чисел, константу 0, функциональные символы $x+y$, $x \cdot y$, Sx для соответствующих арифметических операций. В качестве атомарных формул используется лишь равенство термов $t=r$.

Изобразим теперь элементы языка Ag в языке ZF^+ , а затем с помощью стандартного перевода перейдем в язык ZF. Константу 0 языка Ag можно изобразить в виде замкнутого терма \emptyset языка ZF^+ . Переменные для натуральных чисел языка Ag изображаются в виде подчиненных переменных, подчиненных условию $x \in \omega$. Функциональные символы языка Ag также выражаются соответствующими термами ZF^+ , например, терму Sx языка Ag соответствует терм $x \cup \{x\}$ языка ZF^+ .

Таким образом, всякой замкнутой формуле A языка Ag соответствует некоторая замкнутая формула A' языка ZF^+ , полученная путем замены элементов языка Ag на соответствующие элементы теоретико-множественного языка. Наконец, полученную формулу A' можно перевести в язык ZF с помощью перевода, указанного в § 4, п. 4. Далее, нетрудно проверить, что если A — замкнутая формула Ag и $Ag \vdash A$, то $ZF \vdash A'^0$. Это доказывается с помощью громоздкой, но вполне тривиальной индукции по построению выводов в теории Ag.

Полученный перевод и является естественной *интерпретацией* теории Ag в теории ZF. Это уточнение интуитивно ясной идеи, что теория множеств содержит арифметику. Подобным образом можно построить в теории ZF интерпретации и других рассматривавшихся нами в [1] теорий. Теория ZF содержит в этом смысле все остальные изученные нами теории. Заметим, что понятие интерпретации является чисто синтаксическим: описывается формальное преобразование формул одного языка в формулы другого и доказывается, что это преобразование сохраняет выводимость. Нет никакой надобности вникать в семантику формул языка Ag или языка ZF.

Это дает возможность чисто синтаксически доказывать результаты об *относительной непротиворечивости* теорий. Так, если непротиворечива теория ZF, то непротиворечива и теория Ag. В самом деле, если Ag противоречива, то найдется предложение A такое, что $Ag \vdash A \wedge \neg A$. Но тогда $ZF \vdash A'^0 \wedge \neg A'^0$,

так что и теория ZF также оказывается противоречивой. Подобным образом из непротиворечивости ZF следует непротиворечивость и других рассматривавшихся нами теорий. С точки зрения оснований математики важно, что такое сведение по непротиворечивости остальных теорий к ZF происходит без обращения к семантике теорий и само может быть сформулировано в очень скромной теории, например в Ag .

4. Теперь мы можем дать уточненную формулировку независимости аксиомы выбора и континуум-гипотезы от теории множеств.

Теорема. *Аксиома выбора (т. е. формула AC^0) не зависит от теории ZF.*

Теорема. *Континуум-гипотеза не зависит от теории ZFC.*

Эти замечательные результаты были получены К. Геделем и П. Козеном. А именно в 1939 году Гедель показал совместимость аксиомы выбора и континуум-гипотезы с теорией ZF, построив замечательную интерпретацию ZFC с континуум-гипотезой в теории ZF, а в 1963 году Коэн показал совместимость отрицания аксиомы выбора с ZF и совместимость отрицания континуум-гипотезы с ZFC, также указав некоторые конкретные интерпретации. В настоящее время установлена независимость от ZF и от ZFC многих интересных теоретико-множественных утверждений. Доказательства этих результатов можно найти в книгах [5, 12, 13].

Обсудим в связи с вышеупомянутыми теоремами еще раз «наивный» вопрос, можно ли, например, считать континуум-гипотезу H истинным утверждением и в каком, собственно, смысле? Из вышеупомянутых результатов следует, что *если теория ZF непротиворечива*, то в ZFC нельзя вывести ни формулу H , ни формулу $\neg H$. Так как теория ZFC содержит все традиционно употребляемые средства доказательства математических утверждений, то отсюда следует, что H нельзя ни доказать, ни опровергнуть *традиционными математическими средствами*. Набор ZFC традиционных математических средств доказательства оказывается существенно неполным.

Но может быть можно предложить более мощные теории, в рамках которых вопрос относительно H решался бы уже определенным образом? Ответ тривиален. Конечно, можно. Из вышеуказанных результатов следует, что если ZF непротиворечива, то непротиворечива и теория $ZFC + H$, полученная путем добавления к ZFC утверждения H в качестве новой нелогической аксиомы. Непротиворечива также и теория $ZFC + \neg H$. Проблема состоит, таким образом, в том, какую из этих двух теорий предпочесть и на каких основаниях. В настоящее время не видно достаточных оснований для выбора одной из этих теорий в качестве «действительно правильной», и практические математики стараются в своих рассуждениях не использовать ни H , ни $\neg H$. Впрочем, некоторые авторитеты, например П. Коэн

[12, гл. IV, § 13], считают, что более естественно рассматривать континуум-гипотезу как ложное утверждение.

Можно подойти к проблеме и с иной точки зрения. В параграфах 2—5 мы рассматривали некую воображаемую структуру \mathcal{M}_0 объектов, удовлетворяющих аксиомам ZF. Так вот, истинно утверждение H в \mathcal{M}_0 или нет? Заметим, что все, что нам требовалось от \mathcal{M}_0 — это чтобы в ней выполнялись аксиомы ZF. Никоткуда не следует, что все такие структуры изоморфны, вполне возможно, что в одной модели ZFC будет истинно H , а в другой — будет истинно утверждение $\neg H$. Именно так дело и обстоит. Если ZF непротиворечива, то непротиворечива и каждая из теорий ZFC+ H и ZFC+ $\neg H$. По известной теореме Геделя о полноте каждая из этих теорий имеет модель и любую из этих двух моделей можно взять в качестве структуры \mathcal{M}_0 . С точки зрения оснований математики полезно отметить, что само доказательство теоремы Геделя о полноте в рассматриваемом случае не требует мощных теоретико-множественных идей, заложенных в аксиомах ZF, оно может быть естественно формализовано, например, в рамках арифметики второго порядка Ar_2 (см. [3]).

5. Обсудим еще один вопрос, связанный с теорией ZF и часто вызывающий недоразумения. Из теоремы Геделя о полноте следует, в частности, что если теория ZF непротиворечива, то она допускает счетную модель M . Как согласовать это обстоятельство с тем очевидным фактом, что в ZF выводимо существование несчетных множеств (например, таковым является множество $P\omega$)?

Модель M теории ZF имеет вид $\langle X, \varepsilon \rangle$, где X — непустое множество объектов модели и ε — отношение, $\varepsilon \subseteq X \times X$, которое интерпретирует принадлежность \in на множестве X . При этом все аксиомы ZF истинны в этой интерпретации.

Формула $\exists x(x=P\omega)$ истинна в M , и поэтому найдется объект $a \in X$ такой, что $M \models (a=P\omega)$. (Здесь и ниже мы пишем для краткости формулы ZF^+ , а не ZF. Имеется в виду, конечно, что их следует перевести в язык ZF с помощью стандартного перевода § 4, п. 4.)

Формула, утверждающая, что x счетно, имеет вид

$$\text{Count}(x) \equiv \exists f((1-1)(f) \wedge \text{dom } f = x \wedge \text{rng } f = \omega).$$

В ZF выводимо, что $P\omega$ — несчетное множество, и, значит, этот факт истинен в M , т. е.

$$M \models \neg \text{Count}(a).$$

По определению истинности в структуре последнее означает, что не существует объекта $b \in X$ такого, что

$$M \models (1-1)(b) \wedge \text{dom } b = a \wedge \text{rng } b = \omega,$$

т. е. среди элементов X не существует объекта b такого, что в M было бы истинно, что b есть взаимно-однозначная

функция с областью определения a и областью значений ω . Какой природы само a , при этом совсем не важно, например, a может быть с внешней точки зрения счетным множеством, так что существует биекция a на ω . Просто среди элементов X не существует биекции с точки зрения модели M . Истинность утверждений зависит от того, в какой модели эти утверждения рассматриваются. Один и тот же объект, будучи элементом различных моделей, в этих моделях может обладать различными свойствами.

Указанное явление обычно называется *парадоксом Сколема*. Следует иметь в виду, конечно, что никакого парадокса в собственном смысле этого слова здесь нет. Имеется, может быть, лишь некоторая путаница понятий, сбивающая неспециалиста.

Подобное недоразумение часто связано и с категоричностью некоторых структур, например натурального ряда.

В ZF можно доказать с помощью хорошо известных из теории чисел рассуждений, что все структуры Пеано, т. е. структуры вида $\langle \omega^*, 0^*, S^* \rangle$, где ω^* — множество, $0^* \in \omega^*$ и S^* — операция $S^*: \omega^* \rightarrow \omega^*$, удовлетворяющие аксиомам Пеано (см. § 4, п. 1), изоморфны, т. е. структуры Пеано изображают единственный натуральный ряд с точностью до изоморфизма. В этом и состоит свойство категоричности натурального ряда.

Тем не менее в различных моделях могут быть и неизоморфные натуральные ряды! Рассмотрим две модели теории ZF: $M_1 = \langle X_1, \varepsilon_1 \rangle$ и $M_2 = \langle X_2, \varepsilon_2 \rangle$. Имеем $ZF \vdash \exists x (x = \omega)$, так что существуют объекты $\omega_1 \in X_1$ и $\omega_2 \in X_2$ такие, что $M_1 \models \omega_1 = \omega$ и $M_2 \models \omega_2 = \omega$. Можно рассмотреть и «множество натуральных чисел» в каждой из моделей:

$$N_1 = \{a \in X_1 \mid M_1 \models a \varepsilon_1 \omega_1\},$$

$$N_2 = \{a \in X_2 \mid M_2 \models a \varepsilon_2 \omega_2\}.$$

В каждой из моделей M_i имеется свой объект 0_i и своя операция S_i прибавления единицы. Но структура $\langle N_i, 0_i, S_i \rangle$ совсем не обязана удовлетворять аксиомам Пеано с внешней точки зрения. Необходимо только, чтобы в модели M_i были истинны соответствующие формулы. Структуры $\langle N_i, 0_i, S_i \rangle$ могут быть вовсе не изоморфны содержательно понимаемому натуральному ряду и не изоморфны между собой. Структура, относительно которой в модели истинно, что она есть структура Пеано, может и не быть структурой Пеано с внешней точки зрения. Мы увидим далее, что если теория ZF непротиворечива, то она действительно имеет модели с неизоморфными натуральными рядами.

6. В параграфах 2—4 мы рассматривали принципы ZF как содержательно истинные, имеющие место в некоторой структуре \mathcal{M}_0 , и пытались показать, что основные математические понятия можно выразить в языке ZF, а многие математические

утверждения имеют место в \mathcal{M}_0 . Средства доказательства при этом никак не уточнялись.

Затем мы определили формальную аксиоматическую теорию ZF, в которой имеется возможность получить наши предыдущие результаты в форме точных выводов. Для построения выводов в теории ZF нет необходимости вникать в семантику языка ZF, но для изучения *самой* теории ZF, например для получения результатов о независимости, также нужна, конечно, некоторая математика. Эта математика вновь будет использоваться содержательным, неформализованным образом. По отношению к изучаемой теории ее называют *метаматематикой*. В принципе исследователь может заинтересоваться и метаматематикой теории и также попытаться ее формализовать.

Какова же должна быть метаматематика теории? С точки зрения оснований математики есть стремление сделать метаматематику как можно более бедной, с тем чтобы рассуждения в ней были максимально простыми и убедительными. В такой ситуации можно сказать, что результаты, относящиеся к очень сложной теории, получены с помощью очень убедительных приемов доказательства. Так, результаты, касающиеся независимости аксиомы выбора и континуум-гипотезы, могут быть естественно формализованы в рамках арифметики \mathcal{A}_γ так, что в этих доказательствах вовсе не будет фигурировать понятие множества. Многие другие результаты математической логики могут быть получены при ограниченном использовании теории множеств, например в рамках арифметики второго порядка.

С другой стороны, за пределами оснований математики математическую логику можно развивать как обычную математическую дисциплину, подобную топологии или теории функций. В такой ситуации нет оснований как-то специально ограничивать метаматематику. Можно, например, в качестве метаматематики взять *саму теорию* ZF. Так и поступают во многих исследованиях по теории моделей, нестандартному анализу и т. п. В этом смысле теория ZF выступает в роли «всеобщего мира множеств» для всех математиков.

Следует подчеркнуть, что теория ZF отнюдь не является единственно возможной подходящей базой для развития математики на точной основе. Ее широкое использование объясняется значительными формальными удобствами, а также отчасти исторической традицией. В настоящее время в логике разработаны очень интересные и многообещающие подходы к развитию математических теорий, не опирающиеся на понятие множества. Одним из них является *конструктивное направление в математике*, широко разрабатываемое в нашей стране. Кроме того, имеются и другие теоретико-множественные системы, могущие с успехом конкурировать с теорией ZF. Среди них прежде всего можно назвать *теорию типов* Рассела и Уайтхеда, а также *теорию множеств Куайна*.

Согласно известной теореме Геделя, непротиворечивость ZF не может быть установлена средствами ZF, т. е. практически не может быть установлена традиционными математическими средствами. Поэтому, если мы желаем исследовать непротиворечивость таких мощных теорий, приходится находить математические средства, с одной стороны, достаточно убедительные в том или ином отношении, а с другой стороны, выходящие за пределы традиционной математической практики.

Подробное обсуждение многих из затронутых здесь вопросов читатель может найти в книге [11].

7. Сделаем еще несколько замечаний относительно аксиоматического метода в математической логике. Термин «аксиоматический метод» используется в различных смыслах, что иногда ведет к недоразумениям.

Прежде всего это *содержательно-аксиоматический метод*. Он употребляется, когда изучается род структур, удовлетворяющих одному и тому же списку свойств. Например, один род структур составляют группы, другой род структур — кольца, третий род структур — структуры Пеано и т. п. Под *аксиомами* при этом понимаются просто конкретные условия, которым должна удовлетворять любая из структур изучаемого рода. Эти условия понимаются содержательно и записываются на рабочем математическом языке, например на русском или английском. Впрочем, часто аксиомы какого-либо рода структур записывают и на точном логико-математическом языке, но понимают содержательно как утверждения о структурах; для одних структур эти аксиомы могут быть истинны, а для других — ложны.

Как понимать эти аксиомы, зависит от принятой метаматематики. Особенно это существенно в тех случаях, когда в формулировках аксиом, кроме объектов структуры, фигурируют и другие объекты, например множества. Это так называемые *неэлементарные аксиомы*.

Так, среди аксиом для структуры Пеано: $\langle \omega, 0, S \rangle$ имеется и такая: для всякого подмножества $x \subseteq \omega$ выполняется принцип полной математической индукции

$$0 \in x \wedge (\forall y \in x) (Sy \in x) \supset (\forall y \in \omega) (y \in x).$$

Чтобы понимать это утверждение, следует понимать, что такое подмножество данного множества и по каким правилам следует с этими подмножествами обращаться. Если мы в качестве метаматематики принимаем содержательно понимаемые принципы ZF (так это и делается в практической математике), то понимание этой аксиомы в достаточной степени проясняется, можно получать содержательные следствия из нее, например доказать, что все структуры Пеано изоморфны. Можно представлять себе ситуацию и так, что мы фиксировали некоторую воображаемую структуру \mathcal{M}_0 , согласованную с принципами ZF. Только из \mathcal{M}_0 мы и черпаем все объекты математического ис-

следования. После этого можно доказать, что все структуры Пеано в \mathcal{M}_0 изоморфны между собой.

Несколько иной подход возникает при изучении теорий *формально аксиоматическим методом*. Основные факты об изучаемых структурах мы оформляем в виде формальной аксиоматической теории T . Новые факты о теории T извлекаются (по крайней мере, в принципе) с помощью аппарата формальной выводимости. Для их получения первоначальная семантика T не имеет никакого значения, и нет необходимости привлекать модели теории T . Более того, среди моделей формальной аксиоматической теории T могут оказаться и такие, которые мы не включали в число подразумеваемых моделей. Так, если теория ZF непротиворечива, то среди ее моделей имеется и счетная модель.

Под *аксиомами* теории T теперь понимаются просто ее нелогические аксиомы, т. е. формулы специального вида, смысл этих аксиом в принципе не важен при формулировке самой теории T . Но, конечно, если мы стремимся выводить в T нетривиальные факты, следует снабдить T достаточно богатым списком аксиом. Так, если мы изучаем структуры Пеано, записывая принцип индукции на языке ZF, следует добавить в теорию и аксиомы, описывающие свойства множеств.

§ 1. МАШИНЫ ТЬЮРИНГА

1. Известно, какое большое значение в математике имеют алгоритмы. Коротко говоря, алгоритм есть точное предписание, в соответствии с которым по любому входному объекту из данного класса входных объектов можно эффективно получать выходные объекты. Например, алгоритм умножения десятичных рациональных чисел «столбиком» позволяет по двум десятичным записям получить некоторую третью согласно хорошо известному предписанию. Алгоритмами являются схема Горнера для вычисления значений многочлена, алгоритм Евклида для отыскания наибольшей общей меры двух отрезков, алгоритм перемножения двух квадратных матриц и многие другие.

Чтобы подвергнуть понятие алгоритма точному анализу, следует несколько сузить тот класс алгоритмов, которые мы намерены рассматривать. Прежде всего мы предполагаем, что наши алгоритмы представляют собой *дискретные* предписания, подобные программам для вычислительных машин, а не подобные аналоговым или графическим вычислительным устройствам. Предписание должно содержать указания к выполнению конечного числа элементарных, четко различимых шагов. Поэтому и входные данные изучаемых алгоритмов должны представлять собой дискретные, *конструктивные объекты*, такие как натуральные числа, рациональные числа, целочисленные матрицы и т. п. Входными данными не могут быть, например, произвольные отрезки прямой или произвольные действительные числа, так как их, в общем случае, нельзя задавать конечным образом.

Заметим, что в математике заметную роль играют и точные предписания, перерабатывающие неконструктивные объекты. Такого рода алгоритмы также изучаются в математической логике, но для первоначального изучения мы ограничимся самым важным классом чисто дискретных устройств. Таким образом, алгоритм умножения десятичных рациональных чисел, схема Горнера для многочленов с рациональными коэффициентами, алгоритм перемножения квадратных матриц с натуральными элементами суть алгоритмы нашего класса, а алгоритм отыскания наибольшей общей меры двух произвольных отрезков остается пока вне нашего рассмотрения.

Понятие конструктивного объекта все еще очень расплывчато, но разумно предположить, что интересующие нас конструктивные объекты можно эффективно кодировать в виде слов в некотором алфавите. Таким образом, можно считать, что с каждым алгоритмом A связан некоторой алфавит Σ — *внешний алфавит* A , и наш алгоритм получает на вход слова в алфавите Σ и вырабатывает в качестве результатов также слова в алфавите Σ . Мы будем изучать только такие *словарные* алгоритмы.

Алгоритм является единой инструкцией, на вход которой можно подавать любое из, вообще говоря, бесконечного списка входных слов. В этом состоит свойство *массовости* алгоритма. Удобно считать, что на вход алгоритму A можно подать *любое* слово во внешнем алфавите Σ . Получив входное слово, следует применять к нему предписание алгоритма A шаг за шагом. При этом шаги предписания могут обрываться и мы получим выходное слово как результат работы алгоритма, а может случиться, что предписание ведет к бесконечной последовательности элементарных действий, и тогда алгоритм A не определен на данном входном слове.

Далее, мы считаем, что изучаемые алгоритмы обладают свойством *детерминированности*: каждый элементарный шаг однозначно определяется предыдущей ситуацией. В наш алгоритм не встроены случайные или вероятностные механизмы. В частности, если дважды проделать вычисления алгоритма над одним и тем же входом, то результаты также будут одинаковы.

Наконец, наши алгоритмы должны обладать свойством *замкнутости*, т. е. выполнение вычислений определяется только предписанием, и для вычисления не требуется привлекать, кроме входного слова, каких-либо процессов или вычисляющих устройств извне.

Можно предположить, что все действия, которые может производить любой алгоритм указанного типа, можно разложить на некоторые канонические элементарные шаги. Такой анализ привел Тьюринга (в 1936 году) к понятию вычислительных машин, названных впоследствии его именем.

2. Перейдем к точным определениям. Напомним, что *алфавит* есть по определению непустое конечное множество символов, элементы алфавита называются его *буквами*. Слово в алфавите Σ есть конечная последовательность (может быть и пустая) его букв. Слово в алфавите Σ имеет, следовательно, вид $a_0 a_1 \dots a_n$, где $a_i \in \Sigma$. Множество всех слов в алфавите Σ обозначим через Σ^* .

Основная операция на словах — операция *приписывания* слова к слову: если дано слово A , имеющее вид $a_0 a_1 \dots a_n$, и слово B вида $b_0 b_1 \dots b_m$, то можно образовать новое слово AB вида $a_0 a_1 \dots a_n b_0 \dots b_m$, полученное *приписыванием* (в другой терминологии — *соединением*, *конкатенацией*) слов A и B .

Пустое слово обозначается через Λ . Конечно, $A\Lambda = \Lambda A = A$. Фиксируем два алфавита Σ и S . Σ назовем *внешним алфавитом*, а S — назовем *внутренним алфавитом*, или *алфавитом состояний*. Предположим, что символы \rightarrow , R , L не входят ни в Σ , ни в S .

Командой назовем слово одного из следующих трех видов:

$$qa \rightarrow rb$$

$$qa \rightarrow rbR$$

$$qa \rightarrow rbL$$

(где $q, r \in S$; $a, b \in \Sigma$). Команды, подобно формулам языка, можно читать по-русски. Команда первого вида читается так: «находясь в состоянии q и наблюдая букву a , следует перейти в состояние r и напечатать букву b ». Команда второго вида: «находясь в состоянии q и наблюдая букву b , следует перейти в состояние r , напечатать букву b и затем передвинуться вправо». Команда третьего вида читается так же, как и команда второго вида, но следует только в конце читать «...и затем передвинуться влево».

Список команд или *программа* (в алфавитах Σ , S) есть по определению конечная последовательность команд.

Пусть теперь фиксированы алфавиты Σ , S , а также две буквы $q_0, q_1 \in S$ и одна буква $a_0 \in \Sigma$. Мы называем q_0 — *начальным состоянием*, q_1 — *финальным* или *заключительным* состоянием, а букву a_0 мы назовем *бланком* (*пустой клеткой*).

Алфавит $S \times \Sigma$ мы назовем алфавитом *наблюдаемых букв*: про букву $\langle q, a \rangle \in S \times \Sigma$ мы говорим, что a *наблюдается* в состоянии q .

Конфигурацией на ленте (или *машинным словом*) называется слово в алфавите $\Sigma \cup (S \times \Sigma)$, содержащее в точности одно вхождение наблюдаемой буквы. Таким образом, конфигурация всегда имеет вид $A \langle q, a \rangle B$, где $A, B \in \Sigma^*$ и буква a наблюдается в состоянии q .

Конфигурация называется *начальной*, если она имеет вид $\langle q_0, a \rangle B$. Здесь q_0 — начальное состояние, а слово A в данном случае пусто. Конфигурация называется *финальной* (или *заключительной*), если она имеет вид $A \langle q_1, a \rangle B$, т. е. наблюдаемая буква наблюдается в заключительном состоянии.

Пусть теперь K_1 и K_2 — конфигурации и k — команда. Определим, что значит, что команда k *переводит* конфигурацию K_1 в K_2 ; символически мы записываем это отношение как

$$k : K_1 \triangleright K_2.$$

А именно пусть K_1 имеет вид $A \langle q, a \rangle B$. Если левая часть команды k не имеет вида qa , то $k : K_1 \triangleright K_2$ автоматически считается ложным (команда k *неприменима* к конфигурации K_1). Пусть левая часть команды k имеет вид qa . Далее разберем три случая в зависимости от строения команды k .

1) k есть $qa \rightarrow rb$. В этом случае $k:K_1 \triangleright K_2 \Leftrightarrow K_2$ имеет вид $A \langle r, b \rangle B$.

2) k есть $qa \rightarrow rbR$. Здесь рассмотрим два случая:

2a) слово B не пусто, $B = cB'$, тогда $k:K_1 \triangleright K_2 \Leftrightarrow K_2$ есть $Ab \langle r, c \rangle B'$;

2b) слово B пусто, тогда $k:K_1 \triangleright K_2 \Leftrightarrow K_2$ есть $Ab \langle r, a_0 \rangle$;

3) k есть $qa \rightarrow rbL$. Здесь также рассмотрим два подслучая:

3a) слово A не пусто, $A = A'c$, тогда

$k:K_1 \triangleright K_2 \Leftrightarrow K_2$ есть $A' \langle r, c \rangle bB$;

3b) слово A пусто, тогда

$k:K_1 \triangleright K_2 \Leftrightarrow K_2$ есть $\langle r, a_0 \rangle bB$.

Легко видеть, что если $k:K_1 \triangleright K_2$, то конфигурация K_2 определяется однозначно по команде k и конфигурации K_1 .

Можно представлять себе, что конфигурация есть слово, записанное на некоторой ленте, разделенной на клеточки, в каждой клеточке записано по букве алфавита Σ . Над одной из букв записано еще состояние, эта буква наблюдается машиной.

Перевести K_1 в K_2 командой k — это значит «выполнить» команду k , проделать то, что «она требует». После выполнения команды наблюдаемой может стать уже иная буква левее или правее исходной. При этом, если приходится выходить за пределы ленты влево и вправо, то выполнение команды автоматически предусматривает добавление новой клеточки, на которой считается напечатанной буква a_0 — «пустая клеточка», «бланк».

Машина Тьюринга есть по определению набор

$$M = \langle \Sigma, S, \Pi, q_0, q_1, a_0 \rangle,$$

Σ — внешний алфавит,

S — внутренний алфавит,

Π — программа машины,

q_0 — начальное состояние,

q_1 — заключительное состояние,

a_0 — бланк.

Команда программы Π машины M обычно записывают вертикально, нумеруя

$$\begin{array}{c} k_1 \\ k_2 \\ \vdots \\ k_m \end{array}$$

сверху вниз.

Мы говорим, что машина M переводит конфигурацию K_1 в K_2 , и пишем $M:K_1 \triangleright K_2$, если

1) существует команда k_i программы машины M такая, что $k_i:K_1 \triangleright K_2$ и

2) для всех $j < i$ не существует конфигурации K_3 такой, что $k_j:K_1 \triangleright K_3$.

Таким образом, если $M:K_1 \triangleright K_2$, то конфигурация K_2 определяется однозначно по M и K_1 . Конечно, вполне может оказаться, что некоторая конфигурация K_1 не переводится машиной M ни в какую конфигурацию K_2 .

Протокол вычислений машины Тьюринга M есть (конечная или бесконечная) последовательность конфигураций

$$K_0, K_1, \dots, K_n, \dots$$

такая, что

- 1) K_0 — начальная конфигурация;
- 2) $M:K_i \triangleright K_{i+1}$;
- 3) если конфигурация K_i входит в протокол и $M:K_i \triangleright K_{i+1}$, причем K_i не заключительная конфигурация, то K_{i+1} также входит в протокол вычислений (т. е. протокол не оканчивается конфигурацией K_i , если вычисления можно продолжить дальше);

4) протокол вычислений может содержать не более одной заключительной конфигурации, и если протокол действительно содержит заключительную конфигурацию, то этот протокол конечен и заключительная конфигурация есть последний его член.

Мы пишем $M \sqsubset K_0 \sqsupset = K_n$, если K_0 — начальная конфигурация, K_n — заключительная конфигурация и существует протокол вычислений машины M , начинающийся с K_0 и оканчивающийся K_n . Для данных M и K_0 протокол вычислений, если он существует, может быть только один, так что результат вычислений K_n , если он существует, определен однозначно. Мы говорим, что M определена на K_0 , если соответствующая заключительная конфигурация K_n существует.

Машина может быть не определена на начальной конфигурации K_0 по двум причинам: либо потому, что в процессе построения протокола вычислений мы приходим к конфигурации, к которой не применима ни одна команда машины, либо потому, что протокол вычислений бесконечен и не приводит к заключительной конфигурации. Разумеется, по данной машине M и конфигурации K_0 отнюдь не видно сразу, будет ли M определена на K_0 , и если нет, то какой из вышеупомянутых случаев будет иметь место.

Если дано слово $A \in \Sigma^*$, то по нему можно стандартным образом изготовить некоторую начальную конфигурацию A^0 для машины M , как говорят, *подать слово A на вход* машины M . А именно если A не пусто, $A = aA'$, то следует взять $A^0 = \langle q_0, a \rangle A'$. Если же A пусто, то $A^0 = \langle q_0, a_0 \rangle$. Коротко говоря, следует «заставить» машину наблюдать первую букву A в начальном состоянии. Если же в A вовсе нет букв, то, конечно, следует заставить наблюдать пустую клеточку машины.

С другой стороны, по любой конфигурации K машины мы можем изготовить некоторое слово $K^1 \in \Sigma^*$. Мы говорим, что слово K^1 *записано на ленте* в конфигурации K . А именно надлежит поступить следующим образом:

1) следует, во-первых, убрать состояние из наблюдаемой буквы, т. е. заменить наблюдаемую букву $\langle q, a \rangle$ в конфигурации K на обычную букву a , так что получится некоторое слово A в алфавите Σ ;

2) затем следует в слове A стереть все пустые клетки, идущие подряд, слева и справа, т. е. если $A = BCD$, где B и D составлены только из буквы a_0 , а слово C не начинается и не кончается буквой a_0 , то следует оставить в качестве K^1 слово C .

Может оказаться, что K^1 есть пустое слово (если в K фигурировала лишь буква a_0 из Σ).

3. Таким образом, машина Тьюринга M с внешним алфавитом Σ порождает функцию, перерабатывающую некоторые слова в алфавите Σ в слова же в алфавите Σ . А именно для $A, B \in \Sigma^*$, $M(A) = B \Leftrightarrow$ существуют конфигурации K_0 и K_n такие, что $M \vdash K_0 \vdash K_n$ и $(K_0)^0 = A$, $(K_n)^1 = B$. Коротко говоря, если на вход подать слово A , то соответствующий протокол вычислений заканчивается и на ленте оказывается записанным слово B .

Заметим, что функция M может быть определена отнюдь не на всех словах в алфавите Σ . Мы говорим, что машина M *определена* на слове A , и пишем $M(A)$, если M определена на начальной конфигурации A^0 .

Машины Тьюринга можно представлять себе как определенный тип вычислительных машин, обрабатывающих слова в алфавитах и имитирующих действие реальных вычислительных машин или действия человека-вычислителя. Машина работает на ленте, разделенной на клеточки—ячейки. В каждый момент времени рассматривается лишь конечный кусок этой ленты, но по мере надобности лента продолжается неограниченно в обе стороны.

Машина может печатать на ленте некоторые буквы алфавита и обозрывать в каждый момент только одну клеточку ленты. Ее действия происходят шагами, однозначным образом определяемыми ее состоянием в данный момент, содержимым обозреваемой ячейки и программой машины. Машина останавливается, когда приходит в заключительное состояние. Может оказаться, что она, стартуя с некоторого слова, никогда не остановится.

4. Определим теперь, что значит, что функция вычислима с помощью машины Тьюринга.

Напомним некоторые обозначения. Если X_1, \dots, X_k суть множества, то через $X_1 \times \dots \times X_k$ мы обозначаем их декартово произведение, т. е. множество всех упорядоченных наборов $\langle x_1, \dots, x_k \rangle$, где $x_1 \in X_1, \dots, x_k \in X_k$. Если X и Y — множества, то через $X \rightarrow Y$ мы обозначим множество всех функций с областью определения X , принимающих значения в множестве Y . Вместо $f \in (X \rightarrow Y)$ традиционно пишем $f: X \rightarrow Y$. Функция от k аргументов есть функция вида $f: X_1 \times \dots \times X_k \rightarrow Y$, так что $f(\langle x_1, \dots,$

$x_k\}) \in Y$. Обозначение $f(\langle x_1, \dots, x_k \rangle)$, по традиции, сокращаем до $f(x_1, \dots, x_k)$.

Через $X \rightarrow Y$ обозначим теперь множество всех *частичных* функций из X в Y , т. е. $f: X \rightarrow Y \Leftrightarrow$ область определения f включена в X (и не обязательно совпадает со всем множеством X) и значения f принимаются из множества Y . В частности, функция $f: X_1 \times \dots \times X_k \rightarrow Y$ от k аргументов может быть и не определена для некоторых наборов $\langle x_1, \dots, x_k \rangle$. Множеству $X_1 \times \dots \times X_k \rightarrow Y$ принадлежит, в частности, и нигде не определенная функция.

Нас специально интересуют функции, перебатывающие слова. Если Δ — алфавит, то через Δ^* мы обозначаем множество всех слов в алфавите Δ . Пусть $\Delta_1, \dots, \Delta_k, \Delta$ суть алфавиты. Машины Тьюринга можно приспособить для вычисления функций вида $\Delta_1^* \times \dots \times \Delta_k^* \rightarrow \Delta^*$. С этой целью фиксируем некоторую новую букву q , не входящую в рассматриваемые алфавиты, и будем изображать наборы слов из $\Delta_1^* \times \dots \times \Delta_k^*$ в виде слов в алфавите $\Delta_1 \cup \dots \cup \Delta_k \cup \{q\}$. Например, упорядоченную тройку $\langle x_1, x_2, x_3 \rangle$, где $x_i \in \Delta_i^*$, изобразим в виде одного слова $x_1 q x_2 q x_3$.

Пусть $f: \Delta_1^* \times \dots \times \Delta_k^* \rightarrow \Delta^*$ есть (вообще говоря, частичная) функция от k аргументов. Будем говорить, что машина Тьюринга

$$M = \langle \Sigma, S, \Pi, q_0, q_1, a_0 \rangle$$

вычисляет f , если

- 1) $\Delta \cup \Delta_1 \cup \dots \cup \Delta_k \cup \{q\} \subseteq \Sigma$;
- 2) для всяких x_1, \dots, x_k , $x_i \in \Delta_i^*$, набор $\langle x_1, \dots, x_k \rangle$ принадлежит области определения функции $f \Leftrightarrow !M(x_1 q \dots q x_k)$ и $M(x_1 q \dots q x_k) \in \Delta^*$;
- 3) для всяких x_1, \dots, x_k , $x_i \in \Delta_i^*$, если $!M(x_1 q \dots q x_k)$ и $M(x_1 q \dots q x_k) \in \Delta^*$, то $f(x_1, \dots, x_k) = M(x_1 q \dots q x_k)$;
- 4) для удобства потребуем еще, чтобы бланк a_0 машины M был отличен от всех букв алфавита $\Delta \cup \dots \cup \Delta_k \cup \Delta \cup \{q\}$.

Подчеркнем, что внешний алфавит машины M , вычисляющей функцию f , может быть существенно шире алфавита $\Delta \cup \Delta_1 \cup \dots \cup \Delta_k \cup \{q\}$. Как говорят, машина M — *над алфавитом* $\Delta \cup \Delta_1 \cup \dots \cup \Delta_k \cup \{q\}$. Мы требуем лишь, чтобы M вела себя «как функция f » лишь на входах $x_1 q \dots q x_k$. Буквы внешнего алфавита Σ , не входящие в $\Delta \cup \dots \cup \Delta_k \cup \{q\}$, могут использоваться при обработке слов вида $x_1 q \dots q x_k$, но только в промежуточных конфигурациях протокола вычислений. Функция $f: \Delta_1^* \times \dots \times \Delta_k^* \rightarrow \Delta^*$ называется *вычислимой (по Тьюрингу)*, если существует машина Тьюринга, вычисляющая f .

5. Рассмотрим теперь функции от натуральных чисел $f: \omega \times \dots \times \omega \rightarrow \omega$. Для таких функций также естественно вводит-

ся понятие вычислимости. Будем изображать натуральные числа в виде слов в алфавите $\{|\}$:

0 1 2 3 4 5 ...

\emptyset \top $||$ $|||$ $||||$ $|||||$...

Иногда, чтобы отличать так изображаемые натуральные числа от «обычных» теоретико-множественно понимаемых натуральных чисел (например, по фон Нейману), слова из палочек называются *нумералами* соответствующих натуральных чисел.

Нумерал числа n обозначают через \bar{n} , например $\bar{3} = |||$

Тогда каждая функция на натуральных числах, скажем, $f: \omega \times \omega \rightarrow \omega$, естественно изображается некоторой функцией $\bar{f}: \{|\}^* \times \{|\}^* \rightarrow \{|\}^*$ на словах из палочек

$$f(m_1, m_2) = m_3 \Leftrightarrow \bar{f}(\bar{m}_1, \bar{m}_2) = \bar{m}_3.$$

Числовая функция называется *вычислимой* по Тьюрингу, если вычислима по Тьюрингу соответствующая функция на словах из палочек.

Функции $f: \omega \times \dots \times \omega \rightarrow \omega$, вычислимые по Тьюрингу, называются еще *частично-рекурсивными функциями* (термин «частично» напоминает о том, что f не обязательно определена на всем множестве $\omega \times \dots \times \omega$). Если же частично-рекурсивная функция f от k -аргументов определена на всех наборах натуральных чисел $\langle n_1, \dots, n_k \rangle$, т. е. $f: \omega \times \dots \times \omega \rightarrow \omega$, то она называется *общерекурсивной* или даже просто *рекурсивной* функцией.

Коротко говоря, частично-рекурсивная функция — это функция, для которой существует вычисляющий ее алгоритм, заданный в виде машины Тьюринга. Для общерекурсивной функции этот алгоритм для любых входных натуральных чисел обязательно заканчивает свою работу и выдает значение функции.

Можно показать, что все обычные функции теории чисел типа $x+y$, xy , x^y , $[\sqrt{x}]$ и т. п. являются рекурсивными, не всюду определенные функции, например x/y , $x-y$ $[\lg(x-y)]$ являются частично-рекурсивными. Следующая машина M вычисляет функцию $f(m, n) = m + 2n$. Внешний алфавит M есть $\{|\, q, a_0\}$, где a_0 — бланк. Внутренний алфавит $\{q_0, q_1, q_2, q_3, q_4, q_5\}$, где q_0 — начальное, а q_1 — финальное состояние. Программа машины имеет вид:

1. $q_0| \rightarrow q_2|$
2. $q_0q \rightarrow q_2q$
3. $q_2| \rightarrow q_2|R$
4. $q_2q \rightarrow q_2qR$
5. $q_2a_0 \rightarrow q_3a_0L$
6. $q_3| \rightarrow q_4a_0L$
7. $q_4| \rightarrow q_4|L$

8. $q_4q \rightarrow q_4qL$

9. $q_4a_0 \rightarrow q_5|L$

10. $q_5a_0 \rightarrow q_2|$

11. $q_3q \rightarrow a_0q_1$

Протокол вычислений можно кратко описать следующим образом. Пусть начальная конфигурация имеет вид $\langle q_0| \rangle |||q||$. Под действием команд 1, 2 машина переходит в состояние q_2 : $\langle q_2| \rangle |||q||$ и затем под действием 3 и 4 движется вправо до появления пустой клеточки справа $|||q|| \langle q_2, a_0 \rangle$. После этого машина двигается влево и переходит в состояние q_3 : $|||q \langle q_3, | \rangle$ (команда 5). Затем машина стирает обозреваемую палочку, переходит в состояние q_4 и в этом состоянии движется влево до появления пустой клеточки слева (команды 6, 7, 8): $\langle q_4, a_0 \rangle |||q|$. Под действием команд 9 и 10 машина печатает слева две палочки и переходит вновь в состояние q_2 : $\langle q_2| \rangle |||q|$. Далее вновь начинают работать команды 3 и 4 и цикл повторяется, машина идет в конец слова, стирает после q одну палочку, идет влево и печатает слева две палочки и т. д., пока правее вовсе не останется палочек. Тогда возникнет конфигурация $|||q| \langle q_3, q \rangle$. Команда 11 сотрет q , и машина остановится в заключительной конфигурации $|||q| \langle q_1, a_0 \rangle$.

Упражнение. Постройте машину Тьюринга, вычисляющую функцию $f(m, n) = m \cdot n$.

§ 2. ТЕЗИС ЧЕРЧА

1. Кажется очевидным, что всякая функция на словах $f: \Sigma^* \rightarrow \Sigma^*$, вычислимая по Тьюрингу, эффективно вычислима. В самом деле, алгоритм для вычисления такой функции и задается собственно машиной Тьюринга, о которой идет речь в определении вычислимости.

В 1936 году Черч выдвинул тезис о том, что *всякая эффективно вычисляемая функция является вычислимой по Тьюрингу*. Точнее говоря, Черч использовал не понятие вычислимости по Тьюрингу, а некоторое иное точное понятие вычислимости (понятие лямбда-определимости), математически эквивалентное нашему определению вычислимости. В свете анализа, проведенного в начале предыдущего параграфа, тезис Черча можно сформулировать и так: всякий словарный, дискретный, массовый, детерминированный, замкнутый алгоритм может быть задан в виде машины Тьюринга.

Коротко говоря, машины Тьюринга могут имитировать любой алгоритм указанного типа.

Сразу отметим, что тезис Черча не является математическим утверждением, так как в его формулировку входит понятие алгоритма в интуитивном смысле этого слова. Тезис Черча нельзя поэтому доказать или опровергнуть в рамках традиционной математической практики. Это не математический ре-

зультат, а скорее естественнонаучное наблюдение. В пользу тезиса Черча можно привести, однако, сильные доводы естественно-научного характера.

Во-первых, за многие столетия развития в математике накопилось огромное количество алгорифмов. Все они оказались вычислимыми по Тьюрингу. Этот экспериментальный материал должен убедить нас в справедливости тезиса Черча не меньше, чем, например, соответствующие эксперименты убеждают физиков в законе сохранения энергии.

Во-вторых, почти одновременно с понятием вычислимости по Тьюрингу были предложены и другие подходы к понятию вычислимости, внешне сильно отличающиеся друг от друга. Все они оказались эквивалентными. Упомянем важнейшие из этих подходов:

исчисление равенств Эрбрана и Геделя,
частично-рекурсивные функции по Клини,
лямбда-определимость Черча,
канонические системы Поста,
нормальные алгорифмы Маркова,
алгорифмы Колмогорова—Успенского.

В-третьих, эффективно вычислимые функции с интуитивной точки зрения должны быть замкнуты относительно некоторых важных операций: композиции, разветвления, итерации и т. д. Оказывается, что семейство по Тьюрингу функций, как и ожидается, замкнуто относительно этих операций.

Наконец, отметим, что тезис Черча *не нужен* с чисто математической точки зрения и, собственно, в математических утверждениях никогда не применяется. Там, где нужна вычислимость в математическом рассуждении, мы всегда можем использовать вычислимость по Тьюрингу.

Тезис Черча важен для приложений математики в естествознании. Он объясняет ту большую роль, которую играет понятие алгорифма в точной форме (т. е., например, в форме вычислимости по Тьюрингу) в современной математике.

Например, иногда удается точно доказать, что некоторая конкретная функция $f: \Sigma^* \rightarrow \Sigma^*$ не вычислима по Тьюрингу. Тезис Черча указывает, что бесполезно было бы фактически искать алгорифм (в интуитивном смысле этого слова), вычисляющий f .

Функция f и интуитивно будет невычислимой.

Таким образом, открывается ценная возможность точными математическими средствами обнаружить невычислимость некоторых функций.

§ 3. РЕКУРСИВНЫЕ И РЕКУРСИВНО-ПЕРЕЧИСЛИМЫЕ МНОЖЕСТВА И ПРЕДИКАТЫ

1. В предыдущем параграфе мы дали определение вычислимой функции. Теперь попытаемся определить, что понимать под вычислимым предикатом. При этом следует рассматри-

вать, конечно, *словарные* предикаты $P(x_1, \dots, x_k)$, где x_i суть слова в некотором алфавите, например, x_1 пробегает слова в алфавите Δ_1 , x_2 — слова в алфавите Δ_2 и т. д. Про такой предикат P мы говорим, что он типа $\Delta_1^* \times \dots \times \Delta_k^*$, указывая область пробегания каждого аргументного места.

Словарному предикату P можно сопоставить некоторую функцию $\chi_P: \Delta_1^* \times \dots \times \Delta_k^* \rightarrow \{0, 1\}$, так называемую *характеристическую функцию* предиката, такую, что для всех наборов $\langle x_1, \dots, x_k \rangle$, где $x_i \in \Delta_i^*$:

$P(x_1, \dots, x_k)$ истинно $\Leftrightarrow \chi_P(x_1, \dots, x_k) = 1$. С другой стороны, словарному предикату можно сопоставить *множество-отношение* данного предиката, а именно множество $X_P \subseteq \Delta_1^* \times \dots \times \Delta_k^*$ такое, что

$$P(x_1, \dots, x_k) \text{ истинно} \Leftrightarrow \langle x_1, \dots, x_k \rangle \in X_P.$$

Напротив, если задано множество слов $X \subseteq \Sigma^*$, то можно рассмотреть *предикат принадлежности* этого множества: для всякого слова $x \in \Sigma^*$

$$P(x) \text{ истинно} \Leftrightarrow x \text{ принадлежит } X.$$

Чаше всего рассматривают словарные предикаты, для которых алфавиты $\Delta_1, \dots, \Delta_k$ совпадают. Например, предикаты, определенные на натуральных числах (*арифметические предикаты*), можно трактовать как словарные предикаты, определенные на нумералах, т. е. словах в алфавите $\{\mid\}$.

2. Наше первое определение вычислимости таково: предикат P типа $\Delta_1^* \times \dots \times \Delta_k^*$ *рекурсивен* (или *разрешим*), если вычислима по Тьюрингу соответствующая характеристическая функция χ_P .

Интуитивно предикат P рекурсивен, если существует алгоритм, выясняющий для каждого набора $\langle x_1, \dots, x_k \rangle$ слов, истинно $P(x_1, \dots, x_k)$ или нет.

Повторим еще раз определение рекурсивного предиката непосредственно в терминах машин Тьюринга. Предикат P типа $\Delta_1^* \times \dots \times \Delta_k^*$ называется рекурсивным (или разрешимым), если существует машина Тьюринга $M = \langle \Sigma, S, \Pi, q_0, q_1, a_0 \rangle$ такая, что

$$1) \Delta_1 \cup \dots \cup \Delta_k \cup \{0, 1\} \cup \{q\} \subseteq \Sigma$$

(здесь, как и раньше, q — новая буква, которая служит для записи наборов слов);

2) для всяких x_1, \dots, x_k ($x_i \in \Delta_i^*$) определено значение $M(x_1 q \dots q x_k)$, и это значение равно 0 или 1;

3) $P(x_1, \dots, x_k)$ истинно тогда и только тогда, когда $M(x_1 q \dots q x_k) = 1$.

Множество $X \subseteq \Sigma^*$ назовем *рекурсивным*, если рекурсивен соответствующий предикат принадлежности $x \in X$, где x пробегает множество Σ^* .

Обычные арифметические предикаты $x \leq y$, $x + y = z$ « x — простое число», « x делит y » являются рекурсивными. Аккурат-

ное доказательство этого факта состоит в непосредственном построении соответствующих разрешающих машин Тьюринга. Но и без такого построения разрешимость этих предикатов должна быть очевидной читателю. В самом деле, очевидно, что существует эффективный метод проверки, верно или нет, например, что x делит y для натуральных x и y . В дальнейшем изложении мы будем обычно опускать конкретное построение необходимых машин Тьюринга в случаях, когда ясно, что нужный алгоритм существует. Методы построения машин Тьюринга рассматриваются на семинарских занятиях, о них можно прочесть и в более подробных учебниках математической логики (см. список литературы).

Аналогично, рекурсивными являются многие обычные множества натуральных чисел: множество четных чисел, множество простых чисел, множество чисел, делящихся на 7. Число называется совершенным, если оно равно сумме своих делителей (отличных от него самого). Множество всех нечетных совершенных чисел рекурсивно, так как есть простой способ выяснять по данному числу, является ли оно нечетным и совершенным. Тем не менее неизвестно, пусто это множество или нет, так как до сих пор неизвестно, существуют ли указанные числа.

3. Второе определение вычислимости — *рекурсивная перечислимость* предиката накладывает меньшие требования на соответствующий алгоритм.

Интуитивно, предикат P рекурсивно перечислим, если существует процедура, позволяющая устанавливать истинность $P(x_1, \dots, x_k)$ в случае, когда $P(x_1, \dots, x_k)$ истинно. Если же $P(x_1, \dots, x_k)$ ложно, то наша процедура иногда будет это устанавливать, а иногда процесс будет продолжаться неограниченно. Таким образом, проверяя данный набор x_1, \dots, x_k на истинность, мы не полностью уверены, что получим ответ.

Точное определение таково: предикат P типа $\Delta_1^* \times \dots \times \Delta_k^*$ называется *рекурсивно-перечислимым*, если существует машина Тьюринга $M = \langle \Sigma, S, \Pi, q_0, q_1, a_0 \rangle$ такая, что

- 1) $\Delta_1 \cup \dots \cup \Delta_k \cup \{0, 1\} \cup \{q\} \subseteq \Sigma$;
- 2) для всяких $x_1, \dots, x_k (x_i \in \Delta_i^*)$, если определено значение $M(x_1 q \dots q x_k)$, то это значение равно 0 или 1;
- 3) $P(x_1, \dots, x_k)$ истинно тогда и только тогда, когда $M(x_1 q \dots q x_k) = 1$.

Множество $X \subseteq \Sigma^*$ назовем *рекурсивно-перечислимым*, если рекурсивно-перечислим соответствующий предикат принадлежности.

4. Непосредственно из определений видно, что *всякий рекурсивный предикат (множество) является и рекурсивно-перечислимым*.

Лемма. Если предикат $P(x_1, \dots, x_k, y)$ рекурсивно-перечислим, то рекурсивно-перечислим и предикат $\exists y P(x_1, \dots, x_k, y)$.

▷ Пусть y пробегает множество Δ^* и пусть M — машина,

фигурирующая в определении рекурсивной перечислимости предиката $P(x_1, \dots, x_k, y)$. Убедимся, что существует соответствующая машина M' для предиката $\exists y P(x_1, \dots, x_k, y)$. Мы не будем выписывать программу M' , а лишь опишем в содержательных терминах, как M' работает. Мы надеемся, что после этих пояснений построение M' не составит принципиальных затруднений (хотя останется довольно громоздким делом). Пересчитаем все слова в алфавите Δ в виде определенной эффективно заданной последовательности

$$Y_1, Y_2, Y_3, \dots$$

Рассмотрим набор x_1, \dots, x_k и опишем процесс получения значения $M'(x_1q \dots qx_k)$.

Вычисление происходит этапами. На первом этапе машина M' образует набор $x_1q \dots qx_kY_1$ и делает один шаг в вычислении значения $M(x_1q \dots qx_kqY_1)$. Если за один шаг обнаружилось $M(x_1q \dots qx_kqY_1) = 1$, то работа M' заканчивается, и полагаем $M'(x_1q \dots qx_k) = 1$.

На втором этапе (если на первом этапе работа не закончилась) машина M' производит уже два шага в вычислении двух значений:

$$M(x_1q \dots qx_kqY_1)$$

и

$$M(x_1q \dots qx_kqY_2).$$

Если за два шага вычислений M обнаружится, что хоть одно из значений равно единице, то работу M' заканчиваем и полагаем $M'(x_1q \dots qx_k) = 1$.

На третьем этапе (если на предыдущих этапах работа не закончилась) машина M' производит уже три шага в вычислении трех значений:

$$M(x_1q \dots qx_kqY_1),$$

$$M(x_1q \dots qx_kqY_2),$$

$$M(x_1q \dots qx_kqY_3).$$

Если за три шага вычислений M обнаружится, что хоть одно из этих значений равно единице, то работу M' заканчиваем и полагаем $M'(x_1q \dots qx_k) = 1, \dots$. И так далее, выполняем этап за этапом.

Если на некотором этапе вычислится значение $M(x_1q \dots qx_kqY_j) = 0$, то мы вычеркиваем набор $x_1q \dots qx_kqY_j$ из списка подвергаемых испытанию, но из-за этого набора работу не заканчиваем, а переходим к следующему этапу в поисках набора, для которого $M(x_1q \dots qx_kqY_j) = 1$. Машина M' и будет подтверждать рекурсивную перечислимость $\exists y P(x_1, \dots, x_k, y)$.

Заметим, что M' принимает лишь значение 1 или не определена. Поэтому даже если $P(x_1, \dots, x_k, y)$ — рекурсивный предикат, машина M' подтверждает лишь рекурсивную перечислимость $\exists y P(x_1, \dots, x_k, y)$. И действительно, как мы увидим ниже,

может быть построен рекурсивный предикат P такой, что предикат $\exists yP$ уже только рекурсивно-перечислим, но не рекурсивен. \square

Лемма. Для всякого рекурсивно-перечислимого предиката $Q(x_1, \dots, x_k)$ существует рекурсивный предикат $P(x_1, \dots, x_k, y)$ такой, что

$$Q(x_1, \dots, x_k) \Leftrightarrow \exists yP(x_1, \dots, x_k, y).$$

\triangleright Пусть M — машина, фигурирующая в определении рекурсивной перечислимости Q . Рассмотрим следующий предикат: $P(x_1, \dots, x_k, y)$ истинно $\Leftrightarrow y$ есть конечный протокол вычислений машины M такой, что в начальной конфигурации y на ленте записан набор $x_1q \dots qx_k$, а в конечной конфигурации y на ленте записана единица. Конечные протоколы M мы рассматриваем как слова в подходящем алфавите Δ так, что $y \in \Delta^*$.

Можно убедиться, что предикат P разрешим. Интуитивно очевидно, что по данному слову y можно эффективно выяснить, действительно ли y является протоколом вычислений машины M и что записано на ленте в различных конфигурациях протокола y . Аккуратное доказательство, которое мы опустим, состоит в громоздком построении соответствующей распознающей машины.

Теперь по свойству машины M имеем

$$Q(x_1, \dots, x_k) \Leftrightarrow \exists yP(x_1, \dots, x_k, y). \quad \square$$

Последняя лемма оправдывает термин «перечислимый» в отношении рекурсивно-перечислимого предиката Q . Чтобы убедиться в истинности $Q(x_1, \dots, x_k)$, следует последовательно перечислять все слова Y_1, Y_2, \dots в некотором алфавите Δ и последовательно проверять на истинность рекурсивный предикат P :

$$P(x_1, \dots, x_k, Y_1), P(x_1, \dots, x_k, Y_2), \dots$$

Если Q истинен, то этот процесс закончится и мы найдем соответствующее Y_i . Если же Q ложно, то процесс будет продолжаться неограниченно и мы можем так и не узнать, что Q ложно.

5. Отметим еще, что применение логических связок $\wedge \vee \supset \neg$ к рекурсивным предикатам дает рекурсивные же предикаты.

Доказательство состоит в непосредственном построении распознающих машин для сложных предикатов по данным машинам составляющих предикатов. С интуитивной точки зрения метод распознавания истинности, например предиката $P(x_1, \dots, x_k) \vee Q(x_1, \dots, x_k)$ очевиден, если известен метод распознавания истинности для P и Q .

Что касается рекурсивно-перечисливых предикатов, то применение связок $\wedge \vee$ к ним ведет вновь к рекурсивно-перечислимым предикатам. Однако в общем случае это уже неверно для связок \supset и \neg . Позже мы обсудим пример рекурсивно-

перечислимого предиката, отрицание которого не рекурсивно перечислимо.

Если рассмотреть отношения, соответствующие предикатам, то из предыдущих замечаний следует, что объединение и пересечение любых двух рекурсивных (рекурсивно-перечислимых) множеств вновь рекурсивно (рекурсивно-перечислимо). Дополнение рекурсивного множества слов в некотором алфавите (до множества всех слов в этом алфавите) вновь рекурсивно. Но дополнение рекурсивно-перечислимого множества может быть и не рекурсивно-перечислимо.

В языке арифметики Аг естественно определить

$$x \leq y \Leftrightarrow \exists z (x + z = y), \\ x < y \Leftrightarrow x \leq y \wedge \neg (x = y).$$

Если дан арифметический предикат $P(x, y_1, \dots, y_n)$, то можно определить новые предикаты, полученные путем применения *ограниченных кванторов*:

$$(\forall x \leq z) P(x, y_1, \dots, y_n) \Leftrightarrow \forall x (x \leq z \supset P(x, y_1, \dots, y_n)); \\ (\exists x \leq z) P(x, y_1, \dots, y_n) \Leftrightarrow \exists x (x \leq z \wedge P(x, y_1, \dots, y_n)); \\ (\forall x < z) P(x, y_1, \dots, y_n) \Leftrightarrow \forall x (x < z \supset P(x, y_1, \dots, y_n)); \\ (\exists x < z) P(x, y_1, \dots, y_n) \Leftrightarrow \exists x (x < z \wedge P(x, y_1, \dots, y_n)).$$

Результирующие предикаты уже от аргументов z, y_1, \dots, y_n . Заметим

$$\neg (\forall x \leq z) P(x, y_1, \dots, y_n) \Leftrightarrow (\exists x \leq z) \neg P(x, y_1, \dots, y_n).$$

Упражнение. Укажите аналогичные законы для преобразования отрицаний остальных ограниченных кванторов.

Применение ограниченного квантора к рекурсивному (рекурсивно-перечислимому) предикату приводит к рекурсивному же (рекурсивно-перечислимому) предикату. Доказательство состоит в непосредственном построении соответствующих машин Тьюринга для результирующих предикатов по данным машинам Тьюринга.

6. Теорема Поста. Если предикаты P и $\neg P$ одновременно рекурсивно-перечислимы, то они необходимо оба и рекурсивны.

▷ Пусть P и $\neg P$ — предикаты типа $\Delta_1^* \times \dots \times \Delta_k^*$ и пусть M_1 и M_2 — машины Тьюринга, фигурирующие в определении рекурсивной перечислимости для P и $\neg P$ соответственно. Определим машину M' , распознающую истинность, например, предиката P . Как и раньше, мы не будем выписывать программу M' , а лишь опишем, как M' работает. Рассмотрим набор $\langle x_1, \dots, x_k \rangle$ и опишем процесс получения значения $M'(x_1 q \dots q x_k)$. Вычисление происходит этапами. На первом этапе машина M' производит один шаг в вычислении значений $M_1(x_1 q \dots q x_k)$ и $M_2(x_1 q \dots q x_k)$. На втором этапе производятся два

шага вычисления $M_1(x_1q \dots qx_k)$ и $M_2(x_1q \dots qx_k)$. На третьем этапе — три шага вычисления этих двух значений и так далее. Вычисление заканчивается, как только вычислится одно из значений $M_1(x_1q \dots qx_k)$ или $M_2(x_1q \dots qx_k)$. Если обнаружится, что $M_1=1$ или $M_2=0$, то положим $M'=1$. Если же окажется, что $M_1=0$ или $M_2=1$, то $M'=0$.

Из определения машин M_1 и M_2 следует, что не может быть одновременно $M_1=1$ и $M_2=1$ (это означало бы, что для набора $\langle x_1, \dots, x_k \rangle$ истинны одновременно P и $\neg P$). Также не может быть одновременно $M_1=0$ и $M_2=0$ (что означало бы, что набор $\langle x_1, \dots, x_k \rangle$ не удовлетворяется ни P , ни $\neg P$). В то же время необходимо $M_1(x_1q \dots qx_k)=1$ или $M_2(x_1q \dots qx_k)=1$, так как $P(x_1, \dots, x_k)$ или $\neg P(x_1, \dots, x_k)$. Отсюда следует, что алгоритм M' действительно распознает истинность предиката P . \square

7. Упомянем теперь предварительно о некоторых результатах теории алгоритмов, относящихся собственно к логике.

Ясно, что формулы и термы такого логико-математического языка, как ZF^+ , можно трактовать как слова в подходящем алфавите Σ . В самом деле, как видно из определения гл. 1, § 2, п. 1, выражения ZF^+ строятся с помощью символов: $,$ (формальная запятая), $(,)$, \in , \wedge , \vee , \supset , \neg , \forall , \exists , $\{, \}$, $|$, \emptyset , ω , P , \cup , \leftrightarrow и бесконечного набора переменных. Но переменные в свою очередь, можно рассматривать как слова, составленные из двух символов v и $|$ и скобок, если положить $x \Leftarrow (v|)$, $y \Leftarrow (v||)$, $z \Leftarrow (v|||)$ и т. д.

Аналогично выражения языков Ag , Ag_2 , ZF также можно трактовать как слова в некоторых алфавитах. Это происходит потому, что выражения рассматриваемых языков строятся из конечного числа явно указанных символов по строго фиксированным правилам. Такие языки назовем *явно заданными*. Мы не будем давать точного математического определения явно заданного языка. Читатель, если угодно, под явно заданным языком может понимать просто один из конкретных логико-математических языков, описанных в книге [1] или в этой книге, хотя изложенные общие результаты верны и по отношению ко многим другим языкам. Следует иметь в виду, однако, что в принципе рассматриваются и логико-математические языки, выражения которых заданы некоторым косвенным теоретико-множественным способом. Для них представление выражений в виде слов некоторого алфавита может быть и невозможным. Таковы, например, языки, содержащие несчетное множество констант. Однако языки, используемые для построения формальных аксиоматических теорий, обычно оказываются явно заданными.

Таким образом, явно заданный язык Ω определяет алфавит Σ_Ω такой, что выражения Ω суть специальные слова в этом алфавите. Мы считаем, что множество Fm_Ω всех формул языка Ω и множество Tm_Ω всех термов языка Ω суть рекурсивные подмножества множества Σ_Ω^* всех слов алфавита Σ_Ω . По отноше-

нию к конкретному языку, например ZF^+ , это может быть строго доказано путем построения соответствующей распознающей машины Тьюринга. Впрочем, интуитивно соответствующая распознающая процедура должна быть ясной и без этого громоздкого построения: по каждому слову в алфавите языка всегда можно судить, является ли это слово формулой ZF^+ или не является.

Далее, множество Ax_Ω всех аксиом исчисления предикатов явно заданного языка Ω также является рекурсивным подмножеством Σ_α^* , так как по формуле всегда можно судить, имеет ли она вид аксиомы исчисления предикатов или нет (например, имеет ли рассматриваемая формула вид $A \supset (B \supset A)$).

Дерево формул в языке Ω мы определяли (см. введение) как некоторую двумерную фигуру, но должно быть ясно, что дерево формул явно заданного языка можно трактовать и как слово в некотором алфавите. Так, вместо

$$\frac{D_1, D_2}{A}$$

можно писать $(D_1, D_2/A)$ и т. п. Существенно при этом, что множество всех деревьев выводов является рекурсивным подмножеством множества всех слов этого алфавита, так как по слову можно эффективно выяснить, является ли оно деревом формул и выполняется ли в этом дереве необходимое структурное требование.

Заметим теперь, что множество нелогических аксиом в рассмотренных нами формальных аксиоматических теориях, таких, как Ag , $Ag2$, ZF , также является рекурсивным множеством, и это теории в явно заданных языках. Мы назовем такие аксиоматические теории *явно заданными*. Опять-таки не будем давать строгого математического определения явно заданной аксиоматической теории, читатель может считать, что это просто одна из рассмотренных нами теорий.

Фиксируем некоторый обширный алфавит Σ_0 такой, что выводы во всех явно заданных (интересующих нас) формальных аксиоматических теориях записываются в виде слов в алфавите Σ_0 .

Для явно заданной формальной аксиоматической теории T рассмотрим предикат $\text{Prf}_T(y, x)$, где x, y суть слова в алфавите Σ_0 . А именно $\text{Prf}_T(y, x)$ истинно тогда и только тогда, когда y есть дерево вывода теории T с нижней формулой x . Важный факт состоит в том, что это рекурсивный предикат. Для конкретной теории, например для ZF , это может быть аккуратно проверено, путем построения соответствующей машины Тьюринга, но процедура распознавания истинности этого предиката должна быть и без того ясной: по данному слову y следует сначала выяснить, является ли это слово выводом теории T , и затем нужно определить, верно ли, что нижней формулой этого вывода является формула x .

Для явно заданной теории T рассмотрим предикат $\text{Pr}_T(x)$, где x — слово в алфавите Σ_0 . А именно $\text{Pr}_T(x)$ истинно тогда и только тогда, когда x есть формула, выводимая в теории T . Очевидно,

$$\text{Pr}_T(x) \Leftrightarrow \exists y \text{Prf}_T(y, x).$$

Отсюда следует

Теорема. *Множество формул, выводимых в явно заданной формальной аксиоматической теории T , рекурсивно-перечислимо. В частности, рекурсивно-перечислимо множество формул, выводимых в исчислении предикатов явно заданного языка.*

▷ См. § 3, п. 4. □

Множество всех предложений, выводимых в теории T , обозначим через $[T]$. Для явно заданных теорий это — рекурсивно-перечислимое множество. Теория называется *разрешимой*, если это множество оказывается рекурсивным. Известно, что теории Ar , Ar_2 , ZF неразрешимы. Замечательно, что некоторые важные формальные аксиоматические теории разрешимы. Такова, например, элементарная теория действительных чисел (см. [5]).

Обозначим через $\text{Th}_{\text{Ar}}(\omega)$ множество всех предложений языка Ar , истинных в стандартной модели ω . Известно, что это множество не является даже рекурсивно-перечислимым.

В силу известной теоремы Геделя о полноте множество всех логических законов языка Ω совпадает с множеством формул, выводимых в исчислении предикатов этого языка. Таким образом, для явно заданного языка множество всех его логических законов оказывается рекурсивно-перечислимым. Для рассмотренных нами до сих пор языков множество логических законов не является рекурсивным. Но можно указать простые языки, для которых это множество рекурсивно. Таким например, будет язык с одним сортом переменных, без констант и функциональных символов и с конечным набором *одноместных* предикатов (так называемая *теория одноместных предикатов*).

§ 4. ПРИМИТИВНО-РЕКУРСИВНЫЕ ФУНКЦИИ, ГЕДЕЛЕВА НУМЕРАЦИЯ, АРИФМЕТИКА С ПРИМИТИВНО-РЕКУРСИВНЫМИ ТЕРМАМИ

1. Определим теперь важный класс общерекурсивных арифметических функций. Функции этого класса называются *примитивно-рекурсивными*.

С этой целью введем индуктивно понятие *примитивно-рекурсивного (п. р.) описания*. Каждое п. р. описание представляет собой слово вида (k, l, \dots) , где k, l — натуральные числа. Число l называется количеством аргументов описания, $l > 0$. Каждому п. р. описанию одновременно сопоставляется арифметическая функция — примитивно-рекурсивная функция, *имеющая данное описание*. Определение содержит семь пунктов,

первые три пункта составляют базис индукции, а последние представляют собой индуктивный шаг, показывая, как можно строить новые п. р. описания из уже имеющих.

$$1) (1, 1), F(x) = x + 1.$$

$$2) (2, 1), F(x) = x.$$

3) $(3, 1, m), F(x) = m$ для произвольного натурального числа m .

4) Пусть $l > 0, 1 \leq i_1, \dots, i_s \leq l$ и g — п. р. описание с s аргументами, причем g сопоставлена функция G . Тогда можно определить новое п. р. описание $(4, l, g, i_1, \dots, i_s)$, которому сопоставляется функция

$$F(x_1, \dots, x_l) = G(x_{i_1}, \dots, x_{i_s}).$$

Мы говорим, что F получена из G путем перестановки, отождествления и введения фиктивных аргументов.

5) Пусть g — п. р. описание с l аргументами, которому сопоставлена функция G , и h — п. р. описание с p аргументами, которому сопоставлена функция H . Тогда можно определить новое п. р. описание $(5, l+p-1, g, h)$, которому сопоставляется функция

$$F(y_1, \dots, y_p, x_2, \dots, x_l) = G(H(y_1, \dots, y_p), x_2, \dots, x_l).$$

Мы говорим, что F получена из G и H подстановкой.

6) Пусть g — п. р. описание с двумя аргументами, которому сопоставлена функция G , и m — произвольное натуральное число. Тогда можно образовать новое п. р. описание $(6, 1, g, m)$, которому сопоставляется функция F , удовлетворяющая следующим тождествам:

$$\begin{cases} F(0) = m, \\ F(x+1) = G(x, F(x)). \end{cases}$$

Мы говорим, что F получается из G и m с помощью примитивной рекурсии.

7) Пусть h — п. р. описание с m аргументами и g — п. р. описание с $m+2$ аргументами. Тогда можно определить новое п. р. описание $(7, m+1, g, h)$, которому сопоставляется функция F такая, что

$$\begin{cases} F(0, x_1, \dots, x_m) = H(x_1, \dots, x_m), \\ F(x+1, x_1, \dots, x_m) = G(x, F(x, x_1, \dots, x_m), x_1, \dots, x_m). \end{cases}$$

В этой ситуации мы также говорим, что F получена из функций G и H примитивной рекурсией.

Арифметическая функция называется примитивно-рекурсивной, если она сопоставлена по указанным выше правилам какому-либо п. р. описанию.

Упражнение. Обозначим через $+$ примитивно-рекурсивную функцию от двух аргументов, имеющую следующее п. р. описание:

$$(7, 2, (4, 3, (1, 1), 2), (2, 1)).$$

Убедитесь, что выполняются следующие тождества:

$$y+0=y;$$
$$y+(x+1)=(y+x)+1.$$

Постройте естественное п. р. описание для функции умножения двух натуральных чисел.

Совершенно ясно, что всякая п. р. функция является вычислимой, так как процесс определения такой функции с помощью примитивно-рекурсивного описания дает одновременно и способ вычисления значений функции для любых числовых значений ее аргументов. Точный математический факт состоит в том, что *всякая примитивно-рекурсивная функция — общерекурсивна*. Доказательство состоит в том, чтобы указать соответствующую машину Тьюринга для каждого примитивно-рекурсивного описания. Это построение осуществляется индукцией по длине рассматриваемого п. р. описания.

Обратное, как мы увидим дальше, неверно: существуют общерекурсивные функции, не являющиеся примитивно-рекурсивными. Тем не менее п. р. функции образуют весьма обширный класс вычислимых функций, замкнутый относительно многих естественных операций: все практически определяемые общерекурсивные функции, как правило, оказываются примитивно-рекурсивными.

Арифметический предикат $P(x_1, \dots, x_n)$ назовем *примитивно-рекурсивным*, если существует примитивно-рекурсивная функция $f(x_1, \dots, x_n)$, принимающая в качестве значений лишь 0 и 1, и такая, что

$$P(x_1, \dots, x_n) \Leftrightarrow f(x_1, \dots, x_n) = 1.$$

Мы часто будем опускать громоздкую проверку того, что те или иные конкретные функции или предикаты являются примитивно-рекурсивными. Подробные методы для такой проверки развиваются, например, в книгах [3, 4, 18].

2. С помощью примитивно-рекурсивных функций можно определить взаимно-однозначное соответствие между парами натуральных чисел и натуральными числами.

Точнее, можно определить три п. р. функции $j(x, y)$, $j_1(z)$, $j_2(z)$ такие, что выполняются следующие тождества:

$$j_1(j(x, y)) = x,$$
$$j_2(j(x, y)) = y,$$
$$j(j_1(z), j_2(z)) = z.$$

Например, можно определить

$$j(x, y) = (\max^2(x, y) + y) + (\max(x, y) - x).$$

Тогда дополнительно выполняются соотношения:

$$j(0, 0) = 0, \quad x \leq j(x, y), \quad y \leq j(x, y).$$

Далее, нумерация n -ок натуральных чисел при каждом n может быть введена, например, с помощью следующих примитивно-рекурсивных функций:

$$v_1(x) = x,$$

$$v_2(x_1, x_2) = j(x_1, x_2),$$

$$v_{n+1}(x_1, x_2, \dots, x_{n+1}) = j(x_1, v_n(x_2, \dots, x_n)),$$

для которых имеются соответствующие обратные п. р. функции

$$\delta_i^n(v_n(x_1, \dots, x_n)) = x_i,$$

где $n \geq 1, 1 \leq i \leq n$.

Наконец, можно ввести единое примитивно-рекурсивное взаимно-однозначное соответствие между n -ками натуральных чисел при всех n и натуральными числами. А именно для каждого натурального $n \geq 1$ определим

$$c(x_0, \dots, x_{n-1}) = j(n-1, v_n(x_0, \dots, x_{n-1})) + 1,$$

а пустому набору натуральных чисел, при $n=0$, сопоставим число 0.

Таким образом, натуральные числа в зависимости от способа кодирования можно отождествлять с парами, тройками и произвольными кортежами натуральных чисел.

Упомянем еще несколько примитивно-рекурсивных функций в связи с нашей нумерацией. Двухместная п. р. функция $x \bullet y$ соединения кортежей такова, что

$$c(x_0, \dots, x_{m-1}) * c(y_0, \dots, y_{n-1}) = c(x_0, \dots, x_{m-1}, y_0, \dots, y_{n-1}),$$

$$x * 0 = 0 * x = x.$$

Двухместная п. р. функция $[x]_z$ «высекает» элемент кортежа x : $[0]_z = 0, [c(x_0, x_1, \dots, x_{n-1})]_z = x_z$ при $z < n$ и $[c(x_0, x_1, \dots, x_{n-1})]_z = 0$ при $z \geq n$. Примитивно-рекурсивная функция $lh(x)$ определяет длину кортежа:

$$lh(0) = 0, lh(c(x_0, x_1, \dots, x_{n-1})) = n.$$

3. Мы намерены систематическим образом нумеровать натуральными числами слова в различных алфавитах, и прежде всего нас интересуют слова в алфавите Σ_0 (см. § 3, п. 7), который мы назовем *основным алфавитом*. Будем считать, что основной алфавит содержит буквы q и $|$, так что в нем можно естественно записывать наборы натуральных чисел, если отождествлять натуральные числа со словами в алфавите $|$.

Фиксируем бесконечную последовательность различных символов-букв: a_0, a_1, a_2, \dots , которую назовем *основной последовательностью букв*. Алфавит назовем *простым*, если он состоит из конечного числа букв основной последовательности. Простой алфавит назовем *приведенным*, если он содержит буквы основной последовательности лишь с нечетными номерами. Будем считать, что алфавит Σ_0 содержит ровно k букв и эти буквы встречаются в основной последовательности в качестве

первых k букв с нечетными номерами, т. е. алфавит Σ_0 состоит в точности из букв $a_1, a_3, \dots, a_{2k+1}$. Таким образом, основной алфавит является приведенным.

Пусть A — слово в простом алфавите, имеющее вид $a_{i_1} a_{i_2} \dots a_{i_m}$. Сопоставим этому слову натуральное число $\gamma A = c(i_1, \dots, i_m)$. Для пустого слова положим $\gamma A = 0$. Натуральное число γA назовем *гедделевым номером* слова A . Отображение γ устанавливает взаимно-однозначное соответствие между всеми словами в простых алфавитах и натуральными числами.

Далее мы желали бы нумеровать натуральными числами машины Тьюринга. нас интересуют главным образом машины, обрабатывающие слова в основном алфавите, т. е. вычисляющие функции типа $\Sigma_0^* \rightarrow \Sigma_0^*$, но некоторая сложность возникает из-за того, что такие машины могут содержать и другие буквы, используемые, например, в промежуточных вычислениях. Машину Тьюринга назовем *приведенной*, если ее внешний алфавит и алфавит ее внутренних состояний суть приведенные алфавиты.

Лемма. Для каждой машины Тьюринга M , вычисляющей функцию $f: \Sigma_0^ \rightarrow \Sigma_0^*$, может быть построена приведенная машина Тьюринга M' , вычисляющая ту же самую функцию.*

▷ Достаточно взаимно-однозначным образом заменить все буквы внутреннего и внешнего алфавитов машины M , отличные от букв основного алфавита, на некоторые буквы из основной последовательности с нечетными номерами. □

В частности, если $\Delta_1, \Delta_2, \dots, \Delta_h, \Delta$ суть алфавиты, включенные в основной алфавит Σ_0 и не содержащие буквы q , а машина M вычисляет функцию $f: \Delta_1^* \times \dots \times \Delta_h^* \rightarrow \Delta^*$, то может быть построена приведенная машина M' , вычисляющая ту же функцию.

Пусть теперь M — приведенная машина Тьюринга,

$$M = \langle \Sigma, S, \Pi, q_0, q_1, a_0 \rangle.$$

Можно трактовать тогда M как слово в простом алфавите. А именно будем считать, что буквы \rightarrow, R, L , участвующие в написании команд машины, входят в основную последовательность в качестве букв с четными номерами. Например, это есть соответственно буквы a_0, a_2, a_4 . Машина M как слово в простом алфавите получится, если написать подряд друг за другом следующие слова:

- 1) последовательность букв Σ (для определенности можно считать, что буквы записываются в порядке возрастания номеров в основной последовательности);
- 2) разделительная буква с четным номером (например, это может быть a_6);
- 3) последовательность букв S ;
- 4) разделительная буква a_6 ;
- 5) последовательность команд $\Pi, k_1 k_2 \dots k_m$;

6) разделительная буква a_6 ;

7) слово $q_0 q_1 a_0$.

Таким образом, каждую приведенную машину Тьюринга можно рассматривать как слово в простом алфавите, и, следовательно, каждая такая машина имеет определенный геделев номер.

Далее, конечные протоколы работы приведенных машин Тьюринга также очевидным образом можно трактовать как слова в простом алфавите и, следовательно, приписать таким протоколам геделевы номера.

4. Пусть $\Delta_1, \dots, \Delta_k, \Delta$ — алфавиты, включенные в основной алфавит и не содержащие разделительной буквы q . Пусть f — функция типа $\Delta_1^* \times \dots \times \Delta_k^* \rightarrow \Delta^*$. Определим тогда функцию $f^\vee: \omega \times \dots \times \omega \rightarrow \omega$, «моделирующую» работу функции на геделевых номерах. А именно:

1) если набор m_1, \dots, m_k натуральных чисел таков, что хоть одно из m_i не является геделевым номером слова алфавита Δ_i , то $f(m_1, \dots, m_k) = 0$;

2) для всякого набора слов A_1, \dots, A_k, A , где $A_i \in \Delta_i^*$, $A \in \Delta$, имеем $f(A_1, \dots, A_k) = A \Leftrightarrow f^\vee(\gamma A_1, \dots, \gamma A_k) = \gamma A$.

Аналогично для всякого словарного предиката P типа $\Delta_1^* \times \dots \times \Delta_k^*$ можно ввести моделирующий арифметический предикат P^\vee такой, что

1) если набор m_1, \dots, m_k таков, что хоть одно из натуральных чисел m_i не является геделевым номером слова алфавита Δ_i , то $P(m_1, \dots, m_k)$ ложно;

2) для всякого набора слов A_1, \dots, A_k , $A_i \in \Delta_i^*$: $P(A_1, \dots, A_k) \Leftrightarrow P^\vee(\gamma A_1, \dots, \gamma A_k)$.

Лемма. Функция f вычислима тогда и только тогда, когда вычислима функция f^\vee .

Предикат P рекурсивен (рекурсивно-перечислим) тогда и только тогда, когда рекурсивен (рекурсивно-перечислим) предикат P^\vee .

Громоздкое доказательство, состоящее в построении необходимых машин по исходным машинам, мы опускаем.

5. отождествим натуральные числа со словами в алфавите $|$. Тогда геделев номер натурального числа как слова отнюдь не совпадает с самим этим натуральным числом (за исключением нуля). Но соответствие между натуральным числом и его геделевым номером является примитивно-рекурсивным. Точнее, могут быть построены две одноместные примитивно-рекурсивные функции Gd и Gd^{-1} такие, что для всякого натурального числа m :

$$Gd(m) = \gamma m, \quad Gd^{-1}(\gamma m) = m.$$

6. Геделева нумерация открывает возможность говорить о сложных конструктивных объектах на языке арифметики. Мы хотели бы записывать различные свойства этих объектов на языке Ag , а затем доказывать их в формальной аксиоматиче-

ской теории Аг. Основная трудность при осуществлении этой программы состоит в том, что в теории Аг мало функций — лишь сложение и умножение. Поэтому мы формулируем более выразительную аксиоматическую теорию Ag^+ , в которой, грубо говоря, имеется функциональный символ для каждой примитивно-рекурсивной функции.

Язык Ag^+ содержит один сорт переменных x, y, z, \dots , константу 0 и для каждого примитивно-рекурсивного описания p от l аргументов — функциональный символ от l аргументов. Для конкретности можно считать, что само п. р. описание p , как слово в алфавите $(, , , , |$, и является функциональным символом Ag^+ . В частности, в Ag^+ имеются и все функциональные символы Аг (см. введение). Например, S имеет п. р. описание $(1, 1)$, $+$ имеет п. р. описание $(7, 2, (4, 3, (1, 1), 2), (2, 1))$.

Термы языка Ag^+ строятся последовательно из переменных и константы 0 с помощью функциональных символов. Точнее, индуктивное определение термина языка Ag^+ состоит из следующих трех пунктов:

- 1) всякая переменная есть терм;
- 2) константа 0 есть терм;
- 3) если f — функциональный символ от l аргументов и уже построены некоторые термы t_1, \dots, t_l , то можно построить новый терм $f(t_1, \dots, t_l)$.

Атомарные формулы Ag^+ имеют вид $(t=r)$, где t, r — произвольные термы языка Ag^+ .

Таким образом, язык Аг естественно рассматривать как часть языка Ag^+ : всякое выражение Аг является в то же время и выражением языка Ag^+ .

Нелогические аксиомы Ag^+ , так же как и нелогические аксиомы Аг, делятся на три группы: аксиомы равенства, аксиомы Пеано и определяющие аксиомы для п. р. описаний. Аксиомы равенства и аксиомы Пеано имеют тот же вид, что и в Аг, с той лишь разницей, что в схеме аксиом индукции теперь допускается произвольная формула языка Ag^+ . Определяющие же аксиомы для п. р. описаний просто копируют те тождества, которым должны по определению удовлетворять те п. р. функции, которые имеют эти описания.

Если m — натуральное число, то через \tilde{m} или через m^\sim обозначим терм Аг вида $SS \dots S0$, где функциональный символ S берется m раз. Это естественное изображение числа m в языке Аг. Примерами аксиом теории Ag^+ являются следующие формальные равенства:

$$(2, 1) (x) = x;$$

$$(3, 1, m) (x) = \tilde{m};$$

$$(6, 1, g, m) (0) = \tilde{m};$$

$$(6, 1, g, m) (Sx) = g(x, (6, 1, g, m) (x)),$$

где g — произвольное п. р. описание с двумя аргументами.

Таким образом, теория Ag составляет часть теории Ag^+ : все, что выводится в Ag , выводится и в Ag^+ .

Ag^+ является уже весьма выразительной теорией и позволяет записать и доказать практически все обычные факты, относящиеся к теории чисел. В частности, при наличии некоторого навыка в построении формальных выводов несложно установить выводимость в Ag^+ всех необходимых нам свойств геделевых номеров слов в простых алфавитах.

7. Замечательно, что, как обнаружил Гедель, теории Ag и Ag^+ оказываются по существу эквивалентными. Это объясняется тем, что, используя некоторые теоретико-числовые соображения, удастся равенства сложных термов языка Ag^+ изобразить формулами теории Ag , т. е. формулами, содержащими лишь сложение и умножение.

Точнее, для всякой формулы A языка Ag^+ определяется ее перевод — формула A^0 языка Ag , содержащая те же параметры, что и формула A . Далее, для каждого терма t языка Ag^+ и переменной y определяется формула $(y=t)^*$ языка Ag с теми же параметрами, что и формула $y=t$. Этот перевод обладает многими свойствами, которые и следует от него ожидать. Мы перечислим их в следующих шести леммах.

Л е м м а 1. Если A — формула Ag , t — терм Ag , то в Ag выводимы эквивалентности:

$$A^0 \equiv A, \\ (y=t)^* \equiv (y=t).$$

Л е м м а 2. Если A — формула Ag^+ и t — терм Ag^+ , то в Ag^+ выводимы эквивалентности:

$$A^0 \equiv A, \\ (y=t)^* \equiv (y=t).$$

Л е м м а 3. Если A — формула Ag^+ , t — терм Ag^+ , то имеют место следующие эквивалентности:

$$Ag^+ \vdash A \Leftrightarrow Ag \vdash A^0, \\ Ag^+ \vdash (y=t) \Leftrightarrow Ag \vdash (y=t)^*.$$

Л е м м а 4. Если A — формула Ag , то

$$Ag^+ \vdash A \Leftrightarrow Ag \vdash A.$$

Последняя лемма показывает, что относительно узкого языка Ag обе теории эквивалентны. Дополнительно можно отметить, что указанный перевод сохраняет логическую структуру формул. Это обстоятельство проявляется в том, что имеют место следующие две леммы.

Л е м м а 5. В теории Ag выводимы эквивалентности

$$(A \wedge B)^0 \equiv A^0 \wedge B^0, \\ (A \vee B)^0 \equiv A^0 \vee B^0,$$

$$(A \supset B)^0 \equiv A^0 \supset B^0,$$

$$(\neg A)^0 \equiv \neg A^0,$$

$$(\forall x A)^0 \equiv \forall x A^0,$$

$$(\exists x A)^0 \equiv \exists x A^0.$$

Здесь A, B — произвольные формулы языка Ag^+ .

Л е м м а 6. В теории Ag выводимы следующие эквивалентности

$$(t=r)^0 \equiv \exists y((y=t)^* \wedge (y=r)^*),$$

$$(A(x||t))^0 \equiv \exists y((A(x||y))^0 \wedge (y=t)^*),$$

$$(z=r(x||t))^* \equiv \exists y((z=r(x||y))^* \wedge (y=t)^*).$$

Здесь A — формула Ag^+ , t, r — термы Ag^+ и переменная y не входит свободно в A, r, t .

Мы не будем заниматься точной формулировкой перевода и доказательством лемм 1—6. Необходимую технику для этого заинтересованный читатель может найти в руководствах [3], [4].

Заметим в заключение, что теория Ag^+ , так же как и теория Ag , является явно заданной. Формулы, термы, выводы теории Ag^+ естественно трактуются как слова в основном алфавите и, следовательно, имеют геделевы номера.

§ 5. НЕКОТОРЫЕ ТЕОРЕМЫ ОБЩЕЙ ТЕОРИИ АЛГОРИФМОВ

1. Универсальная функция. Для каждого $n \geq 1$ рассмотрим $(n+2)$ -местный предикат *Клини*, определенный на натуральных числах: $T_n(e, x_1, \dots, x_n, z)$.

А именно $T_n(e, x_1, \dots, x_n, z)$ истинно тогда и только тогда, когда e есть геделев номер некоторой (приведенной) машины Тьюринга M , а z есть геделев номер конечного протокола вычислений Z для машины M , на вход которой подан набор чисел x_1, \dots, x_n (т. е. подано слово $x_1 q \dots q x_n$, где натуральные числа рассматриваются как слова в алфавите $\{ \}$), причем Z содержит заключительную конфигурацию.

Довольно очевидно, что T_n — разрешимый предикат, так как по данным e, x_1, \dots, x_n, z можно судить, верно T_n или нет. Важный факт (который мы оставим без проверки, детали читатель может найти, например, в [2]) состоит в том, что этот предикат *примитивно-рекурсивен*, т. е. существует примитивно-рекурсивная функция $t_n(e, x_1, \dots, x_n, z)$, принимающая в качестве значений лишь 0 и 1, и такая, что

$$T_n(e, x_1, \dots, x_n, z) \Leftrightarrow t_n(e, x_1, \dots, x_n, z) = 1.$$

Может быть построена также одноместная примитивно-рекурсивная функция $U(z)$ такая, что если z — геделев номер конечного протокола вычислений Z некоторой (приведенной) ма-

шины Тьюринга M и Z содержит заключительную конфигурацию K такую, что на ленте K записано натуральное число k , то $U(z)=k$. Если же z не имеет указанного вида, то $U(z)=0$.

Если $P(y, x_1, \dots, x_s)$ — общерекурсивный (в частности, примитивно-рекурсивный) предикат, то можно определить частично-рекурсивную функцию $f(x_1, \dots, x_s)$ следующей инструкцией: для данного набора x_1, \dots, x_s выясняем последовательно, истинно ли $P(0, x_1, \dots, x_s)$, $P(1, x_1, \dots, x_s)$, $P(2, x_1, \dots, x_s)$, Как только дойдем до наименьшего m такого, что $P(m, x_1, \dots, x_s)$ истинно, положим $f(x_1, \dots, x_s)=m$. Если же такого m не существует, то функция $f(x_1, \dots, x_s)$ по определению на наборе x_1, \dots, x_s считается неопределенной. Приведенную инструкцию можно оформить в виде программы некоторой машины Тьюринга и, таким образом, доказать частичную рекурсивность f .

Будем говорить, что функция f получена из предиката P операцией минимизации, а значение $f(x_1, \dots, x_s)$ обозначим через $\mu y P(y, x_1, \dots, x_s)$ (читается: «наименьшее y такое, что $P(y, x_1, \dots, x_s)$ »).

Уточним еще понятие подстановки для частично определенных функций. Пусть, например, g — одноместная, а f — s -местная частичные функции. Рассмотрим функцию h , определяемую следующим образом: $h(x_1, \dots, x_s)$ определено тогда и только тогда, когда: 1) определено $f(x_1, \dots, x_s)$ и 2) определено значение $g(f(x_1, \dots, x_s))$, и в этом случае

$$h(x_1, \dots, x_s) = g(f(x_1, \dots, x_s)).$$

В такой ситуации мы говорим, что функция h получена подстановкой функции f в функцию g . Такое определение подстановки для частичных функций обеспечивает, что функция h оказывается частично-рекурсивной, коль скоро таковы функции f и g .

Кратко определение h можно записать в виде

$$h(x_1, \dots, x_s) \simeq g(f(x_1, \dots, x_s)).$$

Знак \simeq (условное равенство) здесь и далее означает, что выражение слева определено тогда и только тогда, когда определено выражение справа, и в случае определенности эти выражения совпадают.

Определим теперь для каждого $n \geq 1$ $(n+1)$ -местную частично-рекурсивную функцию \mathcal{U}_n таким образом, что

$$\mathcal{U}_n(e, x_1, \dots, x_n) \simeq U(\mu z T_n(e, x_1, \dots, x_n, z)).$$

Основное свойство \mathcal{U}_n выражается следующей классической теоремой Клини о нормальной форме.

Теорема а. Для всякой частично-рекурсивной функции f найдется натуральное m такое, что для всех x_1, \dots, x_n :

$$f(x_1, \dots, x_n) \simeq \mathcal{U}_n(m, x_1, \dots, x_n).$$

▷ В качестве такого m можно взять геделев номер машины Тьюринга, вычисляющей функцию f . Требуемое утверждение вытекает из определения предиката T_n и функции f . \square

Пусть K — произвольный класс n -местных (вообще говоря, частичных) функций. Функция $g(x, x_1, \dots, x_n)$ называется *универсальной для класса K* , если:

1) для всякого натурального m соответствующая n -местная функция $g(m, x_1, \dots, x_n)$ принадлежит K ;

2) для всякой функции $f(x_1, \dots, x_n)$ из K найдется натуральное m такое, что n -местная функция $g(m, x_1, \dots, x_n)$ совпадает с f .

Непосредственным следствием предыдущей теоремы является следующая

Теорема об универсальной функции. *Частично-рекурсивная функция \mathcal{U}_n является универсальной для класса всех n -местных частично-рекурсивных функций.*

Укажем сразу же и некоторый отрицательный результат.

Теорема. *Не существует $(n+1)$ -местной частично-рекурсивной функции, универсальной для класса всех n -местных общерекурсивных функций.*

▷ Ограничимся случаем $n=1$. Допустим, что такая функция $G(x, x_1)$ существует. Так как при каждом m функция $G(m, x_1)$ общерекурсивна, то функция $G(x, x_1)$ определена при всех x, x_1 и, таким образом, также является общерекурсивной. Но тогда общерекурсивна и функция $g(x) = G(x, x) + 1$. Рассмотрим m такое, что $g(x) = G(m, x)$. Имеем

$$G(m, x) = g(x) = G(x, x) + 1.$$

Подставляя сюда вместо x число m , получим противоречие. \square

Идею этого доказательства можно использовать для построения общерекурсивной функции, не являющейся примитивно-рекурсивной.

Определим функцию $g(y, x)$ следующей инструкцией: если y не есть геделев номер п.р. описания, то $g(y, x) = 0$; если же y есть геделев номер п.р. описания функции F , то $g(y, x) = F(x)$.

Указанную инструкцию можно преобразовать в машину Тьюринга и, таким образом, доказать, что g — общерекурсивная функция. Рассмотрим теперь общерекурсивную функцию F такую, что $F(x) = g(x, x) + 1$.

Лемма. *Общерекурсивная функция F не является примитивно-рекурсивной.*

▷ Предположим противное, и пусть m — геделев номер п.р. описания функции F . Тогда

$$g(m, x) = F(x) = g(x, x) + 1.$$

Подставляя вместо x натуральное m , получим противоречие. \square

Для каждого натурального m соответствующую n -местную частично-рекурсивную функцию $U(\mu z T_n(m, x_1, \dots, x_n, z))$ мы обозначим через $\{m\}^n$. Число m назовем *геделевым номером частично-рекурсивной функции* $\{m\}^n$. Таким образом, всякая частично-рекурсивная функция имеет некоторый геделев номер. Индекс n в обозначении $\{m\}^n$ будем иногда опускать (и очень надеемся, что читатель не спутает это традиционное обозначение частично-рекурсивной функции с обозначением одноэлементного множества в теории множеств).

Из определения предиката T_n очевидно, что

$$! \{m\}^n(x_1, \dots, x_n) \Leftrightarrow \exists z T_n(m, x_1, \dots, x_n, z).$$

Напомним, что выражение слева означает, что функция $\{m\}^n$ определена на наборе x_1, \dots, x_n .

Лемма. Для всякого рекурсивно-перечислимого множества A натуральных чисел существует частично-рекурсивная функция g такая, что g принимает лишь одно значение 1, и для всякого x

$$! g(x) \Leftrightarrow x \in A.$$

▷ Если A рекурсивно-перечислимо, то существует машина Тьюринга M , вычисляющая предикат $x \in A$ согласно определению § 3, п. 3. Определим теперь функцию g инструкцией: $g(x) = 1$ тогда и только тогда, когда $M(x) = 1$; во всех остальных случаях g не определена. Эту инструкцию можно оформить в виде некоторой машины Тьюринга, вычисляющей g . □

Если m — геделев номер функции g , о которой идет речь в лемме, то, очевидно,

$$x \in A \Leftrightarrow \exists z T_1(m, x, z).$$

Натуральное m , для которого выполняется эта эквивалентность, мы назовем *геделевым номером рекурсивно-перечислимого множества* A . Рекурсивно-перечислимое множество с геделевым номером m обозначим через \mathscr{W}_m . Таким образом, по определению

$$x \in \mathscr{W}_m \Leftrightarrow \exists z T_1(m, x, z).$$

2. Невычислимые множества и функции.

Теорема. Множество натуральных чисел $\{x \mid \exists z T_1(x, x, z)\}$ рекурсивно-перечислимо, но не рекурсивно.

▷ Рекурсивная перечислимость рассматриваемого множества следует из леммы в § 3, п. 4 и рекурсивности предиката T_1 . Если бы наше множество было рекурсивным, то рекурсивно было бы и его дополнение $A = \{x \mid \neg \exists z T_1(x, x, z)\}$, а следовательно, A было бы и рекурсивно-перечислимым. Покажем, однако, что множество A отлично от всякого рекурсивно-перечислимого множества. В самом деле, рассмотрим произвольное

рекурсивно-перечислимое множество \mathcal{W}_m и предположим, что $A = \mathcal{W}_m$. Тогда для всякого x

$$\begin{aligned} \exists z T_1(m, x, z) &\Leftrightarrow x \in \mathcal{W}_m \Leftrightarrow x \in A \\ &\Leftrightarrow \neg \exists z T_1(x, x, z). \end{aligned}$$

Подставляя в эту цепочку эквивалентностей вместо x число m , приходим к противоречию. \square

С л е д с т в и е. *Множество натуральных чисел $\{x\} \neg \exists z T_1(x, x, z)$ не является рекурсивно-перечислимым.*

▷ Фактически это уже доказано нами выше, но следует также и непосредственно из формулировки предыдущей теоремы и теоремы Поста § 3, п. 6. \square

С л е д с т в и е. *Существует всюду определенная невычислимая функция.*

▷ Определим функцию g следующим образом: для каждого натурального x положим $g(x) = 1$, если $\exists z T_1(x, x, z)$, и $g(x) = 0$, если $\neg \exists z T_1(x, x, z)$. g — всюду определена и если бы была вычислимой, то оказалась бы общерекурсивной функцией. Но $g(x) = 1 \Leftrightarrow \neg \exists z T_1(x, x, z)$ и предикат справа оказался бы рекурсивным. \square

3. Проблема остановки. Мы говорим, что *проблема остановки* для машины Тьюринга M разрешима, если существует общерекурсивная функция h , принимающая лишь два значения 0 или 1, и такая, что $h(x) = 1$ тогда и только тогда, когда машина M , работая на входе x , дает конечный протокол вычислений.

Неформально говоря, проблема остановки разрешима, если существует «распознаватель остановки» h , узнающий по x , остановится ли машина M на x или нет.

Т е о р е м а. *Существует машина Тьюринга M с неразрешимой проблемой остановки.*

▷ Пусть $a(x)$ — общерекурсивная функция, тождественно равная нулю. Рассмотрим частично-рекурсивную функцию $g(x)$, равную $a(\mu z T_1(x, x, z))$. Тогда $!g(x) \Leftrightarrow \exists z T_1(x, x, z)$ и $!g(x) \Rightarrow g(x) = 0$. Пусть M' — приведенная машина Тьюринга, вычисляющая g . «Подправим» программу машины M' и получим машину M такую, что

- 1) M также вычисляет g ,
- 2) если машина M' на входе x дает конечный протокол вычислений без заключительной конфигурации или конечный протокол, в заключительной конфигурации которого на ленте записано непустое слово, то машина M на входе x не остановится, т. е. порождает бесконечный протокол вычислений.

Такую модификацию машины M' всегда можно произвести, дописав несколько команд снизу в программе M' , изменив ее заключительное состояние и расширив алфавит.

Тогда $\exists z T_1(x, x, z) \Leftrightarrow M$ остановится на x . Теперь невозможность соответствующей функции h для M следует из теоремы п. 2. \square

Подчеркнем, что для M неразрешима именно *массовая проблема* остановки: не существует *единого* алгоритма h , который бы узнавал, остановится ли M на x для *всякого* натурального числа x . Если же исследователю предложить конкретное натуральное число m и спросить, верно ли, что M остановится на входе m , то нет оснований полагать, что наш исследователь не справится с этим вопросом по некотором размышлении. Невозможен лишь единый механический способ решения этого вопроса для всех m сразу.

4. Рекурсивно-неотделимые множества. Напомним, что через ω мы обозначаем множество всех натуральных чисел. Два множества $A_0, A_1 \subseteq \omega$ назовем *рекурсивно-отделимыми*, если существуют два рекурсивно-перечислимых множества B_0, B_1 такие, что

$$A_0 \subseteq B_0, A_1 \subseteq B_1, B_0 \cap B_1 = \emptyset, B_0 \cup B_1 = \omega.$$

Про множества B_0 и B_1 мы будем говорить, что они *рекурсивно отделяют* множества A_0 и A_1 . По теореме Поста (§ 3, п. 6) множества B_0 и B_1 в этом случае необходимо являются и рекурсивными.

Упражнение. Если множества A_0, A_1 рекурсивно-неотделимы и не пересекаются (т. е. $A_0 \cap A_1 = \emptyset$), то оба они не рекурсивны. Более того, всякое множество C такое, что $A \subseteq C, C \cap B = \emptyset$ также не рекурсивно.

Определим два предиката:

$$W_0(x, y) \equiv T_1(j_2x, x, y) \wedge (\forall z \leq y) \neg T_1(j_1x, x, z);$$

$$W_1(x, y) \equiv T_1(j_1x, x, y) \wedge (\forall z \leq y) \neg T_1(j_2x, x, z).$$

Можно проверить, что оба предиката примитивно-рекурсивны.

Определим теперь два рекурсивно-перечислимых множества:

$$V_0 \equiv \{x \mid \exists y W_0(x, y)\};$$

$$V_1 \equiv \{x \mid \exists y W_1(x, y)\}.$$

Теорема. *Рекурсивно-перечислимые множества V_0, V_1 не пересекаются и являются рекурсивно-неотделимыми.*

▷ Допустим, что $x \in V_0$ и $x \in V_1$. Тогда найдутся y_0 и y_1 такие, что $W_0(x, y_0)$ и $W_1(x, y_1)$, т. е.

- 1) $T_1(j_2x, x, y_0)$;
- 2) $(\forall z \leq y_0) \neg T_1(j_1x, x, z)$;
- 3) $T_1(j_1x, x, y_1)$;
- 4) $(\forall z \leq y_1) \neg T_1(j_2x, x, z)$.

Из 1) и 4) следует, что $y_1 < y_0$, в то время как из 2) и 3) следует $y_0 < y_1$, и мы приходим к противоречию.

Итак, множества V_0 и V_1 не пересекаются.

Рассмотрим теперь произвольные перечислимые множества B_0, B_1 такие, что $V_0 \subseteq B_0, V_1 \subseteq B_1, B_0 \cap B_1 = \emptyset$, и покажем, что в этом случае необходимо $B_0 \cup B_1 \neq \omega$.

Пусть B_0 имеет геделев номер m_0 и B_1 — номер m_1 . Положим $m = j(m_0, m_1)$. Мы утверждаем, что $m \notin B_0$ и $m \notin B_1$. Докажем, например, $m \notin B_0$ ($m \notin B_1$ доказывается аналогично).

С этой целью предположим противное и пусть $m \in B_0$. Тогда $m \in \mathcal{W}_{m_0}$, т. е. $\exists y T_1(m_0, m, y)$, и ввиду $m_0 = j_1 m$

$$\exists y T_1(j_1 m, m, y). \quad (1)$$

Далее, ввиду $m \in B_0$ и $B_0 \cap B_1 = \emptyset$ имеем

$$m \notin B_1, \quad (2)$$

т. е. $m \notin \mathcal{W}_{m_1}$, что означает $\neg \exists y T_1(m_1, m, y)$, что, в свою очередь, ввиду $m_1 = j_2 m$ означает

$$\forall y \neg T(j_2 m, m, y). \quad (3)$$

Из (1) и (3) следует

$$\exists y (T_1(j_1 m, m, y) \wedge (\forall z \leq y) \neg T_1(j_2 m, m, z)),$$

т. е. $\exists y W_1(m, y)$ и, следовательно, $m \in B_1$. Однако это противоречит утверждению (2). \square

Теорема о неполной функции. Существует частично-рекурсивная функция f , принимающая в качестве значений лишь 0 и 1, и такая, что не существует общерекурсивной функции h , дополняющей f , т. е. такой функции h , что для всякого x ; для которого определено $f(x)$, имеет место $f(x) = h(x)$.

\triangleright Согласно лемме § 5 п. 1 найдутся частично-рекурсивные функции g_0 и g_1 такие, что $x \in V_i \Leftrightarrow !g_i(x)$. Определим теперь частично-рекурсивную функцию f следующей инструкцией: для данного x вычисляем одновременно $g_0(x)$ и $g_1(x)$. Если определено $g_0(x)$, то положим $f(x) = 0$, если же определено $g_1(x)$, то положим $f(x) = 1$. Если же не определены обе функции, то считается неопределенной и $f(x)$. Заметим, что не могут быть одновременно определены оба значения $g_0(x)$ и $g_1(x)$, так как $V_0 \cap V_1 = \emptyset$.

Мы утверждаем, что функция f — искомая.

Предположим, что существует общерекурсивное пополнение функции f . Пусть это будет общерекурсивная функция h . Пусть a — примитивно-рекурсивная функция такая, что $a(0) = 0$, $a(x+1) = 1$. Определим $h'(x) = a(h(x))$. Тогда h' является общерекурсивной функцией, принимающей лишь значения 0 и 1 и также дополняющей f . Определим, далее, рекурсивные множества:

$$B_0 = \{x | h'(x) = 0\}, \quad B_1 = \{x | h'(x) = 1\}.$$

Из определений легко видеть, что B_0, B_1 рекурсивно отделяют множества V_0, V_1 , что невозможно по предыдущей теореме. \square

5. Теорема о рекурсии. Фиксируем набор натуральных чисел

$$a = \langle e, k_1, \dots, k_m \rangle$$

и определим для этого набора натуральное число t_a следующим образом. Построим приведенную машину M_a . А именно

для всякого набора y_1, \dots, y_n натуральных чисел значение $M_a(y_1 q \dots y_n q)$ определено тогда и только тогда, когда определена универсальная функция $\mathcal{U}_{m+n}(e, k_1, \dots, k_m, y_1, \dots, y_n)$, и тогда $M_a(y_1 q \dots q y_n) = \mathcal{U}_{m+n}(e, k_1, \dots, k_m, y_1, \dots, y_n)$. Теперь в качестве t_a возьмем геделев номер машины M_a .

Заметим теперь, что наша инструкция дает способ вычисления по набору a соответствующего t_a . Эту инструкцию можно оформить в виде машины Тьюринга. Таким образом, функция

$$S_n^m(e, x_1, \dots, x_m) = t_{\langle e, x_1, \dots, x_m \rangle}$$

является общерекурсивной. Более тщательный анализ показывает, что S_n^m является даже примитивно-рекурсивной функцией. Таким образом, имеет место

Лемма. Может быть построена примитивно-рекурсивная функция S_n^m от $(m+1)$ аргументов такая, что для всякого набора натуральных чисел $e, x_1, \dots, x_m, y_1, \dots, y_n$ имеем

$$\{e\}(x_1, \dots, x_m, y_1, \dots, y_n) \simeq \{S_n^m(e, x_1, \dots, x_m)\}(y_1, \dots, y_n).$$

Теорема Клини о рекурсии. Для всякой частично-рекурсивной функции $\psi(z, x_1, \dots, x_n)$ можно построить натуральное число e такое, что

$$\{e\}(x_1, \dots, x_n) \simeq \psi(e, x_1, \dots, x_n).$$

▷ Рассмотрим частично-рекурсивную функцию

$$\varphi(y, x_1, \dots, x_n) \simeq \psi(S_n^1(y, y), x_1, \dots, x_n).$$

Пусть f — се геделев номер и $e = S_n^1(f, f)$. Мы утверждаем, что это e — искомое. В самом деле,

$$\begin{aligned} \{e\}(x_1, \dots, x_n) &\simeq \{S_n^1(f, f)\}(x_1, \dots, x_n) \simeq \\ &\{f\}(f, x_1, \dots, x_n) \simeq \varphi(f, x_1, \dots, x_n) \simeq \\ &\psi(S_n^1(f, f), x_1, \dots, x_n) \simeq \psi(e, x_1, \dots, x_n). \quad \square \end{aligned}$$

§ 1. НЕПОЛНОТА И НЕРАЗРЕШИМОСТЬ АКСИМАТИЧЕСКИХ ТЕОРИЙ

Мы установим неполноту и неразрешимость некоторых аксиоматических теорий. Большую часть доказательств мы проводим для конкретной аксиоматической теории Ag^+ , но развитые методы, как мы увидим, будут применимы и ко многим другим теориям.

1. Теорема о неподвижной точке. Напомним, что если A — слово в простом алфавите, то через γA мы обозначаем его геделев номер (гл. II, § 4, п. 3). Далее, если m — натуральное число, то через m^\sim или через \tilde{m} мы обозначим терм языка Ag^+ , естественно изображающий натуральное число в языке Ag^+ , т. е. терм вида $SS \dots S0$, где функциональный символ S берется m раз (гл. II, § 4, п. 6). Таким образом, для каждого слова в простом алфавите можно определить терм — естественное изображение этого слова в Ag^+ :

$$[A] \equiv (\gamma A)^\sim.$$

Пусть $t(x_1, \dots, x_n)$, $r(x_1, \dots, x_n)$ — термы Ag^+ , где x_1, \dots, x_n — список различных переменных, среди которых содержатся все параметры термов t и r . Функциональные символы t и r задают примитивно-рекурсивные описания некоторых примитивно-рекурсивных функций и, таким образом, сами термы естественно определяют некоторые примитивно-рекурсивные функции $F(x_1, \dots, x_n)$ и $G(x_1, \dots, x_n)$ соответственно.

Л е м м а. Для всяких натуральных k_1, \dots, k_n

1) если $F(k_1, \dots, k_n) = G(k_1, \dots, k_n)$, то

$$Ag^+ \vdash t(\tilde{k}_1, \dots, \tilde{k}_n) = r(\tilde{k}_1, \dots, \tilde{k}_n);$$

2) если $F(k_1, \dots, k_n) \neq G(k_1, \dots, k_n)$, то

$$Ag^+ \vdash \neg t(\tilde{k}_1, \dots, \tilde{k}_n) = r(\tilde{k}_1, \dots, \tilde{k}_n).$$

Мы не будем доказывать эту техническую лемму (см., например, [3], [4]). Суть ее состоит в том, что система Ag^+ обладает достаточными выразительными возможностями: произвольное примитивно-рекурсивное вычисление с конкретными натуральными числами выразимо в Ag^+ .

Заметим, что мы отнюдь не утверждаем, например, что если $F(k_1, \dots, k_n) = G(k_1, \dots, k_n)$ для всех натуральных чисел k_1, \dots, k_n ,

то $\text{Ag}^+ \vdash t(x_1, \dots, x_n) = r(x_1, \dots, x_n)$ со свободными переменными x_1, \dots, x_n . Как мы увидим позже, это и неверно в общем случае. Речь идет лишь о выводимости формальных равенств с конкретными натуральными числами k_1, \dots, k_n .

Вышеприведенную лемму называют леммой о нумерической выразимости примитивно-рекурсивных функций в Ag^+ . Слово «нумерической» как раз подчеркивает, что речь идет о выводимости конкретных числовых равенств. Если t, r — замкнутые термы, то в стандартной модели ω они имеют определенное числовое значение. Из предыдущей леммы следует, что:

если $\omega \models t = r$, то $\text{Ag}^+ \vdash t = r$;

если $\omega \models \neg t = r$, то $\text{Ag}^+ \vdash \neg t = r$.

Теорема о неподвижной точке. Пусть A — формула Ag^+ и x — переменная. Тогда может быть построена формула B такая, что

$$\text{Ag}^+ \vdash B \equiv A([B]).$$

Здесь $A([B]) \equiv A(x \parallel [B])$.

[> Фиксируем переменную x . Рассмотрим примитивно-рекурсивную функцию $SUB(y, z)$ такую, что если y есть геделев номер некоторой формулы C , то $SUB(y, z)$ есть геделев номер формулы $C(x \parallel z)$. Функция SUB имеет определенное примитивно-рекурсивное описание, и, следовательно, можно определить соответствующий ей терм $Sub(y, z)$ теории Ag^+ .

Теперь по данной формуле A построим формулу

$$A' \equiv A(Sub(x, x)) \quad (1)$$

и определим

$$B \equiv A'([A']). \quad (2)$$

Отсюда следует, что

$$\gamma B = SUB(\gamma A', \gamma A'),$$

и по предыдущей лемме тогда

$$\text{Ag}^+ \vdash [B] = Sub([A'], [A']). \quad (3)$$

Далее, мы видим, что в Ag^+ выводима следующая цепочка эквивалентностей:

$$B \equiv A'([A']) \equiv A(Sub([A'], [A'])) \equiv A([B]).$$

Здесь первая эквивалентность вытекает из определения (2) формулы B . Вторая эквивалентность имеет место ввиду определения (1) формулы A' . Третья эквивалентность следует из выводимости (3). \square

2. Теорема Геделя о неполноте. С помощью геделевой нумерации выводимость в теории Ag^+ может изучаться в рамках самой формальной теории Ag^+ (см. по этому поводу гл. II, § 3, п. 7 и § 4, п. 6). В частности, может быть построен примитивно-рекурсивный предикат $PRF(y, x)$, истинный тогда и только

тогда, когда y есть геделев номер вывода в Ag^+ для формулы с геделевым номером x . Как всякий примитивно-рекурсивный предикат, он может быть выражен в форме $F(y, x) = 1$ для некоторой примитивно-рекурсивной функции F , принимающей лишь значения 0 и 1. Если f — п. р. описание функции F , то предикату PRF в Ag^+ естественно соответствует формула $f(y, x) = S0$, которую мы обозначим через $Prf(y, x)$.

Рассмотрим теперь формулу

$$Pr(x) \equiv \exists y Prf(y, x),$$

которая утверждает, что x есть геделев номер формулы Ag^+ , выводимой в Ag^+ .

Пусть A_0 — конкретная формула, отрицание которой выводимо в Ag^+ . Например, можно взять в качестве A_0 формулу $0 = S0$. Рассмотрим замкнутую формулу

$$Con \equiv \forall y \neg Prf(y, [0 = S0]).$$

Эта формула утверждает, что теория Ag^+ непротиворечива.

С помощью теоремы о неподвижной точке определим теперь замкнутую формулу Геделя ν таким образом, что

$$Ag^+ \vdash \nu \equiv \neg Pr([\nu]). \quad (1)$$

Исходя из этой эквивалентности, можно сказать, что формула ν «утверждает, что она сама не выводима».

Лемма 1. Если теория Ag^+ непротиворечива, то формула ν не выводима в Ag^+ .

▷ Предположим противное, и пусть p — геделев номер вывода формулы ν в теории Ag^+ . Тогда имеет место $PRF(p, \nu v)$, и по лемме п. 1 отсюда $Ag^+ \vdash Prf(\bar{p}, [\nu])$ и, далее, $Ag^+ \vdash Pr([\nu])$, что ввиду (1) влечет $Ag^+ \vdash \neg \nu$. С другой стороны, по допущению $Ag^+ \vdash \nu$, и мы заключаем, что теория Ag^+ противоречива вопреки предположению. □

З а м е ч а н и е. Напомним, что теория T называется непротиворечивой, если не существует формулы A теории T такой, что одновременно $T \vdash A$ и $T \vdash \neg A$. Теория T непротиворечива тогда и только тогда, когда существует формула, не выводимая в T . Читателю может показаться, что допущение о непротиворечивости излишне в формулировке предыдущей леммы. Кажется очевидным, что это допущение и так имеет место. Теория Ag^+ , конечно, непротиворечива. В самом деле, все выводимые в Ag^+ формулы истинны в стандартной модели ω в то время, как, например, формула $0 = S0$ ложна в стандартной модели и, следовательно, не выводима в Ag^+ .

Дело в том, что мы хотели бы, чтобы и формулировка, и доказательство леммы 1 естественно формализовались в теории Ag^+ , а, как мы увидим ниже, формула Con как раз не выводима в Ag^+ . Только что намеченное нами выше доказательство непротиворечивости Ag^+ использует некоторые теоретико-мно-

жественные соображения из теории моделей и не может быть непосредственно переведено в язык Ag^+ .

Пусть T — теория в языке Ag^+ . Будем говорить, что теория T ω -непротиворечива, если не существует формулы $A(y)$ с одной свободной переменной такой, что

1) $T \vdash \exists y A(y)$,

2) для всякого натурального m имеет место $T \vdash \neg A(\bar{m})$.

Упражнение. Докажите, что если T ω -непротиворечива, то теория T и просто непротиворечива.

Л е м м а 2. Если Ag^+ ω -непротиворечива, то в Ag^+ не выводима формула $\neg v$.

[> Предположим противное, и пусть $Ag^+ \vdash \neg v$. Согласно (1) тогда $Ag^+ \vdash Pr([v])$, т. е. $Ag^+ \vdash \exists y Prf(y, [v])$. С другой стороны, возьмем произвольное натуральное m . Имеем $\neg PRF(m, \gamma v)$, так как по лемме 1 формула v не выводима. По лемме в п. 1 отсюда $Ag^+ \vdash \neg Prf(\bar{m}, [v])$, и мы получаем ω -противоречие в Ag^+ . \square

Теорема Геделя о неполноте. Если теория Ag^+ ω -непротиворечива, то формула v и формула $\neg v$ не выводимы в этой теории.

[> Это следствие лемм 1 и 2. \square

З а м е ч а н и е. Напомним, что теория называется полной, если для всякой замкнутой формулы A имеем $T \vdash A$ или $T \vdash \neg A$. Теория Ag^+ оказывается, таким образом, неполной.

Формула v истинна в стандартной модели ω , так как в силу леммы 1 не существует вывода v в Ag^+ и, следовательно, истинна формула $\neg Pr([v])$. Таким образом, v — пример истинного предложения, не выводимого в Ag^+ .

Далее, неверно, что $Ag^+ \vdash \neg Prf(y, [v])$ со свободной переменной y , так как иначе было бы нетрудно получить и $Ag^+ \vdash v$. В то же время для всякого конкретного натурального числа m имеем $Ag^+ \vdash \neg Prf(\bar{m}, [v])$. Теории Ag^+ как бы не хватает «обобщающей абстракции», в силу чего и возникает ее неполнота.

Рассмотрим теорию $T = Ag^+ + \neg v$, получаемую добавлением к Ag^+ в качестве новой нелогической аксиомы (ложного!) предложения $\neg v$. Теория T непротиворечива, так как если бы в T выводилось противоречие, то по законам логики $Ag^+ \vdash v$. Тем не менее теория T ω -противоречива. Действительно, $\neg v$ эквивалентно $\exists y Prf(y, [v])$, так что $T \vdash \exists y Prf(y, [v])$. В то же время, для всякого натурального m по лемме 1 $\neg PRF(m, \gamma v)$, и, значит, в Ag^+ и тем более в T выводимо $\neg Prf(\bar{m}, [v])$.

3. Вторая теорема Геделя. Как формулировка, так и доказательство леммы 1 предыдущего пункта касаются лишь конструктивных объектов и после перехода к геделевым номерам допускают естественную формализацию в рамках теории Ag^+ . Проводя такую формализацию, мы получим, что имеет место утверждение

$$Ag^+ \vdash Con \supset \neg Pr([v]).$$

Далее, имеет место следующее утверждение:

$$\neg \exists y PRF(y, \gamma v) \Rightarrow (Ag^+ \text{ непротиворечива}).$$

▷ В самом деле, пусть Ag^+ противоречива. Тогда в Ag^+ выводима всякая формула и, в частности, v . Если D — вывод v в Ag^+ , то имеет место $PRF(\gamma D, \gamma v)$ и, следовательно, $\exists y PRF(y, \gamma v)$ вопреки предположению. □

Только что проведенное доказательство допускает несложную формализацию внутри теории Ag^+ , в результате чего получаем утверждение

$$Ag^+ \vdash \neg \exists y Prf(y, [v]) \supset Con,$$

т. е.

$$Ag^+ \vdash \neg Pr([v]) \supset Con.$$

Сопоставляя две полученные выводимости, заключаем

$$Ag^+ \vdash \neg Pr([v]) \equiv Con.$$

С другой стороны, по построению v имеем

$$Ag^+ \vdash v \equiv \neg Pr([v]).$$

Таким образом, окончательно

$$Ag^+ \vdash Con \equiv v.$$

Теорема (вторая теорема Геделя). Если теория Ag^+ непротиворечива, то формула Con не выводима в Ag^+ .

▷ Это следствие предыдущей выводимости и леммы 1 предыдущего пункта. □

Вторая теорема Геделя утверждает факт принципиальной важности: непротиворечивость рассматриваемой теории не может быть доказана средствами самой теории. Для доказательства непротиворечивости теории необходимо привлекать понятия, выходящие за ее рамки. Так, выше нам удалось установить непротиворечивость теории Ag^+ , привлекая теоретико-модельные соображения. Подобным образом можно показать, что непротиворечивость ZF не может быть установлена в рамках самой теории ZF. Очень интересной задачей является отыскание достаточно убедительных средств, выходящих за пределы ZF.

4. Теорема о неполноте в форме Россера. Как показал Россер, с помощью подходящего подбора формулы в теореме о неполноте можно заменить требование ω -непротиворечивости на более слабое требование простой непротиворечивости.

Рассмотрим примитивно-рекурсивную функцию $NEG(x)$ такую, что если x — геделев номер формулы A , то $NEG(x)$ есть геделев номер формулы $\neg A$. По примитивно-рекурсивному описанию этой функции можно построить соответствующий терм $Neg(x)$ теории Ag^+ .

С помощью теоремы о неподвижной точке построим формулу ρ так, что

$$Ag^+ \vdash \rho \equiv \forall y (Prf(y, [\rho]) \supset (\exists z \leq y) Prf(z, Neg([\rho]))).$$

Теорема Россера о неполноте. Если теория Ag^+ непротиворечива, то в ней не выводимы ни формула ρ , ни формула $\neg \rho$.

Предположим, что $Ag^+ \vdash \rho$. Тогда найдется m такое, что $PRF(m, \gamma\rho)$. Если бы для некоторого i было бы $PRF(i, \gamma(\neg \rho))$, то теория Ag^+ оказалась бы противоречивой вопреки предположению.

Таким образом, имеем

$$PRF(m, \gamma\rho) \wedge (\forall z \leq m) \neg PRF(z, NEG(\gamma\rho)).$$

Это примитивно-рекурсивное соотношение может быть выведено в Ag^+ , и мы получаем

$$Ag^+ \vdash Prf(\tilde{m}, [\rho]) \wedge (\forall z \leq \tilde{m}) \neg Prf(z, Neg([\rho])).$$

Отсюда

$$Ag^+ \vdash \exists y (Prf(y, [\rho]) \wedge (\forall z \leq y) \neg Prf(z, Neg([\rho]))).$$

Но эта последняя формула логически эквивалентна $\neg \rho$. Таким образом, $Ag^+ \vdash \neg \rho$, что противоречит допущению $Ag^+ \vdash \rho$.

Допустим теперь, что $Ag^+ \vdash \neg \rho$. Тогда найдется натуральное m такое, что $PRF(m, NEG(\gamma\rho))$, и по лемме п. 1 тогда $Ag^+ \vdash Prf(\tilde{m}, Neg([\rho]))$.

Отсюда по законам формальной арифметики получим

$$Ag^+ \vdash (\forall y \geq \tilde{m}) (\exists z \leq y) Prf(z, Neg([\rho])),$$

и, далее, ослабляя это утверждение,

$$Ag^+ \vdash (\forall y \geq \tilde{m}) (Prf(y, [\rho]) \supset (\exists z \leq y) Prf(z, Neg([\rho]))). \quad (1)$$

Кроме того, имеет место содержательное утверждение

$$(\forall y \leq m) (PRF(y, \gamma\rho) \Rightarrow (\exists z \leq y) PRF(z, NEG(\gamma\rho))) \quad (2)$$

просто в силу того, что для всякого y предикат $PRF(y, \gamma\rho)$ ложен (иначе было бы $Ag^+ \vdash \rho$ и Ag^+ оказалась бы противоречивой).

Примитивно-рекурсивное соотношение (2) может быть выведено в Ag^+ , и таким образом:

$$Ag^+ \vdash (\forall y \leq \tilde{m}) (Prf(y, [\rho]) \supset (\exists z \leq y) Prf(z, Neg([\rho]))). \quad (3)$$

Сопоставляя (1) и (3), по законам формальной арифметики получим

$$Ag^+ \vdash \forall y (Prf(y, [\rho]) \supset (\exists z \leq y) Prf(z, Neg([\rho]))),$$

что есть не что иное, как $Ag^+ \vdash \rho$. Таким образом, $Ag^+ \vdash \rho$ и $Ag^+ \vdash \neg \rho$, и мы опять приходим к противоречию. \square

5. Теорема Леба. Так называется следующая

Теорема. Если $Ag^+ \vdash Pr([A]) \supset A$, то $Ag^+ \vdash A$.

\triangleright По теореме о неподвижной точке найдем формулу ψ такую, что

$$Ag^+ \vdash \psi \equiv (Pr([\psi]) \supset A). \quad (1)$$

Л е м м а. $Ag^+ \vdash \psi \Rightarrow Ag^+ \vdash A$.

▷ Пусть $Ag^+ \vdash \psi$ и m — соответствующий вывод. Тогда $PRF(m, \gamma\psi)$ и, значит, $Ag^+ \vdash Pr(\tilde{m}, [\psi])$, отсюда $Ag^+ \vdash Pr([\psi])$. Ввиду (1) тогда $Ag^+ \vdash \psi \equiv A$. По допущению отсюда $Ag^+ \vdash A$. □

Формализуя доказательство этой леммы, получим

$$Ag^+ \vdash Pr([\psi]) \supset Pr([A]). \quad (2)$$

Далее, по допущению теоремы имеем

$$Ag^+ \vdash Pr([A]) \supset A. \quad (3)$$

Из (3) и (2)

$$Ag^+ \vdash Pr([\psi]) \supset A,$$

что ввиду (1) дает $Ag^+ \vdash \psi$. Но тогда по лемме имеем $Ag^+ \vdash A$. □

6. **Неразрешимость.** Обозначим через $[Ag^+]^0$ множество всех геделевых номеров предложений, выводимых в Ag^+ и через $[Ag^+]^1$ — множество всех геделевых номеров предложений, опровержимых в Ag^+ , т. е.

$$[Ag^+]^1 \equiv \{\gamma B \mid Ag^+ \vdash \neg B, B \text{ — предложение}\}.$$

Т е о р е м а. Множества $[Ag^+]^0$ и $[Ag^+]^1$ всех номеров выводимых и соответственно опровержимых предложений Ag^+ рекурсивно-неотделимы.

▷ Рассмотрим примитивно-рекурсивные предикаты $W_0(x, y)$ и $W_1(x, y)$ (гл. II, § 5, п. 4). Они задаются арифметическими формулами, которые мы также будем обозначать через W_0 и W_1 . Как было показано во второй главе, множества $V_0 = \{x \mid \exists y W_0(x, y)\}$ и $V_1 = \{x \mid \exists y W_1(x, y)\}$ не пересекаются, т. е.

$$\exists y W_1(x, y) \Rightarrow \neg \exists y W_0(x, y).$$

Доказательство этого утверждения можно формализовать в Ag^+ и получить

$$Ag^+ \vdash \exists y W_1(x, y) \supset \neg \exists y W_0(x, y). \quad (1)$$

Определим теперь формулу $B(x) \equiv \exists y W_0(x, y)$.

Если $m \in V_0$, то существует k такое, что $W_0(m, k)$, значит, $Ag^+ \vdash W_0(\tilde{m}, \tilde{k})$, следовательно, $Ag^+ \vdash \exists y W_0(\tilde{m}, y)$, т. е. $Ag^+ \vdash B(\tilde{m})$ и, значит, $\gamma B(\tilde{m}) \in [Ag^+]^0$. Итак,

$$m \in V_0 \Rightarrow \gamma B(\tilde{m}) \in [Ag^+]^0. \quad (2)$$

Пусть теперь $m \in V_1$, тогда существует k такое, что $W_1(m, k)$. Отсюда $Ag^+ \vdash W_1(\tilde{m}, \tilde{k})$ и, далее, $Ag^+ \vdash \exists y W_1(\tilde{m}, y)$. Ввиду (1) тогда $Ag^+ \vdash \neg \exists y W_0(\tilde{m}, y)$, т. е. $Ag^+ \vdash \neg B(\tilde{m})$, и, таким образом, $B(\tilde{m}) \in [Ag^+]^1$. Итак,

$$m \in V_1 \Rightarrow \gamma B(\tilde{m}) \in [Ag^+]^1. \quad (3)$$

Предположим противное, и пусть B_0, B_1 рекурсивно отделяют множества $[Ag^+]^0, [Ag^+]^1$, т. е. B_0, B_1 — рекурсивно-перечислимы,

$$[Ag^+]^0 \subseteq B_0, [Ag^+]^1 \subseteq B_1, \quad (4)$$

$$B_0 \cap B_1 = \emptyset, B_0 \cup B_1 = \omega. \quad (5)$$

Определим тогда рекурсивно-перечислимые множества $B_i' = \{m \mid \gamma B(\tilde{m}) \in B_i\}$. Ввиду (2) — (4) имеем $V_i \subseteq B_i'$, а в силу (5) $B_0' \cap B_1' = \emptyset, B_0' \cup B_1' = \omega$. Таким образом, оказывается, что множества V_0, V_1 также рекурсивно-отделимы, что невозможно. \square

Теорема о неразрешимости. Пусть T — непротиворечивая теория в языке Ag^+ , в которой выводятся все нелогические аксиомы Ag^+ . Тогда T неразрешима.

\triangleright Все, что выводится в Ag^+ , выводится и в T , так что $[T]^0 \subseteq [Ag^+]^0$. Ввиду непротиворечивости T очевидно $[T]^0 \cap [Ag^+]^1 = \emptyset$. Если T была бы разрешимой, то множество $[T]^0$ было бы рекурсивным. Тогда рекурсивным оказалось бы и дополнение $\omega \setminus [T]^0$ этого множества. Но тогда множества $[T]^0$ и $\omega \setminus [T]^0$ рекурсивно отделяют множества $[Ag^+]^0$ и $[Ag^+]^1$. \square

Следствие. Если Ag^+ непротиворечива, то Ag^+ — неразрешимая теория.

\triangleright Достаточно в теореме взять Ag^+ в качестве T . \square

7. Распространение результатов на другие теории. Мы подробно исследовали теорию Ag^+ , но все теоремы настоящего параграфа имеют место и по отношению ко многим другим теориям. Достаточно, чтобы рассматриваемая теория T обладала следующими двумя свойствами:

1) она должна быть явно заданной. Это нужно для того, чтобы на языке арифметики можно было бы говорить о выводимости в теории, например, чтобы был общерекурсивен предикат $PRF_T(y, x)$ и можно было построить соответствующую арифметическую формулу $Prf_T(y, x)$, утверждающую, что y есть геделев номер вывода в теории T формулы с геделевым номером x ;

2) теория T должна содержать арифметику Ag^+ . Это нужно для того, чтобы в рамках теории T можно было бы вывести необходимые свойства арифметических предикатов и, значит, доказать внутри T необходимые свойства предиката выводимости самой теории T . Понятие «содержать арифметику» при этом может трактоваться очень широко. Достаточно, чтобы теория Ag^+ в том или ином смысле интерпретировалась бы в теории T , например, достаточно, чтобы существовал некоторый перевод A^0 формул языка Ag^+ в формулы теории T , перевод, обладающий свойствами, аналогичными указанным в гл. II, § 4, п. 7.

Теории $Ag, Ag^+, Ag2, ZF$ удовлетворяют этим требованиям. Следовательно, все эти теории, в случае их непротиворечивости, неполны, неразрешимы и множества их выводимых и опровержимых предложений рекурсивно-неотделимы. Для каждой из указанных теорий T можно рассмотреть арифметическую

формулу $ConT$, утверждающую, что теория T непротиворечива:

$$ConT \equiv \neg \exists y Pr_T(y, [A_0]),$$

где A_0 — фиксированное предложение T , отрицание которого выводимо в T .

Если теория T непротиворечива, то формула $ConT$ (точнее, ее перевод в теорию T) не выводится в T . В этом и состоит вторая теорема Геделя, сформулированная применительно к теории T .

Мы не будем заниматься далее уточнением класса теорий, для которых имеют место наши теоремы, а упомянем еще об одном классическом результате.

Внимательный анализ показывает, что результат о неразрешимости Ag^+ в п. 6 получается, если вместо всей теории Ag^+ взять лишь фрагмент Ag^- теории Ag , содержащий лишь конечное число нелогических аксиом из Ag . Обозначим конъюнкцию всех аксиом Ag^- через A . Это — предложение языка Ag . Из определения выводимости следует, что

$$Ag^- \vdash B \Leftrightarrow \vdash A \supset B,$$

где справа стоит выводимость в логике предикатов.

Теорема о неразрешимости исчисления предикатов. *Множество формул, выводимых в исчислении предикатов (в языке Ag), не рекурсивно.*

▷ Предположим, что это множество рекурсивно. Тогда существует общерекурсивная функция f такая, что $f(\gamma B) = 1 \Leftrightarrow \vdash B$ для всякой формулы B . Определим тогда общерекурсивную функцию h следующей инструкцией: $h(x)$ равно 0 или 1 и $h(x) = 1 \Leftrightarrow x$ есть геделев номер некоторой формулы B языка Ag и $f(\gamma(A \supset B)) = 1$. Тогда $h(\gamma B) = 1 \Leftrightarrow Ag^- \vdash B$, и теория Ag^- оказывается разрешимой. \square

§ 2. ТЕОРЕМА ГЕДЕЛЯ О ПОЛНОТЕ ИСЧИСЛЕНИЯ ПРЕДИКАТОВ

1. Нетрудно проверить, что всякая формула, выводимая в исчислении предикатов, является общезначимой, т. е. логическим законом. Теорема Геделя о полноте утверждает, что верно и обратное, т. е. всякий логический закон необходимо выводится в исчислении предикатов. Таким образом, исчисление предикатов является адекватным инструментом для получения логических законов.

Напомним, что теория T в языке Ω называется *непротиворечивой*, если не существует формулы C языка Ω такой, что одновременно $T \vdash C$, $T \vdash \neg C$.

Основным результатом этого параграфа является следующая ниже фундаментальная теорема о существовании модели, также, по существу, доказанная Геделем.

Теорема. *Всякая непротиворечивая теория T в языке Ω первого порядка имеет модель.*

▷ Пусть Ω — язык первого порядка и T — непротиворечивая формальная аксиоматическая теория в языке Ω . Мы опишем некоторую модель M для теории T .

Ограничимся случаем, когда язык Ω содержит лишь счетное или конечное множество символов. Теорема верна и для произвольных языков первого порядка, причем и метод доказательства в основном остается тем же, но возникает необходимость в использовании некоторых специфически теоретико-множественных средств, например в так называемой трансфинитной индукции.

Пусть X — множество всех нелогических аксиом теории T .

Фиксируем счетное множество V констант, не входящих в язык Ω , и рассмотрим язык Ψ , получающийся Ω добавлением множества V в качестве множества новых констант. Через St_{Ψ} обозначим множество всех предложений в языке Ψ . Фиксируем некоторое предложение C языка Ω и определим $\top \Leftrightarrow C \vee \bigwedge C$ и $\perp \Leftrightarrow C \wedge \bigwedge C$. Пусть Γ и Δ — конечные множества формул языка Ψ , $\Gamma = \{A_1, \dots, A_m\}$, $\Delta = \{B_1, \dots, B_k\}$. Через $\Gamma \rightarrow \Delta$ обозначим формулу

$$\top \wedge A_1 \wedge \dots \wedge A_m \supset B_1 \vee \dots \vee B_k \vee \perp.$$

Если $k > 0$, $m > 0$, то эта формула эквивалентна $A_1 \wedge \dots \wedge A_m \supset B_1 \vee \dots \vee B_k$. Если $m > 0$, $k = 0$, то $\Gamma \rightarrow \Delta$ эквивалентна $\bigwedge (A_1 \wedge \dots \wedge A_m)$. Если $m = 0$, $k > 0$, то эта формула эквивалентна $B_1 \vee \dots \vee B_k$. Наконец, при $m = 0$, $k = 0$ эта формула эквивалентна \perp .

Пару множеств $Y, Z \subseteq St_{\Psi}$ назовем *совместной*, если для всяких конечных $\Gamma \subseteq Y$, $\Delta \subseteq Z$ формула $\Gamma \rightarrow \Delta$ не выводима в исчислении предикатов.

Пару множеств $Y, Z \subseteq St_{\Psi}$ назовем *полной*, если

- 1) $(A \wedge B) \in Y \Rightarrow A \in Y$ и $B \in Y$;
- 2) $(A \wedge B) \in Z \Rightarrow A \in Z$ или $B \in Z$;
- 3) $(A \vee B) \in Y \Rightarrow A \in Y$ или $B \in Y$;
- 4) $(A \vee B) \in Z \Rightarrow A \in Z$ и $B \in Z$;
- 5) $(A \supset B) \in Y \Rightarrow A \in Z$ или $B \in Y$;
- 6) $(A \supset B) \in Z \Rightarrow A \in Y$ и $B \in Z$;
- 7) $\bigwedge A \in Y \Rightarrow A \in Z$;
- 8) $\bigwedge A \in Z \Rightarrow A \in Y$;
- 9) $\forall x A(x) \in Y \Rightarrow$ для всякого замкнутого термина t языка Ψ , $A(t) \in Y$;
- 10) $\forall x A(x) \in Z \Rightarrow$ существует константа $c \in V$, $A(c) \in Z$;
- 11) $\exists x A(x) \in Y \Rightarrow$ существует константа $c \in V$, $A(c) \in Y$;
- 12) $\exists x A(x) \in Z \Rightarrow$ для всякого замкнутого термина t языка Ψ , $A(t) \in Z$.

Лемма. *Существует полная и совместная пара множеств Y, Z такая, что $X \subseteq Y$.*

▷ Множества Y, Z будем строить постепенно. На каждом этапе будут возникать множества Y_n, Z_n и мы определим $Y =$

$= \bigcup_{n \in \omega} Y_n, Z = \bigcup_{n \in \omega} Z_n$. При переходе к следующему этапу мы увеличиваем запас формул, так что $Y_n \subseteq Y_{n+1}, Z_n \subseteq Z_{n+1}$. Добавление формул производится таким образом, чтобы обеспечить условия полноты 1) — 12) в окончательных множествах Y, Z . Чтобы систематически обрабатывать все формулы, входящие в Y_n, Z_n , удобно занумеровать формулы натуральными числами и рассматривать их в порядке номеров.

Перейдем теперь к точным определениям. *Нумерованным множеством формул* назовем всякое множество F пар вида $\langle n, A \rangle$, где n — натуральное число и A — предложение языка Ψ . При этом выполняются условия:

а) $\langle n, A \rangle \in F, \langle n, B \rangle \in F \Rightarrow A = B$;

б) существует лишь конечное число нечетных n таких, что $\langle n, A \rangle \in F$.

Если $\langle n, A \rangle \in F$, то число n назовем *очередью* формулы A в множестве F . По условию, очередь полностью определяет формулу A в F . Кроме того, лишь конечное число формул имеет нечетную очередь.

По нумерованному множеству F можно образовать множество формул

$$F' = \{A \in \text{St}_\Psi \mid (\exists n \in \omega) (\langle n, A \rangle \in F)\}.$$

Нумерованная пара множеств есть по определению пара нумерованных множеств F, G такая, что из $\langle n, A \rangle \in F, \langle m, B \rangle \in G$ следует $n \neq m$. В этом случае $F \cup G$ есть нумерованное множество. Элемент $\langle n, A \rangle \in F \cup G$ с наименьшей очередью назовем *очередным* в F, G . *Вес* пары F, G есть натуральное число h , определяемое следующим образом. Если $F \cup G$ не содержит элементов с нечетной очередью, то $h = 0$. Если же $2k - 1$ есть наибольшая нечетная очередь в $F \cup G$, то определим $h = k$. Таким образом, если h вес F, G , то $F \cup G$ заведомо не содержит элементов с очередями $2h + 1, 2h + 3$ и т. д.

Занумеруем все замкнутые термы языка Ψ в единую последовательность:

$$t_1, t_2, \dots, t_n, \dots,$$

а также все константы из V — в последовательность

$$c_1, c_2, \dots, c_n, \dots.$$

Приступим к доказательству леммы. Построим нумерованную пару F_1, G_1 такую, что $F_1' = X, G_1' = \{C \wedge \neg C\}$, перенумеровав все формулы из F_1', G_1' четными числами. Вес h_1 этой пары равен нулю. Кроме того, каждая из формул $F_1' \cup G_1'$ не содержит констант из V , так как принадлежит языку Ω . Пара множеств F_1', G_1' совместна, так как множество X непротиворечно.

Далее, индуктивно определим последовательность нумерованных пар $F_1, G_1; F_2, G_2; \dots; F_n, G_n; \dots$ таким образом, что

каждое множество формул $F_n' \cup G_n'$ содержит лишь конечное множество констант из V и пара F_n', G_n' совместна. Вес пары F_n, G_n обозначим через h_n .

Итак, пусть уже построена пара F_n, G_n , покажем, как следует определить F_{n+1}, G_{n+1} . Пусть $\langle m, A \rangle$ есть очередной элемент пары F_n, G_n . Выбросим элемент $\langle m, A \rangle$ из того множества F_n или G_n , куда $\langle m, A \rangle$ входит, и добавим в это же множество новый элемент $\langle 2h_n + 1, A \rangle$. Полученную пару обозначим через P_n, Q_n . Очевидно,

$$P_n' = F_n', \quad Q_n' = G_n'.$$

Далее, разберем случаи в зависимости от вида элемента $\langle m, A \rangle$.

1) $\langle m, A \rangle \in F_n, A = A_1 \wedge A_2$. Тогда $F_{n+1} = P_n \cup \{ \langle 2h_n + 3, A_1 \rangle, \langle 2h_n + 5, A_2 \rangle \}$, $G_{n+1} = Q_n$.

Заметим, что пара F_{n+1}', G_{n+1}' совместна, так как в противном случае нашлись бы конечные множества $\Gamma \subseteq F_n', \Delta \subseteq G_n'$ такие, что $\vdash \{A_1, A_2\} \cup \Gamma \rightarrow \Delta$. Тогда по правилам исчисления предикатов $\vdash \{A_1 \wedge A_2\} \cup \Gamma \rightarrow \Delta$ и, так как $A_1 \wedge A_2 \in F_n'$, пара F_n', G_n' оказалась бы несовместной.

2) $\langle m, A \rangle \in G_n, A = A_1 \wedge A_2$.

Заметим, что одна из пар

$$F_n', G_n' \cup \{A_1\}$$

или

$$F_n', G_n' \cup \{A_2\}$$

необходимо совместна. Действительно, в противном случае найдутся конечные множества $\Gamma \subseteq F_n'$ и $\Delta \subseteq G_n'$ такие, что $\vdash \Gamma \rightarrow \Delta \cup \{A_1\}$ и $\vdash \Gamma \rightarrow \Delta \cup \{A_2\}$. А тогда по правилам исчисления предикатов $\vdash \Gamma \rightarrow \Delta \cup \{A_1 \wedge A_2\}$, и, так как $A_1 \wedge A_2 \in G_n'$, пара F_n', G_n' оказалась бы несовместной. Пусть совместна пара $F_n', G_n' \cup \{A_i\}$. Положим $F_{n+1} = P_n, G_{n+1} = Q_n \cup \{ \langle 2h_n + 3, A_i \rangle \}$.

3) $\langle m, A \rangle \in F_n, A = A_1 \vee A_2$.

Заметим, что одна из пар

$$\{A_1\} \cup F_n', G_n'$$

или

$$\{A_2\} \cup F_n', G_n'$$

необходимо совместна! Действительно, в противном случае найдутся конечные множества Γ и Δ , $\Gamma \subseteq F_n', \Delta \subseteq G_n'$, такие, что $\vdash \{A_1\} \cup \Gamma \rightarrow \Delta$ и $\vdash \{A_2\} \cup \Gamma \rightarrow \Delta$. А тогда $\vdash \{A_1 \vee A_2\} \cup \Gamma \rightarrow \Delta$, и пара F_n', G_n' оказалась бы несовместной. Пусть совместна пара $\{A_i\} \cup F_n', G_n'$. Определим $F_{n+1} = P_n \cup \{ \langle 2h_n + 3, A_i \rangle \}$, $G_{n+1} = Q_n$.

4) $\langle m, A \rangle \in G_n, A = A_1 \vee A_2$. Тогда $F_{n+1} = P_n, G_{n+1} = Q_n \cup \{ \langle 2h_n + 3, A_1 \rangle, \langle 2h_n + 5, A_2 \rangle \}$.

5) $\langle m, A \rangle \in F_n, A = A_1 \supset A_2$.

Заметим, что одна из пар

$$\{A_2\} \cup F_n', G_n'$$

или

$$F_n', G_n' \cup \{A_1\}$$

необходимо совместна. В противном случае для некоторых $\Gamma \subseteq F_n', \Delta \subseteq G_n'$ было бы $\vdash \{A_2\} \cup \Gamma \rightarrow \Delta$ и $\vdash \Gamma \rightarrow \Delta \cup \{A_1\}$. Но тогда $\vdash \{A_1 \supset A_2\} \cup \Gamma \rightarrow \Delta$ и пара F_n', G_n' оказалась бы несовместной.

Если совместна пара $\{A_2\} \cup F_n', G_n'$, то определим $F_{n+1} = P_n \cup \{\langle 2h_n + 3, A_2 \rangle\}$, $G_{n+1} = Q_n$. Если же совместна пара $F_n', G_n' \cup \{A_1\}$, то определим $F_{n+1} = P_n$, $G_{n+1} = Q_n \cup \{\langle 2h_n + 3, A_1 \rangle\}$.

6) $\langle m, A \rangle \in G_n, A = A_1 \supset A_2$. Тогда

$$F_{n+1} = P_n \cup \{\langle 2h_n + 3, A_1 \rangle\},$$

$$G_{n+1} = Q_n \cup \{\langle 2h_n + 5, A_2 \rangle\}.$$

7) $\langle m, A \rangle \in F_n, A = \neg A_1$. Тогда

$$F_{n+1} = P_n, G_{n+1} = Q_n \cup \{\langle 2h_n + 3, A_1 \rangle\}.$$

8) $\langle m, A \rangle \in G_n, A = \neg A_1$. Тогда

$$F_{n+1} = P_n \cup \{\langle 2h_n + 3, A_1 \rangle\}, G_{n+1} = Q_n.$$

9) $\langle m, A \rangle \in F_n, A = \forall x A_1(x)$. Пусть t_1, \dots, t_n — первые n термов в последовательности всех замкнутых термов языка Ψ . Тогда

$$F_{n+1} = P_n \cup \{\langle 2(h_n + j) + 1, A_1(t_j) \rangle \mid j = 1, \dots, n\}, G_{n+1} = Q_n.$$

10) $\langle m, A \rangle \in G_n, A = \forall x A_1(x)$. Выберем первую константу $a \in V$, не встречающуюся в $F_n' \cup G_n'$. Положим $F_{n+1} = P_n$, $G_{n+1} = Q_n \cup \{\langle 2h_n + 3, A_1(a) \rangle\}$. Заметим, что пара F'_{n+1}, G'_{n+1} совместна. В противном случае для некоторых $\Gamma \subseteq F_n', \Delta \subseteq G_n'$ было бы $\vdash \Gamma \rightarrow \Delta \cup \{A_1(a)\}$. Так как константа a не встречается в Γ и Δ , то в выводе $\vdash \Gamma \rightarrow \Delta \cup \{A_1(a)\}$ ее можно заменить на новую переменную z , не входящую в вывод, и получить вывод $\vdash \Gamma \rightarrow \Delta \cup \{A_1(z)\}$ и затем по правилам исчисления предикатов получить $\vdash \Gamma \rightarrow \Delta \cup \{\forall x A_1(x)\}$, что влечет, ввиду $\forall x A_1(x) \in G_n'$ несовместность пары F_n', G_n' .

11) $\langle m, A \rangle \in F_n, A = \exists x A_1(x)$. Выберем первую константу $a \in V$, не встречающуюся в $F_n' \cup G_n'$. Положим $F_{n+1} = P_n \cup \{\langle 2h_n + 3, A_1(a) \rangle\}$, $G_{n+1} = Q_n$. Заметим, что пара F'_{n+1}, G'_{n+1} совместна.

12) $\langle m, A \rangle \in G_n, A = \exists x A_1(x)$. Пусть t_1, \dots, t_n — первые n термов в последовательности всех замкнутых термов языка Ψ . Тогда $F_{n+1} = P_n$, $G_{n+1} = Q_n \cup \{\langle 2(h_n + j) + 1, A_1(t_j) \rangle \mid j = 1, \dots, n\}$.

Наконец, если для элемента $\langle m, A \rangle$ не выполняется ни один из случаев 1)–12), то определим $F_{n+1} = P_n$, $G_{n+1} = Q_n$. Таким образом, по сравнению с F_n и G_n в этом случае происходит лишь изменение очереди у очередного элемента, в то время как $F'_{n+1} = F_n$, $G'_{n+1} = G_n$.

Теперь остается положить

$$Y = \bigcup_{n \in \omega} F_n', Z = \bigcup_{n \in \omega} G_n'.$$

Каждая пара F_n', G_n' была совместной, откуда и следует совместность Y, Z . Нетрудно проверить и полноту Y, Z . Проверим, например, выполнение требования 6) полноты.

Пусть $(A \supset B) \in Z$. Тогда $\langle m, A \supset B \rangle \in G_n$ для некоторых m и n . Увеличивая в случае необходимости номер n , можно добиться, чтобы элемент $\langle m, A \supset B \rangle$ был очередным в G_n . По построению тогда $A \in F'_{n+1}, B \in G'_{n+1}$, т. е. $A \in Y$ и $B \in Z$. Лемма доказана. \square

Теперь мы можем описать искомую модель M . В качестве объектов модели возьмем множество всех выражений вида $[t]$, где t — произвольный замкнутый терм языка Ψ . Константе c языка Ω сопоставим объект $[c]$ модели M . Далее, m -местному функциональному символу f языка Ω сопоставим функцию \bar{f} в модели. А именно для произвольных объектов $[t_1], \dots, [t_m]$ модели M определим $\bar{f}([t_1], \dots, [t_m]) = [f(t_1, \dots, t_m)]$, где справа в квадратных скобках стоит замкнутый терм языка Ψ .

Наконец, m -местному предикатному символу P языка Ω сопоставим предикат \bar{P} модели M . А именно $\bar{P}([t_1], \dots, [t_m])$ истинно тогда и только тогда, когда формула $P(t_1, \dots, t_m)$ принадлежит множеству Y , указанному в лемме..

Описание M закончено. Необходимо проверить, что предложения из множества X истинны в M . С этой целью докажем следующие два утверждения, устанавливающие связь множеств Y, Z с истинностью в модели M . Пусть

A — формула, оцененная в модели M , и

A^* — предложение языка Ψ , которое получается из A стиранием всех квадратных скобок. Тогда

а) $A^* \in Y \Rightarrow M \models A$,

б) $A^* \in Z \Rightarrow$ неверно, что $M \models A$.

Оба утверждения будем доказывать одновременно индукцией по построению формулы A .

Если A — атомарна и $A^* \in Y$, то $M \models A$ по определению истинности в M . Если $A^* \in Z$, то ввиду совместности Y, Z имеем $A^* \notin Y$, и значит, не $M \models A$.

Пусть $A = A_1 \wedge A_2$. Если $A^* \in Y$, то ввиду полноты $A_1^*, A_2^* \in Y$. Тогда по индуктивному предположению $M \models A_1, M \models A_2$, т. е. $M \models A_1 \wedge A_2$. Если $A^* \in Z$, то, опять-таки ввиду полноты, $A_1^* \in Z$ или $A_2^* \in Z$. По индуктивному предположению тогда не $M \models A_1$ или не $M \models A_2$, что означает не $M \models A_1 \wedge A_2$.

Подобным образом рассматриваются и остальные случаи строения A . Рассмотрим еще, например, случай, когда $A = \exists x A_1(x)$. Если $A^* \in Y$, то ввиду полноты $A_1^*(a) \in Y$ для некоторой константы из V . По индуктивному предположению $M \models A_1([a])$, т. е. $M \models \exists x A_1(x)$. Если $A^* \in Z$, то ввиду полноты $A_1^*(t) \in Z$ для всякого замкнутого термина t языка Ψ . По индуктивному предположению не $M \models A_1([t])$ для всякого t и, следовательно, не $M \models \exists x A_1(x)$.

Теперь, так как $X \subseteq Y$, из утверждения а) следует, что $M \models A$ для всех $A \in X$ и, значит, M есть модель теории T . \square

2. Отметим некоторые следствия основной теоремы п. 1.

Теорема (Мальцева о компактности). Пусть X — множество предложений языка Ω . Если каждое конечное подмножество множества X имеет модель, то и все множество X также имеет модель.

▷ В самом деле, в условиях теоремы X образует непротиворечивую аксиоматическую теорию, так как если бы в X выводилось противоречие, то нашлось бы конечное множество $\Gamma \subseteq X$, $\Gamma \vdash C \wedge \neg C$ и конечное множество Γ не имело бы модели. По основной теореме X имеет модель. □

Теорема (о полноте исчисления предикатов). Если формула A есть логический закон, то A выводится в исчислении предикатов.

▷ Можно считать, что A есть предложение, замыкая в случае необходимости A кванторами общности. Если A не выводится в исчислении предикатов, то теория T с единственной нелогической аксиомой $\neg A$ непротиворечива (так как из $\neg A \vdash C \wedge \neg C$ следует $\vdash A$). По основной теореме тогда $\neg A$ имеет модель M . Но тогда неверно, что $M \models A$, и, значит, A не есть логический закон. □

3. Добавим к языку $A\gamma$ новую константу c и к теории $A\gamma$ добавим бесконечную серию нелогических аксиом $0 < c, \bar{1} < c, \bar{2} < c, \dots$. Здесь \bar{n} есть терм вида $SS \dots S0$, соответствующий натуральному числу n .

Каждая конечная подтеория полученной теории T имеет модель. В самом деле, если в конечную подтеорию входят лишь аксиомы $\bar{n} < c$ с числами n , меньшими j , то достаточно взять стандартную модель ω и сопоставить константе c число $(j+1)$.

По теореме о компактности вся теория T имеет модель. Эта модель будет, конечно, и моделью $A\gamma$, но это будет *нестандартная модель*, неизоморфная ω . Действительно, константе c в этой модели соответствует объект, больший, чем все стандартные объекты, соответствующие обычным натуральным числам $0, 1, 2, \dots$.

Заметим, что это имеет место, несмотря на известную теорему о категоричности натурального ряда. Мы уже обсуждали этот эффект в главе о теории множеств.

4. Если язык первого порядка Ω содержит не более чем счетное множество символов, то модель M для непротиворечивой теории T , описанная в основной теореме, оказывается счетной. Отсюда следует факт, являющийся вариантом так называемой теоремы Левенгейма—Сколема.

Теорема. Пусть Ω — язык первого порядка со счетным множеством символов и T — непротиворечивая теория в языке Ω . Тогда теория T имеет модель со счетным носителем.

В частности, если Ω — язык первого порядка со счетным множеством символов и M — произвольная модель для языка

Ω , то существует и счетная модель M' такая, что

$$\text{Th}_\Omega(M) = \text{Th}_\Omega(M').$$

Здесь через $\text{Th}_\Omega(M)$ мы обозначаем множество всех предложений языка Ω , истинных в модели M .

В общепринятой терминологии это замечание можно сформулировать следующим образом: *всякая модель в счетном языке элементарно-эквивалентна счетной модели*.

5. Из полученных результатов следует, что если теория множеств ZF непротиворечива, то она имеет счетную модель. И это несмотря на то, что в ZF можно доказать существование несчетного множества! Это и есть так называемый *парадокс Сколема*, который мы уже обсуждали (гл. I, § 6, п. 5). Еще раз подчеркнем, что указанное обстоятельство не является парадоксом в собственном смысле этого слова и никакого противоречия в полученном результате нет.

§ 3. ТЕОРЕМА ОБ УСТРАНЕНИИ СЕЧЕНИЯ

1. Ключевую роль в теории доказательств играет понятие выводимости в исчислении предикатов. Поэтому важно преобразовать выводы в исчислении предикатов к виду, удобному для обнаружения выводимости. Это существенно и потому, что в настоящее время актуальна задача автоматического поиска вывода в аксиоматических теориях с привлечением вычислительных машин. Основой большинства методов поиска вывода в исчислении предикатов является знаменитая теорема Г. Генцена об устранении сечения, полученная в 1934 году, вариант которой мы изложим в этом параграфе.

Фиксируем некоторый логико-математический язык Ω .

Набором формул мы назовем конечное множество формул языка Ω , в котором допускаются повторения формул. Порядок формул в наборе Γ не имеет значения, но для каждой формулы указано, в скольких экземплярах она присутствует в Γ . В соответствии с этим следует трактовать и операции над наборами. Например, при объединении ГУД наборов количество экземпляров каждой формулы суммируется. Объединение ГУД мы будем коротко записывать в виде $\Gamma\Delta$, так что $\Gamma\Delta$ и $\Delta\Gamma$ есть один и тот же набор. Набор $A\Gamma$ получается из набора Γ присоединением одного экземпляра формулы A .

Секвенцией назовем фигуру вида $\Gamma \rightarrow \Delta$, где Γ и Δ — наборы формул.

Сформулируем теперь некоторое исчисление, в котором будут выводиться секвенции. *Аксиомы* этого исчисления суть произвольные секвенции вида

$$A\Gamma \rightarrow \Delta A,$$

где A — произвольная атомарная формула Ω .

Правила вывода исчисления секвенций построены весьма симметрично и вводят логические связки слева и справа:

$$\begin{aligned}
 (\supset \rightarrow) & \frac{B\Gamma \rightarrow \Delta; \Gamma \rightarrow \Delta A}{(A \supset B)\Gamma \rightarrow \Delta}; (\rightarrow \supset) \frac{A\Gamma \rightarrow \Delta B}{\Gamma \rightarrow \Delta (A \supset B)}; \\
 (\wedge \rightarrow) & \frac{AB\Gamma \rightarrow \Delta}{(A \wedge B)\Gamma \rightarrow \Delta}; (\rightarrow \wedge) \frac{\Gamma \rightarrow \Delta A; \Gamma \rightarrow \Delta B}{\Gamma \rightarrow \Delta (A \wedge B)}; \\
 (\vee \rightarrow) & \frac{A\Gamma \rightarrow \Delta; B\Gamma \rightarrow \Delta}{(A \vee B)\Gamma \rightarrow \Delta}; (\rightarrow \vee) \frac{\Gamma \rightarrow \Delta AB}{\Gamma \rightarrow \Delta (A \vee B)}; \\
 (\neg \rightarrow) & \frac{\Gamma \rightarrow \Delta A}{\neg A\Gamma \rightarrow \Delta}; (\rightarrow \neg) \frac{A\Gamma \rightarrow \Delta}{\Gamma \rightarrow \Delta \neg A}; \\
 (\forall \rightarrow) & \frac{A(x \parallel t) \forall x A\Gamma \rightarrow \Delta}{\forall x A\Gamma \rightarrow \Delta}; (\rightarrow \forall) \frac{\Gamma \rightarrow \Delta (A(y \parallel x))}{\Gamma \rightarrow \Delta \forall y A}; \\
 (\exists \rightarrow) & \frac{A(y \parallel x) \Gamma \rightarrow \Delta}{\exists y A\Gamma \rightarrow \Delta}; (\rightarrow \exists) \frac{\Gamma \rightarrow \Delta \exists x A (A(x \parallel t))}{\Gamma \rightarrow \Delta \exists x A}.
 \end{aligned}$$

Здесь в правилах $(\rightarrow \forall)$, $(\exists \rightarrow)$ переменная x не входит свободно в нижнюю секвенцию рассматриваемого правила вывода.

Выводы в исчислении секвенций мы будем записывать в виде деревьев аналогично выводам в исчислении предикатов (см. введение). Высота вывода по-прежнему есть количество секвенций в наиболее длинной ветви вывода. Кроме того, мы не будем делать различия между формулами и секвенциями, отличающимися лишь переименованием связанных переменных. Таким образом, если имеется вывод с нижней секвенцией S , то этот же вывод считается автоматическим и выводом любой секвенции, полученной из S переименованием связанных переменных.

Будем писать $\vdash \Gamma \rightarrow \Delta$, если секвенция $\Gamma \rightarrow \Delta$ выводима в нашем исчислении секвенций.

Выводы в исчислении секвенций имеют замечательную особенность: секвенции, расположенные выше всякого правила вывода, состоят лишь из подформул формул в секвенции, расположенной в заключении правила вывода. Коротко говоря, при рассмотрении правил вывода «снизу вверх» формулы в секвенциях лишь дробятся, никакие посторонние формулы не появляются. В этом и состоит так называемое *свойство подформульности* исчисления секвенций. Это обстоятельство во многих случаях чрезвычайно облегчает поиск вывода данной секвенции.

Основной результат параграфа состоит в том, что описанное исчисление секвенций эквивалентно обычному исчислению предикатов, и таким образом поиск вывода в исчислении предикатов эквивалентен поиску вывода в нашем исчислении секвенций. Для доказательства основного результата нам придется подробно исследовать исчисление секвенций. Ключевым

фактом здесь является теорема об *устранении правила сечения*. Так называется следующее правило вывода:

$$\frac{\Gamma \rightarrow \Delta A; A\Gamma \rightarrow \Delta}{\Gamma \rightarrow \Delta}.$$

Теорема утверждает, что это правило *допустимо* в исчислении секвенций, т. е. если в исчислении секвенций выводимы посылки $\Gamma \rightarrow \Delta A$ и $A\Gamma \rightarrow \Delta$ этого правила, то необходимо выводима и секвенция $\Gamma \rightarrow \Delta$, являющаяся заключением рассматриваемого правила. Таким образом, если в выводах использовать наряду с другими и правило сечения, то теорема утверждает, что все примененные правила сечения можно затем устранить и получить вывод без сечений.

Заметим, что наша версия исчисления секвенций несколько отличается от той, которую использовал Генцен (см. [15]).

2. Л е м м а. Для всякой формулы A языка Ω имеем

$$\vdash A\Gamma \rightarrow \Delta A.$$

▷ Доказательство проведем индукцией по количеству логических символов в A . Если A — атомарная формула, то наша секвенция является просто аксиомой. Рассмотрим несколько случаев индукции, оставляя остальные случаи читателю.

Пусть $A = B \wedge C$. По индуктивному предположению $\vdash B\Gamma \rightarrow \Delta B$ и $\vdash B\Gamma \rightarrow \Delta C$. Тогда по правилу $(\rightarrow \wedge)$ имеем $\vdash B\Gamma \rightarrow \Delta(B \wedge C)$ и затем по правилу $(\wedge \rightarrow) \vdash (B \wedge C)\Gamma \rightarrow \Delta(B \wedge C)$.

Пусть $A = \forall x B(x)$. Пусть z — переменная, не входящая свободно в секвенцию $\forall x B(x)\Gamma \rightarrow \Delta \forall x B(x)$. По индуктивному предположению $\vdash B(z)\forall x B(x)\Gamma \rightarrow \Delta B(z)$. По правилу $(\forall \rightarrow)$ отсюда имеем $\vdash \forall x B(x)\Gamma \rightarrow \Delta B(z)$ и затем по правилу $(\rightarrow \forall)$ окончательно получаем $\vdash \forall x B(x)\Gamma \rightarrow \Delta \forall x B(x)$. □

3. Если S — секвенция, то через $S(x\|t)$ обозначим результат правильной подстановки терма t вместо свободных вхождений переменной x в каждую формулу, член этой секвенции.

Л е м м а. Если $\vdash S$, z — переменная и t — терм, то $\vdash S(z\|t)$, причем вывод результирующей секвенции имеет ту же высоту и то же количество секвенций, что и данный вывод.

▷ Доказательство проведем индукцией по построению данного вывода $\vdash S$. Если S — аксиома, то $S(z\|t)$ — также аксиома.

Далее следует рассмотреть все случаи, когда S получена по одному из правил вывода. Рассмотрим для примера, лишь случай, когда S получена по правилу $(\rightarrow \forall)$. Тогда S имеет вид $\Gamma \rightarrow \Delta \forall y A$ и получена из секвенции вида $\Gamma \rightarrow \Delta A(y\|x)$, где переменная x не входит свободно в S .

Выберем переменную u , отличную от всех переменных, фигурирующих в рассматриваемых формулах и термах. По индуктивному предположению из выводимости $\Gamma \rightarrow \Delta A(y\|x)$ следует выводимость секвенции $\Gamma \rightarrow \Delta (A(y\|x)(x\|u))$, т. е. секвен-

ции $\Gamma \rightarrow \Delta A(y||u)$. Вновь используя индуктивное предположение, получим, что выводима секвенция $\Gamma(z||t) \rightarrow \Delta(z||t) A(y||u)(z||t)$. Применяя затем правило $(\rightarrow \forall)$, заключаем, что $\vdash \Gamma(z||t) \rightarrow \Delta(z||t) \forall u(A(y||u)(z||t))$. Но с точностью до переименования связанных переменных формула $\forall u(A(y||u)(z||t))$ совпадает с $(\forall yA)(z||t)$. Таким образом, полученный вывод доставляет и вывод секвенции

$$\Gamma(z||t) \rightarrow \Delta(z||t) (\forall yA)(z||t). \quad \square$$

4. Лемма. В исчислении секвенций допустимы следующие правила добавления:

$$\frac{\Gamma \rightarrow \Delta}{A\Gamma \rightarrow \Delta}; \quad \frac{\Gamma \rightarrow \Delta}{\Gamma \rightarrow \Delta A}.$$

Допустимость означает, что из выводимости секвенции $\Gamma \rightarrow \Delta$ вытекает, что и обе секвенции $A\Gamma \rightarrow \Delta$ и $\Gamma \rightarrow \Delta A$ также выводимы. Более того, можно утверждать при этом, что результирующие выводы имеют не большую высоту и не больше секвенций в своем составе, чем данный вывод.

\Rightarrow Доказательство ведем индукцией по высоте данного вывода $\vdash \Gamma \rightarrow \Delta$. Если $\Gamma \rightarrow \Delta$ — аксиома, то аксиомами являются и обе секвенции $A\Gamma \rightarrow \Delta$, $\Gamma \rightarrow \Delta A$. Далее следует разобрать все случаи, когда $\Gamma \rightarrow \Delta$ получена по одному из правил вывода. Пусть, например, $\Gamma \rightarrow \Delta$ получена по правилу $(\exists \rightarrow)$. В этом случае $\Gamma \rightarrow \Delta$ имеет вид $\exists y B\Gamma' \rightarrow \Delta$ и получена из секвенции вида $B(y||x)\Gamma' \rightarrow \Delta$. Выберем новую переменную u , не фигурирующую в рассматриваемых секвенциях. По предыдущей лемме имеем $\vdash B(y||x)(x||u)\Gamma' \rightarrow \Delta$, т. е. $\vdash B(y||u)\Gamma' \rightarrow \Delta$, причем высота и количество секвенций в полученном выводе не меняются. По индуктивному предположению далее $\vdash B(y||u)A\Gamma' \rightarrow \Delta$ и $\vdash B(y||u)\Gamma' \rightarrow \Delta A$. Отсюда по правилу $(\exists \rightarrow)$ выводим $\exists u(B(y||u))A\Gamma' \rightarrow \Delta$, $\exists u(B(y||u))\Gamma' \rightarrow \Delta A$, что с точностью до переименования связанных переменных совпадает с искомыми секвенциями. Переименование переменной x в новую переменную u в этом рассуждении нам понадобилось, чтобы после добавления формулы A выполнялись бы ограничения на переменные в применении правила $(\exists \rightarrow)$. \square

5. Лемма. В исчислении секвенций обратимы все правила вывода. Это означает, что если выводимо заключение какого-либо правила вывода нашего исчисления секвенций, то выводимы и (обе) его посылки. При этом вновь можно утверждать, что результирующие выводы имеют не большую высоту и не большее количество секвенций, чем данный вывод.

\Rightarrow Для правил $(\forall \rightarrow)$ и $(\rightarrow \exists)$ это следует непосредственно из допустимости правила добавления (п. 4). Для каждого из остальных правил вывода лемму доказываем отдельно индукцией по высоте вывода заключения.

Рассмотрим, например, правило $(\supset \rightarrow)$. Пусть дано $\vdash (A \supset B)\Gamma \rightarrow \Delta$, докажем, что $\vdash B\Gamma \rightarrow \Delta$ и $\vdash \Gamma \rightarrow \Delta A$ индукцией по вы-

соте данного вывода. Если исходный вывод есть аксиома, то обе результирующие секвенции также являются аксиомами (здесь существенно, что формула $A \supset B$ не атомарная, в то время как формулы, фигурирующие явно в формулировке аксиом, атомарны). Далее следует рассмотреть случай, когда секвенция $(A \supset B)\Gamma \rightarrow \Delta$ получена по одному из правил вывода. Здесь существенны два подслучая: а) указанное правило не относится к явно выписанной формуле $A \supset B$; б) когда это правило относится к указанной формуле.

В случае а) рассуждения однотипны. Пусть, например, $(A \supset B)\Gamma \rightarrow \Delta$ получена по правилу $(\rightarrow \forall)$. Тогда эта секвенция имеет вид $(A \supset B)\Gamma \rightarrow \Delta' \forall y C$ и получена из секвенции $(A \supset B)\Gamma \rightarrow \Delta' C(y \| x)$. По индуктивному предположению выводимы секвенции $B\Gamma \rightarrow \Delta' (C(y \| x))$ и $\Gamma \rightarrow \Delta' A(C(y \| x))$. Применяя к ним правило $(\rightarrow \forall)$, получим искомые секвенции.

В случае б) утверждение тривиально. Искомые секвенции являются посылками правила вывода $(\supset \rightarrow)$ и, следовательно, выводимы. \square

6. Лемма. В исчислении секвенций допустимы правила сокращения:

$$\frac{A\Gamma \rightarrow \Delta, \Gamma \rightarrow \Delta A A}{A\Gamma \rightarrow \Delta, \Gamma \rightarrow \Delta A},$$

т. е. из выводимости посылок этих правил следует и выводимость их заключений, причем опять-таки высота и количество секвенций в выводах заключений не превосходят этих же показателей в выводах соответствующих посылок.

\triangleright Для всякого натурального n установим следующее утверждение: если дан вывод секвенции S такой, что либо слева, либо справа в S повторяется некоторая формула A (т. е. S имеет вид $A\Gamma \rightarrow \Delta$ или $\Gamma \rightarrow \Delta A A$), причем данный вывод имеет высоту $\leq n$, то выводима и секвенция S' , полученная заменой двух отмеченных экземпляров формулы A одним экземпляром (т. е. S' имеет вид $A\Gamma \rightarrow \Delta$ или соответственно $\Gamma \rightarrow \Delta A$); при этом вывод S' имеет не большую высоту и не большее количество секвенций, чем данный вывод S .

Указанное утверждение докажем индукцией по n . При $n=1$ секвенция S оказывается аксиомой. Но тогда S' также, очевидно, является аксиомой.

Далее следует разобрать все случаи, когда секвенция S получена по одному из правил вывода. Здесь существенны два подслучая:

а) правило, по которому получена S , не касается ни одного из рассматриваемых экземпляров сокращаемой формулы A ;
б) правило, по которому получена S , касается одного из экземпляров сокращаемой формулы A .

В случае а) рассуждения однотипны. Пусть например, S имеет вид $A\Gamma \rightarrow \Delta' (C \supset D)$ и получена по правилу $(\rightarrow \supset)$ из секвенции $A A \Gamma \rightarrow \Delta' D$. Из выводимости этой последней сек-

венции и индуктивного предположения следует, что выводима секвенция $AC\Gamma \rightarrow \Delta'D$. Затем, применяя правило $(\rightarrow \supset)$ к этой секвенции, получим вывод секвенции S' .

В случае *b*) существенна обратимость правил вывода. Допустим, например, что A имеет вид $(C \supset D)$ и секвенция S имеет вид $(C \supset D)(C \supset D)\Gamma \rightarrow \Delta$ и получена по правилу вывода $(\supset \rightarrow)$ из двух секвенций $D(C \supset D)\Gamma \rightarrow \Delta$ и $(C \supset D)\Gamma \rightarrow \Delta C$. Используя обратимость правила $(\supset \rightarrow)$, из выводимости последних двух секвенций заключаем, что выводимы и секвенции $DD\Gamma \rightarrow \Delta$ и $\Gamma \rightarrow \Delta CC$, причем с помощью выводов не большей сложности, чем соответствующие выводы предыдущих секвенций. По индуктивному предположению отсюда заключаем, что выводимы и секвенции $D\Gamma \rightarrow \Delta$ и $\Gamma \rightarrow \Delta C$. По правилу $(\supset \rightarrow)$ отсюда получим вывод $(C \supset D)\Gamma \rightarrow \Delta$.

Рассмотрим еще случай, когда секвенция S имеет вид $\Gamma \rightarrow \Delta \exists yB(y) \exists yB(y)$ и получена из секвенции $\Gamma \rightarrow \Delta B(t) \exists yB(y) \exists yB(y)$ по правилу вывода $(\rightarrow \exists)$. Тогда по индуктивному предположению $\vdash \Gamma \rightarrow \Delta B(t) \exists yB(y)$ и к этой последней секвенции достаточно применить правило $(\rightarrow \exists)$, чтобы получить вывод S' . \square

7. Теорема Генцена. В исчислении секвенций допустимо правило сечения:

$$\frac{\Gamma \rightarrow \Delta A; A\Gamma \rightarrow \Delta}{\Gamma \rightarrow \Delta}.$$

\triangleright Утверждение докажем индукцией по количеству логических связок в формуле A . При фиксированной сложности A утверждение будем доказывать индукцией по сумме высот данных выводов $\Gamma \rightarrow \Delta A$ и $A\Gamma \rightarrow \Delta$.

Начало этой двойной индукции состоит в рассмотрении случая, когда A — атомарная формула и обе выводимые секвенции суть аксиомы. Мы утверждаем, что в этом случае $\Gamma \rightarrow \Delta$ также является аксиомой. В самом деле, если $A \in \Gamma$ и $A \in \Delta$, то это очевидно. Пусть, например, $A \notin \Gamma$. Тогда ввиду того что $\Gamma \rightarrow \Delta A$ является аксиомой, найдется атомарная формула B такая, что $B \in \Gamma$ и $B \in \Delta$. Но тогда $\Gamma \rightarrow \Delta$ является аксиомой.

Разберем теперь несколько случаев индукции.

Пусть A имеет вид $(C \supset D)$ и дано $\vdash \Gamma \rightarrow \Delta(C \supset D)$ и $\vdash (C \supset D)\Gamma \rightarrow \Delta$. Ввиду обратимости правил $(\rightarrow \supset)$ и $(\supset \rightarrow)$ выводимы секвенции $C\Gamma \rightarrow \Delta D$, $D\Gamma \rightarrow \Delta$, $\Gamma \rightarrow \Delta C$. Применяя правило добавления к $D\Gamma \rightarrow \Delta$, получим $\vdash DC\Gamma \rightarrow \Delta$. Из $\vdash C\Gamma \rightarrow \Delta D$ и $\vdash DC\Gamma \rightarrow \Delta$ по правилу сечения (его можно применить в силу индукции по сложности формулы) получим $\vdash C\Gamma \rightarrow \Delta$. Затем применим правило сечения с секвенцией $\vdash \Gamma \rightarrow \Delta C$ и получим $\vdash \Gamma \rightarrow \Delta$.

Пусть A имеет вид $\exists xB(x)$ и дано $\vdash \Gamma \rightarrow \Delta \exists xB(x)$ и $\vdash \exists xB(x)\Gamma \rightarrow \Delta$. Если $\Gamma \rightarrow \Delta \exists xB(x)$ есть аксиома, то $\Gamma \rightarrow \Delta$ — также аксиома, и поэтому $\vdash \Gamma \rightarrow \Delta$ (мы пользуемся тем, что $\exists xB(x)$ — не атомарная формула). Пусть $\Gamma \rightarrow \Delta \exists xB(x)$ полу-

чена по некоторому правилу вывода. Здесь следует разобрать два подслучая: *a*) это правило вывода не относится к рассматриваемому вхождению $\exists xB(x)$ и *b*) это правило вывода есть $(\rightarrow \exists)$, относящееся к рассматриваемому вхождению $\exists xB(x)$.

В случае *a*) рассуждения однотипны для всех правил. Пусть, например, $\Gamma \rightarrow \Delta \exists xB(x)$ имеет вид $\Gamma \rightarrow \Delta' \neg D \exists xB(x)$ и получена из $D\Gamma \rightarrow \Delta' \exists xB(x)$ по правилу $(\rightarrow \neg)$. Из данной выводимости $\vdash \exists xB(x) \Gamma \rightarrow \Delta' \neg D$ ввиду обратимости правила $(\rightarrow \neg)$ заключаем $\vdash \exists xB(x) D\Gamma \rightarrow \Delta'$, причем высота этого последнего вывода не больше, чем высота вывода секвенции $\exists xB(x) \Gamma \rightarrow \Delta' \neg D$. По индуктивному предположению (индукции по сумме высот выводов) применим сечение к $D\Gamma \rightarrow \Delta' \exists xB(x)$ и $\exists xB(x) D\Gamma \rightarrow \Delta'$ и получим в результате $\vdash D\Gamma \rightarrow \Delta'$. После этого применим $(\rightarrow \neg)$ и получим $\vdash \Gamma \rightarrow \Delta$.

В случае *b*) секвенция $\Gamma \rightarrow \Delta \exists xB(x)$ получена по правилу $(\rightarrow \exists)$ из секвенции $\Gamma \rightarrow \Delta B(t) \exists xB(x)$. Из данного вывода $\vdash \exists xB(x) \Gamma \rightarrow \Delta$ по правилу добавления получим $\vdash \exists xB(x) \Gamma \rightarrow \rightarrow \Delta B(t)$, причем высота результирующего вывода не увеличивается. По индуктивному предположению (индукции по сумме высот выводов) применим сечение к

$$\Gamma \rightarrow \Delta B(t) \exists xB(x) \text{ и } \exists xB(x) \Gamma \rightarrow \Delta B(t)$$

и получим $\vdash \Gamma \rightarrow \Delta B(t)$.

Далее, из $\vdash \exists xB(x) \Gamma \rightarrow \Delta$ ввиду обратимости правила $(\exists \rightarrow)$ получим $\vdash B(y) \Gamma \rightarrow \Delta$ с новой переменной y . Затем по лемме о подстановке (п. 3) отсюда получим $\vdash B(t) \Gamma \rightarrow \Delta$. По предположению индукции (индукции по сложности формулы) можно применить сечение к $\Gamma \rightarrow \Delta B(t)$ и $B(t) \Gamma \rightarrow \Delta$ и таким образом получить $\vdash \Gamma \rightarrow \Delta$. \square

8. Теперь мы готовы установить эквивалентность исчисления предикатов и нашего исчисления секвенций. Эта эквивалентность формулируется в двух следующих леммах.

Лемма. Если в исчислении предикатов выводится формула A , то в исчислении секвенций выводится секвенция $\rightarrow A$.

\triangleright Доказательство проведем индукцией по построению вывода $\vdash A$ в исчислении предикатов. Сначала следует вывести в исчислении секвенций все аксиомы исчисления предикатов. Это несложное упражнение на выводимость в исчислении секвенций. Рассмотрим лишь два примера.

Установим $\vdash \rightarrow (A \supset C) \supset (B \supset C) \supset (A \vee B \supset C)$.

В самом деле, по лемме п. 2 $\vdash (A \supset C), (B \supset C) \rightarrow (A \supset C)$. Ввиду обратимости правила $(\rightarrow \supset)$ тогда

$$\vdash A \supset C, B \supset C, A \rightarrow C.$$

Аналогично имеем

$$\vdash A \supset C, B \supset C, B \rightarrow C.$$

Из этих двух секвенций по правилу $(\vee \rightarrow)$ выводим

$$\vdash A \supset C, B \supset C, A \vee B \rightarrow C.$$

Применяя несколько раз правило $(\rightarrow\supset)$, выведем требуемую секвенцию.

Установим $\vdash \rightarrow \forall x(C \supset A(x)) \supset. C \supset \forall x A(x)$, где C не содержит свободно x . По лемме п. 2

$$\vdash \forall x(C \supset A(x)), C \supset A(x) \rightarrow C \supset A(x).$$

По правилу $(\forall \rightarrow)$ отсюда

$$\vdash \forall x(C \supset A(x)) \rightarrow C \supset A(x).$$

Ввиду обратимости правила $(\rightarrow\supset)$ имеем

$$\vdash \forall x(C \supset A(x)), C \rightarrow A(x).$$

Применяя $(\rightarrow\forall)$, получим

$$\vdash \forall x(C \supset A(x)), C \rightarrow \forall x A(x)$$

(заметим, что ограничения на переменные в этом правиле выполнены именно потому, что C не содержит свободно x). Применяя к последней секвенции несколько раз правило $(\rightarrow\supset)$, получим выводимость требуемой секвенции.

Далее необходимо проверить, что правила вывода исчисления предикатов сохраняют выводимость в исчислении секвенций. Пусть, например, $\vdash \rightarrow A$ и $\vdash \rightarrow (A \supset B)$. Покажем, что в этой ситуации $\vdash \rightarrow B$.

Ввиду обратимости правила $(\rightarrow\supset)$ из $\vdash \rightarrow (A \supset B)$ следует $\vdash A \rightarrow B$. Из $\vdash \rightarrow A$ по правилу добавления получим $\vdash \rightarrow B, A$. Теперь остается применить сечение к выводимым секвенциям $\rightarrow B, A$ и $A \rightarrow B$, в результате чего и получим секвенцию $\rightarrow B$. \square

Для каждой секвенции $\Gamma \rightarrow \Delta$ определим некоторую формулу $(\Gamma \rightarrow \Delta)^0$ — перевод секвенции в формулу исчисления предикатов. А именно перевод $A_1, \dots, A_n \rightarrow B_1, \dots, B_m$ есть по определению формула $A_1 \wedge \dots \wedge A_n \supset B_1 \vee \dots \vee B_m$. Перевод секвенции $\rightarrow B_1, \dots, B_m$ есть формула $B_1 \vee \dots \vee B_m$. Перевод секвенции $A_1, \dots, A_n \rightarrow$ есть формула $\neg(A_1 \wedge \dots \wedge A_n)$. Наконец, перевод секвенции \rightarrow есть формула $A \wedge \neg A$ для некоторой фиксированной формулы A .

Лемма. Если секвенция S выводится в исчислении секвенций, то формула S^0 выводится в исчислении предикатов.

[> Необходимо убедиться, что перевод всякой аксиомы исчисления секвенций выводится в исчислении предикатов и что перевод всякого правила вывода допустим в исчислении предикатов. Проверить все эти факты можно с помощью техники естественного вывода в исчислении предикатов (см. нашу книгу [1]), но можно воспользоваться и теоремой о полноте исчисления предикатов и убедиться, что если переводы посылок некоторого правила вывода исчисления секвенций являются логическими законами, то логическим законом является и заключение. \square

§ 4. О ПРОГРАММЕ ГИЛЬБЕРТА ОБОСНОВАНИЯ МАТЕМАТИКИ

1. Пример наивной теории множеств показывает, что привлекательная математическая теория может оказаться внутренне противоречивой. Причины этого можно искать в своеобразии

- а) объектов исследования в математике,
- б) способов рассуждения относительно этих объектов.

Объекты исследования в математике не являются, как правило, экспериментально наблюдаемыми, это мысленные объекты, возникающие как результат сложной многоступенчатой абстракции действительности. В самом деле, в каком именно смысле существуют иррациональные числа, неизмеримые множества действительных чисел, функция Дирихле, нигде не плотное совершенное множество Кантора и т. п.?

Способ рассуждения относительно таких сложных объектов также получается путем сложной экстраполяции способов рассуждения, применяемых в обыденной жизни или по крайней мере в обычном научном обиходе.

Однако так применяемые способы рассуждения по отношению к мысленным объектам могут вести к неожиданным следствиям. Например, математик, склонен, по-видимому, считать, что множество точек в трехмерном координатном евклидовом пространстве $\{ \{x, y, z\} \mid \sqrt{x^2 + y^2 + z^2} \leq 1 \}$ служит хорошей моделью геометрического тела — шара в трехмерном пространстве. Тем не менее в ZFC можно доказать, что этот «шар» можно разбить на четыре подмножества, из которых с помощью движений в евклидовом пространстве можно сложить два шара, равных первоначальному!

Такого рода следствия вызывают подозрение, что ряд фактов, полученных в рамках определенной математической теории, даже непротиворечивой, просто не имеет никакого отношения к физической реальности и является результатом слишком далеко зашедшей экстраполяции! Это, в свою очередь, вызывает ряд трудных философских вопросов, касающихся ценности математического рассуждения, убедительности математического рассуждения, соответствия между установленными математическими фактами и законами окружающего нас мира.

Далее, может вызывать беспокойство принципиальная неполнота математических теорий. Как мы уже отмечали, доказано, что многие интересные теоретико-множественные утверждения (например, континуум-гипотеза) не могут быть ни выведены, ни опровергнуты в системе ZFC. При этом не видно никаких интуитивно очевидных принципов, которые следовало бы добавить к ZFC, чтобы решить вопрос об истинности таких утверждений. Напротив, эксперименты показывают, что можно предложить равно содержательные и математически интересные аксиоматики, расширяющие ZFC, в которых вопрос об ис-

тинности рассматриваемых утверждений решается по-разному. Это вновь наводит на мысль, что принципиально неверно ставить вопрос о том, истинна ли континуум-гипотеза «на самом деле». Никакого «самом деле» нет. Континуум-гипотеза есть утверждение о мысленных объектах, и вопрос об ее истинности существенно зависит от способов рассуждения, применяемых к таким объектам.

Гедель доказал, что всякая достаточно богатая и эффективно аксиоматизированная формальная аксиоматическая теория необходимо неполна. Таким образом, невозможно эффективно описать в виде формальной аксиоматической теории даже такую сравнительно узкую область математики, как теория натуральных чисел. Более того, при попытке такого описания найдутся суждения, независимые от построенной теории, т. е. такие, что без противоречия можно присоединить как суждение, так и его отрицание. По теореме о существовании модели (§ 4) тогда и теория с присоединенным суждением, и теория с отрицанием суждения обе имеют модели, что вновь можно рассматривать как довод в пользу отсутствия некоторой «настоящей, единственно правильной» теории.

Наконец, классический способ рассуждения часто ведет к тому, что доказывается существование объектов и в то же время не указывается никакого способа построения этих объектов, даже если речь идет об объектах простой природы, которые в принципе можно было бы эффективно задавать. Например, доказывается, что всякая непрерывная на отрезке функция достигает максимума в некоторой точке, однако предлагаемое доказательство не дает никакого способа отыскания этой точки. В пользу этой теоремы часто приводят два следующих довода: во-первых, эта теорема кажется геометрически очевидной, во-вторых, никому не удалось пока придумать пример непрерывной функции, не имеющей на отрезке максимума, и тем самым опровергнуть теорему. На первый довод можно ответить, что он не имеет отношения к делу. Факт может быть геометрически и очевиден, но мы в этой теореме имеем дело не с интуитивной геометрией, а с некоторой точной теоретико-множественной моделью непрерывности, и вопрос состоит в том, достаточно ли соответствует эта модель интуиции. Приводимое довольно хитроумное теоретико-множественное доказательство как раз и призвано убедить нас, что модель выбрана удачно. Но мы оспариваем сейчас именно это доказательство, а не интуитивную очевидность факта. Что касается второго довода, то мы и без этой теоремы знаем, что некоторые непрерывные функции имеют максимум. Весь смысл теоремы состоит в том, что для всякой непрерывной функции найдется точка, в которой достигается максимум. Но что значит слово «найдется» в этой теореме и в чем ценность такого доказательства существования объекта, когда не дается никакого способа его построения?

2. Мы видим, что как объект исследования, так и способ рассуждения традиционной математики, и прежде всего теоретико-множественной математики, вызывают серьезную критику.

Неограниченное использование теоретико-множественных концепций ведет к парадоксам.

Неясно, в какой мере объект исследования в математике адекватно соответствует реальности.

Способ рассуждения в математике приводит к неэффективным доказательствам существования даже в случае простых объектов исследования.

Причины парадоксов можно видеть в способах образования понятий, например в использовании неограниченной аксиомы свертывания.

Чтобы избежать парадоксов, достаточно пользоваться ограниченной теорией множеств, например в рамках системы Цермело—Френкеля. Труднее справиться с остальными возражениями. Кроме того, остается неясным, будет ли система Цермело—Френкеля непротиворечивой. Пока у нас имеется лишь прагматическое наблюдение, что известные парадоксы обычным образом не выводятся в этой системе.

Неэффективность в математике связана с тем обстоятельством, что способы рассуждения, относящиеся к конечным множествам, были экстраполированы на бесконечные совокупности. Действительно, пусть, например, $P(x)$ некоторый предикат, где переменная x рассматривается как пробегающая натуральные числа. Утверждение $\exists xP(x) \vee \neg \exists xP(x)$ рассматривается как безусловно верное (это частный случай закона исключенного третьего), но как узнать, что именно верно $\exists xP(x)$ или $\neg \exists xP(x)$? Мы можем последовательно перебирать натуральные числа 0, 1, 2, ... и убеждаться последовательно, что $\neg P(0)$, $\neg P(1)$, ..., но ввиду бесконечности множества натуральных чисел таким способом невозможно убедиться $\neg \exists xP(x)$. Надежда может состоять лишь в удачном отыскании натурального n , для которого $P(n)$.

Можно отметить также, что современные физические представления не дают оснований считать, что в природе имеются актуально существующие бесконечные множества. Математика же довольно часто оперирует с бесконечными множествами, такими, как ω , $P(\omega)$ и т. д., как с некоторыми законченными и данными объектами исследования. В этом состоит применение абстракции актуальной бесконечности в математике. Многие математики, начиная с Гаусса, возражали против применения этой абстракции, тем не менее актуально бесконечные множества широко и существенным образом используются в современной математике.

Более осторожным является применение в математике абстракции потенциальной осуществимости, когда мы признаем лишь возможность неограниченного продолжения построений, отвлекаясь от технических, временных трудностей, но не счита-

ем, что существует множество всех результатов этого построения. Такого рода абстракции вполне достаточно, например, для построения большей части теории натуральных чисел.

3. Радикальный подход к решению обсужденных выше трудностей был предложен Гильбертом в серии работ 1926—1928 гг.

Он предложил разделить суждения классической математики на *действительные* (или *реальные*) и *идеальные*. Только действительные предложения рассматриваются как имеющие содержательный смысл. Действительные предложения должны относиться только к простым конструктивным объектам, не должны использовать актуальной бесконечности. Понимание, в каких случаях действительное предложение истинно, не должно вызывать возражений с точки зрения предыдущей критики. Идеальные же предложения могут быть сколь угодно сложными, они присоединяются к действительным только с целью систематизации теории, для облегчения выводов действительных предложений. Сами по себе идеальные предложения могут и не иметь содержательного смысла.

Рассматриваемая математическая теория позволяет вывести как идеальные, так и реальные суждения. Важно лишь, чтобы для теории выполнялся следующий *принцип корректности*: всякий раз, когда в рассматриваемой теории выводится действительное суждение, оно оказывается истинным с точки зрения содержательного смысла.

Конкретизируем эту идею следующим образом. Рассмотрим формальную аксиоматическую теорию ZF. Это очень богатая выразительными возможностями теория, имеющая дело с очень сложными объектами исследования, в том числе и с актуально бесконечными множествами.

Как нам хорошо известно, в теории ZF интерпретируется теория Ag — формальная арифметика. Фиксируем в языке Ag класс S предложений вида

$$\forall x_1 \dots x_n A(x_1, \dots, x_n),$$

где *A* — разрешимый предикат. Например, *A* может иметь вид равенства ($t=r$) или более сложный вид, но допускающий эффективную проверку для конкретных натуральных чисел. Так, хорошо известно, что в языке Ag можно построить формулу $A(n, x, y, z)$ с четырьмя параметрами, естественно выражающую предикат $x^{n+3} + y^{n+3} \neq z^{n+3}$. Тогда великая теорема Ферма

$$\forall xyzn (x^{n+3} + y^{n+3} \neq z^{n+3})$$

может быть элементом S.

Каждому предложению $B \in S$ соответствует формула B^* языка ZF, полученная интерпретацией *B* в языке ZF. Формулы вида B^* для $B \in S$ и назовем реальными предложениями ZF.

Реальным предложениям можно приписать содержательный смысл естественным образом. Если $B = \forall x_1 \dots x_n A(x_1, \dots, x_n)$, то *B* содержательно истинно, если для всех натуральных чисел

m_1, \dots, m_n результат проверки оцененной формулы $A(m_1, \dots, m_n)$ на истинность всегда дает истину. Указанное определение содержательной истинности реальных предложений полностью формулируется в терминах натуральных чисел, не требует привлечения актуально бесконечных множеств и использует лишь абстракцию потенциальной осуществимости: требуется, чтобы для всякого осуществимого набора m_1, \dots, m_n можно было довести процесс вычисления значения $A(m_1, \dots, m_n)$ до конца. Множество всех натуральных чисел как актуально завершенная совокупность при этом не используется. Элементарность этого определения подтверждается и тем, что оно легко формализуется в теории Аг. Таким образом, логические средства, используемые для определения содержательной истинности реальных суждений, не вызывают, по-видимому, предыдущей критики.

Отметим теперь важный факт. *Если теория ZF непротиворечива, то она удовлетворяет принципу корректности, т. е. всякое выводимое в ZF реальное предложение содержательно истинно.*

В самом деле, пусть ZF — непротиворечивая теория. Возьмем произвольный набор чисел m_1, \dots, m_n и вычислим истинностное значение $A(m_1, \dots, m_n)$. Мы утверждаем, что это окажется истина. В самом деле, иначе разрешимый предикат $\neg A$ на наборе m_1, \dots, m_n принимает значение «истина». Путем громоздкого, но в принципе несложного рассуждения (привлекающего машину Тьюринга, участвующую в определении разрешимого предиката A) можно показать, что тогда в Аг выводится $\neg A(\tilde{m}_1, \dots, \tilde{m}_n)$ и, значит, $\exists x_1 \dots x_n \neg A(x_1, \dots, x_n)$ (здесь \tilde{m}_i есть $S \dots S 0 m_i$ раз) и затем $\neg \forall x_1 \dots x_n A(x_1, \dots, x_n)$. Используя интерпретацию в ZF, получим вывод $\neg (\forall x_1 \dots x_n A(x_1, \dots, x_n))^*$. Если $(\forall x_1 \dots x_n A(x_1, \dots, x_n))^*$ было бы выводимо в ZF, то ZF оказалось бы противоречивой теорией.

Подобное рассуждение применимо и ко многим другим теориям. Таким образом, особое значение приобретает непротиворечивость рассматриваемой теории.

Отметим еще, что внимательный анализ показывает, что утверждение о непротиворечивости ZF (и многих других теорий) может быть само записано как *реальное* утверждение. Поэтому, даже если мы не в состоянии доказать непротиворечивость теории, мы можем *понимать* это утверждение в некотором содержательном смысле и, следовательно, искать содержательные основания для того, чтобы доверять ему или не доверять.

4. Каким же образом можно доказывать непротиворечивость теорий?

Классическим методом здесь является *метод интерпретаций*, с помощью которого вопрос о непротиворечивости одной теории сводится к такому же вопросу относительно другой теории. Фактически метод этот возник задолго до развития точных способов описания теорий методами математической логики. Ана-

литическая геометрия Декарта (1619) может рассматриваться как интерпретация геометрии средствами анализа. Кэли и Клейн (1871) предложили известную интерпретацию геометрии Лобачевского в геометрии Евклида.

С современной точки зрения метод интерпретаций состоит в построении относительной интерпретации одной формальной аксиоматической теории в другой. При этом, как мы уже отмечали, из непротиворечивости второй теории вытекает непротиворечивость первой.

Но как доказывать непротиворечивость мощных теорий, таких, как ZF и арифметика высокого порядка, для которых уже трудно найти надежную математическую теорию для их обоснования?

Гильберт предложил метод доказательства непротиворечивости, не требующий непосредственно применения интерпретаций.

Можно заметить, что хотя ZF описывает и сложную теорию, *сама формулировка* этой формальной аксиоматической теории требует очень элементарных средств. Кроме того, само понятие непротиворечивости формальной аксиоматической теории ZF очень элементарно и не требует упоминания об актуально бесконечных множествах: непротиворечивость ZF означает просто, что всякая строчка символов языка ZF не является выводом формулы вида $C \wedge \neg C$. Но если формулировка проблемы столь элементарна, то можно надеяться и доказать ее столь же элементарными средствами, не вникая в содержание теории, а изучая лишь формальную структуру выводов теории.

Суть программы Гильберта обоснования математики состоит в том, что сначала изучаемую теорию следует формализовать, а затем уже установить ее непротиворечивость логическими средствами, не вызывающими сомнений, *финитными* методами.

Например, непротиворечивость ZF можно выразить в языке \mathcal{A}_γ , и было бы заманчиво вывести это утверждение в формальной теории \mathcal{A}_γ . Тогда вопрос о непротиворечивости ZF свелся бы к вопросу о приемлемости гораздо более простой теории \mathcal{A}_γ .

К сожалению, эта программа не осуществима в полной мере. Как показал Гедель в 1932 г., непротиворечивость всякой достаточно богатой и эффективно аксиоматизированной теории не может быть доказана средствами самой этой теории.

Отсюда следует, в частности, что непротиворечивость ZF нельзя установить даже в ZF и подавно нельзя установить в \mathcal{A}_γ (при условии, конечно, что эти теории непротиворечивы).

Для доказательства непротиворечивости теорий следует искать логические средства, достаточно убедительные, но не формализуемые в рамках исходной теории. Такие логические средства должны иметь интуитивный смысл с некоторой содержательной точки зрения. Необходимые логические средства были разработаны в рамках теории неклассических логик, и, прежде

всего, *математического интуиционизма* [16]. В настоящее время непротиворечивость теорий Ag или $Ag2$ можно считать надежно установленной. Непротиворечивость такой теории, как ZF , гораздо более проблематична.

Независимо от проблемы установления непротиворечивости метод формализации математических теорий, предложенный Гильбертом, является центральным методом в современной теории доказательств.

ЛИТЕРАТУРА

- 1 Колмогоров А. Н., Драгалин А. Г. Введение в математическую логику. — М.: Изд-во Моск. ун-та, 1982.
- 2 Клини С. К. Математическая логика. — М.: Мир, 1973.
- 3 Клини С. К. Введение в метаматематику. — М.: ИЛ, 1957.
- 4 Мендельсон Э. Введение в математическую логику. — М.: Наука, 1976.
- 5 Шенфилд Дж. Математическая логика. — М.: Наука, 1975.
- 6 Гудстейн Р. Л. Математическая логика. — М.: ИЛ, 1961.
- 7 Гильберт Д., Бернайс П. Основания математики, т. 1, 2. — М.: Наука, 1979, 1982.
- 8 Новиков П. С. Элементы математической логики. — М.: Физматгиз, 1959.
- 9 Ершов Ю. Л., Палютин Е. А. Математическая логика. — М.: Наука, 1979.
- 10 Бурбаки Н. Теория множеств. — М.: Мир, 1965.
- 11 Френкель А., Бар-Хиллел И. Основания теории множеств. — М.: Мир, 1966.
- 12 Коэн П. Дж. Теория множеств и континуум-гипотеза. — М.: Мир, 1969.
- 13 Пех Т. Теория множеств и метод форсинга. — М.: Мир, 1973.
- 14 Такеутти Г. Теория доказательств. — М.: Мир, 1978.
- 15 Математическая теория логического вывода. Сборник переводов. — М.: Наука, 1967.
- 16 Драгалин А. Г. Математический интуиционизм. Введение в теорию доказательств. — М.: Наука, 1979.
- 17 Мальцев А. И. Алгоритмы и рекурсивные функции. — М.: Наука, 1965.
- 18 Роджерс Х. Теория рекурсивных функций и эффективная вычислимость. — М.: Мир, 1972.
- 19 Лавров И. А., Максимова Л. Л. Задачи по теории множеств, математической логике и теории алгоритмов. — М.: Наука, 1975.

**Андрей Николаевич Колмогоров,
Альберт Григорьевич Драгалин**

МАТЕМАТИЧЕСКАЯ ЛОГИКА

Дополнительные главы

Зав. редакцией *С. И. Зеленский*
Редактор *А. А. Локшин*
Мл. редактор *О. Е. Силантьева*
Художественный редактор *Л. В. Мухина*
Технический редактор *К. С. Чистякова*
Корректоры *В. П. Кададинская, С. Ф. Будаева*

Тематический план 1984 г. № 96
ИБ № 1920

Сдано в набор 22.07.83
Подписано к печати 20.12.83
Л-96363 Формат 60×90^{1/8}
Бумага тип. № 3
Гарнитура литературная. Высокая печать
Усл. печ. л. 7,5 Уч.-изд. л. 7,6
Тираж 13 150 экз. Заказ 170
Цена 25 коп. Изд. № 2799

Ордена «Знак Почета» издательство
Московского университета
103009, Москва, ул. Герцена, 5/7.
Типография ордена «Знак Почета»
изд-ва МГУ.
Москва, Ленинские горы

М

М



Σ

А. Н. Колмогоров, А. Г. Драгалин
**МАТЕМАТИЧЕСКАЯ
ЛОГИКА**

ДОПОЛНИТЕЛЬНЫЕ ГЛАВЫ

ИЗДАТЕЛЬСТВО
МОСКОВСКОГО
УНИВЕРСИТЕТА

