



*Томский межвузовский центр
дистанционного образования*

В.М. Зюзьков, А.А. Шелупанов

МАТЕМАТИЧЕСКАЯ ЛОГИКА И ТЕОРИЯ АЛГОРИТМОВ

Учебное пособие



ТОМСК – 2001

Министерство образования Российской Федерации

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)**

В.М. Зюзьков, А.А. Шелупанов

МАТЕМАТИЧЕСКАЯ ЛОГИКА И ТЕОРИЯ АЛГОРИТМОВ

Учебное пособие

2001

Рецензент: заведующий кафедрой прикладной математики Томского политехнического университета доктор технических наук, профессор В.А. Кочегуров, профессор Томского Государственного университета П.М. Нагорский.

В.М. Зюзьков, А.А. Шелупанов

Математическая логика и теория алгоритмов: Учебное пособие. – Томск: Томский межвузовский центр дистанционного образования, 2001. – 154 с.

Допущено Министерством образования РФ в качестве учебного пособия для студентов высших учебных заведений, обучающихся по специальностям "Комплексное обеспечение информационной безопасности автоматизированных систем", "Организация и технология защиты информации".

Рекомендовано Учебно-методическим объединением по образованию в области информационной безопасности для межвузовского использования в качестве учебного пособия по специальности "Комплексное обеспечение информационной безопасности автоматизированных систем".

© В.М. Зюзьков, А.А. Шелупанов, 2001
© Томский межвузовский центр
дистанционного образования, 2001

СОДЕРЖАНИЕ

Авторское предисловие	3
Глава 1. ОСНОВЫ ТЕОРИИ МНОЖЕСТВ	5
§1 Начальные понятия теории множеств	5
§2 Операции над множествами. Диаграммы Венна	8
§3 Отношения	10
§4 Функции	13
§5 Эквивалентность	16
§6 Порядок	18
Глава 2. ЛОГИКА ВЫСКАЗЫВАНИЙ	20
§1 Зачем мы изучаем математическую логику?	20
§2 Высказывания	22
§3. Логические связи	23
§4 Формулы логики высказываний	26
§5 Равносильность формул	28
§6 Тавтологично-истинные формулы	31
§7 Нормальные формы формул	32
§8 Разрешимость для логики высказываний	36
Глава 3. БУЛЕВЫ АЛГЕБРЫ	38
§1 Абстрактное определение булевой алгебры	38
§2 Булевы функции. Теорема о нормальной булевой форме	41
§3 Полные системы булевых функций	44
§4 Переключаемые элементы	45
Глава 4. ЛОГИКА ПРЕДИКАТОВ	50
§1 Формулы логики предикатов	50
§2 Интерпретации	53
§3 Выполнимость и общезначимость	55
Глава 5. ИСЧИСЛЕНИЯ	58
§1 Формальные аксиоматические теории	58
§2 Исчисление высказываний	60
§3 Исчисление предикатов	67
§4 Логический вывод	69
§5 Метод резолюций	72
§6 Неполнота математики	78
Глава 6. ТЕОРИЯ АЛГОРИТМОВ	81
§1 Понятие алгоритма и неформальная вычислимость	81
§2 Частично-рекурсивные функции	83
§3 Лямбда-исчисление	86
§4 Машины Тьюринга	95
§5 Тезис Чёрча	98
§6 Некоторые алгоритмически неразрешимые проблемы	99

§7 Сложность алгоритмов	101
Глава 7. ЛОГИЧЕСКИЕ ПАРАДОКСЫ	110
Глава 8. МНОГОЗНАЧНЫЕ ЛОГИКИ	114
§1. Трехзначная система Я.Лукасевича	114
§2. Логика Гейтинга	115
§3. Трехзначная система Бочвара Д.А.	116
§4. К-значная логика Поста Е.П.	117
Литература	118
Методические указания по курсу "Математическая логика и теория алгоритмов"	120
Контрольные задания по курсу "Математическая логика и теория алгоритмов"	124
Контрольная работа №1	124
Контрольная работа №2	138

АВТОРСКОЕ ПРЕДИСЛОВИЕ

Дорогой читатель, перед Вами книжка, которую мы довольно долго писали. Объясняется это прежде всего тем, что материал изложенный в ней сначала тщательно изучался в Учебно-методическом объединении по образованию в области информационной безопасности (Институт криптографии, связи и информатики Академии ФСБ в г. Москве), а затем в Министерстве образования РФ. Наконец книга у Вас в руках и мы надеемся, что она не окажется слишком сложной для Вас. Знания, которые Вы получите из этой книжки Вам пригодятся в дальнейшем, в частности, для усвоения криптографических методов защиты информации, прикладных алгоритмов, программных курсов и некоторых других важных дисциплин.

Данное пособие предназначено для студентов технических специальностей, у которых на курс математической логики отводится довольно мало часов. Поэтому, для того чтобы рассмотреть достаточно много вопросов нам пришлось пойти на сравнительно упрощенное изложение. При изложении материала мы считали, что вводимые понятия важнее теорем, а формулировки теорем важнее их доказательств.

Для начинающих изучение математической логики и теории алгоритмов мы предлагаем наиболее легкие и естественные доказательства. Очень важным, на наш взгляд, является то обстоятельство, что им вовсе не требуется специальных знаний для изучения этого курса, кроме "крепких" школьных знаний.

В пособии предлагается ряд распространенных логических парадоксов, на примере которых мы показываем трудности, возникающие в логике. Порой за простотой понятий и доказательств лежат глубинные, основополагающие принципы логического мышления. Излишне напоминать, что как любая наука математическая логика и теория алгоритмов требуют определенного уровня абстракции и некоторой математической культуры.

Очевидно, что в изложенном материале отсутствуют ряд разделов математической логики и теории алгоритмов. Сделано это нами сознательно в угоду желанию дать точные понятия наиболее важных тем. Авторы пособия не претендуют на глубокие исследования в области математической логики и теории алгоритмов. Наша цель куда более скромна - дать базовые понятия и принципы, и побудить у обучаемых желание всерьёз заняться этой увлекательной наукой, поскольку изучение математической логики должно хотя бы немного стимулировать процессы мышления.

В учебное пособие включены разнообразные логические задачи, упражнения, которые позволяют закрепить теоретический материал.

Материал изложенный в пособии полностью соответствует государственному образовательному стандарту для специальностей "Комплексное обеспечение информационной безопасности автоматизированных систем",

"Организация и технология защиты информации", а так же может быть использован для студентов других специальностей.

Пособие рассчитано на широкий круг читателей и рекомендовано Министерством образования РФ и Учебно-методическим объединением по образованию в области информационной безопасности для студентов обучающихся по специальностям в области информационной безопасности.

Данное учебное пособие можно использовать и как учебно-методические указания, поскольку практически в каждом параграфе пособия имеются примеры и упражнения с подробными указаниями и решениями.

Замечания и предложения направлять по адресу:

634050, Томск, пр-т Ленина, 40

ТУСУР кафедра КИБЭВС

тел. (83822) 413426

e-mail: office@keva.tusur.ru

В.М. Зюзьков,
А.А. Шелупанов

Чтобы что-то узнать, нужно уже
что-то знать.

Станислав Лем

Всякое начало трудно, - эта истина
справедлива для каждой науки.

К. Маркс

Глава 1. ОСНОВЫ ТЕОРИИ МНОЖЕСТВ

Кто неправильно застегнул первую
пуговицу, уже не застегнётся как
следует.

Иоганн Вольфганг Гёте

§1 Начальные понятия теории множеств

Понятие множества является основным, неопределяемым понятием, поэтому мы можем его только пояснить, например, с помощью следующего псевдоопределения.

Определение: Под *множеством* S будем понимать любое собрание определенных и различимых между собою объектов, мыслимое как единое целое. Эти объекты называются *элементами* множества S .

В этом интуитивном определении, принадлежащем немецкому математику Георгу Кантору (1845-1918), существенным является то обстоятельство, что собрание предметов само рассматривается как один предмет, мыслится как единое целое. Что касается самих предметов, которые могут входить во множество, то относительно них существует значительная свобода. Это может быть множество студентов в аудитории, множество целых чисел, множество точек плоскости. Заметим, что канторовская формулировка позволяет рассматривать множества, элементы которых по той или иной причине нельзя точно указать (например, множество простых чисел, множество белых носорогов и т. п.). Не следует думать, что множество обязательно должно содержать в каком-то смысле однородные объекты. Можно объединить в одно множество и королей и капусту.



Георг Кантор

Символом \in обозначается *отношение принадлежности*. Запись $x \in S$ означает, что элемент x принадлежит множеству S . Если элемент x не принадлежит множеству S , то пишут $x \notin S$.

Г. Кантором сформулировано несколько интуитивных принципов, которые естественно считать выполняющимися для произвольных множеств.

Интуитивный принцип объемности

Определение. Множества A и B считаются равными, если они состоят из одних и тех же элементов.

Записывают $A=B$, если A и B равны, и $A \neq B$ - в противном случае.

Пример 1.1. Проиллюстрируем принцип объемности. Множество A всех положительных четных чисел равно множеству B положительных целых чисел, представимых в виде суммы двух положительных нечетных чисел. Действительно, если $x \in A$, то для некоторого целого положительного числа m $x = 2m$; тогда $x = (2m - 1) + 1$, т. е. $x \in B$. Если $x \in B$, то для некоторых целых положительных p и q $x = (2p - 1) + (2q - 1) = 2(p + q - 1)$, т. е. $x \in A$.

Множество, элементами которого являются объекты a_1, a_2, \dots, a_n и только они, обозначают $\{a_1, a_2, \dots, a_n\}$. При этом порядок, в котором элементы расположены при описании множества не имеет значения; не имеет значения также возможность неоднократного повторения одних и тех же элементов при описании множества.

Пример 1.2. В силу принципа объемности $\{2, 4, 6\} = \{4, 2, 6\} = \{2, 4, 4, 6\}$; $\{\{1, 2\}\} \neq \{1, 2\}$, так как единственным элементом множества $\{\{1, 2\}\}$ является множество $\{1, 2\}$, а множество $\{1, 2\}$ состоит из двух элементов: чисел 1 и 2.

При рассмотрении способов задания множеств возникает проблема их эффективного описания. Ее решение обычно основано на интуитивном понятии "форма от x ". Под *формой от x* будем понимать конечную последовательность, состоящую из слов и символа x , такую, что если каждое вхождение x в эту последовательность заменить одним и тем же именем некоторого предмета соответствующего рода, то в результате получится истинное или ложное предложение. Например формами от x являются следующие предложения; " 3 делит x ", " $x^2 + 2x + 1 > x$ ", " $x^2 = 4$ ", " x - родственник Иванова". Напротив, предложения "для всех x $x^2 - 4 = (x - 2)(x + 2)$ ", "существует такое x , что $x > 0$ " и " $(x+1)/(x-1)$ " не являются формами от x .

Обозначим форму от x через $P(x)$.

Интуитивный принцип абстракции

Определение. Любая форма $P(x)$ определяет некоторое множество A , а именно множество тех и только тех предметов a , для которых $P(a)$ - истинное предложение.

Для множества A , определяемого формой $P(x)$, принято обозначение $A = \{x | P(x)\}$.

Пример 1.3.

1. $\{x \mid x - \text{положительное число, меньше 9}\} = \{1, 2, 3, 4, 5, 6, 7, 8\}$.
2. $\{x \mid x - \text{четное число}\}$ - множество четных чисел.

Описанные выше понятия теории множеств с успехом могут быть использованы в началах анализа, алгебры, математической логики и т. д. Однако надо иметь в виду, что при более строгих рассмотрениях такое интуитивное восприятие может оказаться неудовлетворительным.

Парадокс Бертрانا Рассела (1872-1970). (О несовершенстве интуитивных представлений о множествах.) Можно указать такие множества, которые принадлежат самим себе как элементы, например, множество всех множеств, и такие множества, которые не являются элементами самих себя, например, множество $\{1, 2\}$, элементами которого являются числа 1 и 2. Рассмотрим теперь множество $A = \{X \mid X \notin X\}$. Тогда, если $A \notin A$, то, по определению, $A \in A$. С другой стороны, если $A \in A$, то A - одно из тех множеств X , которые не есть элементы самих себя, т. е. $A \notin A$. В любом случае A есть элемент A и A не есть элемент A .



Бертран Рассел

Другая, более популярная форма этого парадокса известна как *парадокс бородбрея*. Владелец парикмахерской в одном селе повесил следующее объявление: "Брею тех и только тех жителей села, кто не бреется сам". Спрашивается, кто бреет бородбрея?

Этот парадокс свидетельствует о том, что широко используемая теория множеств в ее интуитивном, "наивном" изложении является противоречивой. Формализация теории множеств, связанная, в частности, с устранением парадоксов, способствовала развитию не только методов теории множеств, но и такой науки, как математическая логика.

Через \subseteq обозначим *отношение включения* между множествами, т. е. $A \subseteq B$, если каждый элемент множества A есть элемент множества B . Тогда говорят, что A есть *подмножество* множества B . Если $A \subseteq B$ и $A \neq B$, то говорят, что A есть *собственное* подмножество B , и пишут $A \subset B$.

Пример 1.4. Множество четных чисел есть подмножество множества целых чисел; множество рациональных чисел есть подмножество множества действительных чисел; $\{1, 2\} \subseteq \{1, 2, 3, 4\}$.

Заметим, что: а) $X \subseteq X$; б) если $X \subseteq Y$, $Y \subseteq Z$, то $X \subseteq Z$; в) если $X \subseteq Y$ и $Y \subseteq X$, то $X = Y$.

Не надо смешивать отношения принадлежности и включения. Хотя $1 \in \{1\}$, $\{\{1\}\} \subseteq \{\{1\}\}$, не верно, что $1 \in \{\{1\}\}$, так как единственным элементом множества $\{\{1\}\}$ является $\{1\}$.

Множество, не содержащее элементов, называется *пустым* и обозначается \emptyset . Пустое множество есть подмножество любого множества.

Множество всех подмножеств A называется *множеством-степенью* и обозначается $P(A)$.

Пример 1.5. Если $A = \{1, 2, 3\}$, то $P(A) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, A\}$.

В дальнейшем неоднократно будем пользоваться утверждением, что если множество A состоит из n элементов, то множество $P(A)$ состоит из 2^n элементов.

§2 Операции над множествами. Диаграммы Венна

Рассмотрим методы получения новых множеств из уже существующих.

Определение. *Объединением* множеств A и B называется множество $A \cup B$, все элементы которого являются элементами множества A или B :

$$A \cup B = \{x \mid x \in A \text{ или } x \in B\}.$$

Определение. *Пересечением* множеств A и B называется множество $A \cap B$, элементы которого являются элементами обоих множеств A и B :

$$A \cap B = \{x \mid x \in A \text{ и } x \in B\}.$$

Очевидно, что выполняются включения $A \cup B \subseteq A \cup B$ и $A \cap B \subseteq A \cap B$.

Определение. *Относительным дополнением* множества A до множества X называется множество $X \setminus A$ всех тех элементов множества X , которые не принадлежат множеству A :

$$X \setminus A = \{x \mid x \in X \text{ и } x \notin A\}.$$

Определение. *Симметрической разностью* множеств A и B называется множество $A \oplus B = (A \setminus B) \cup (B \setminus A)$.

Определение. Если все рассматриваемые в ходе данного рассуждения множества являются подмножествами некоторого множества U , то это множество U называется *универсальным* для данного рассуждения (контекста).

Определение. *Абсолютным дополнением* множества A называется множество \bar{A} всех тех элементов x , которые не принадлежат множеству A :

$$\bar{A} = U \setminus A.$$

$$\text{Заметим, что } X \setminus A = X \cap \bar{A}.$$

Для наглядного представления отношений между подмножествами какого-либо универсального множества используются диаграммы Венна. В этом случае множества обозначают областями на плоскости и внутри этих областей условно располагают элементы множества. Часто все множе-



Джон Венн

ства на диаграмме размещают внутри квадрата, который представляет собой универсальное множество U . Если элемент принадлежит более чем одному множеству, то на диаграмме, области, отвечающие таким множествам, должны перекрываться, чтобы общий элемент мог одновременно находиться в соответствующих областях (рис.1).

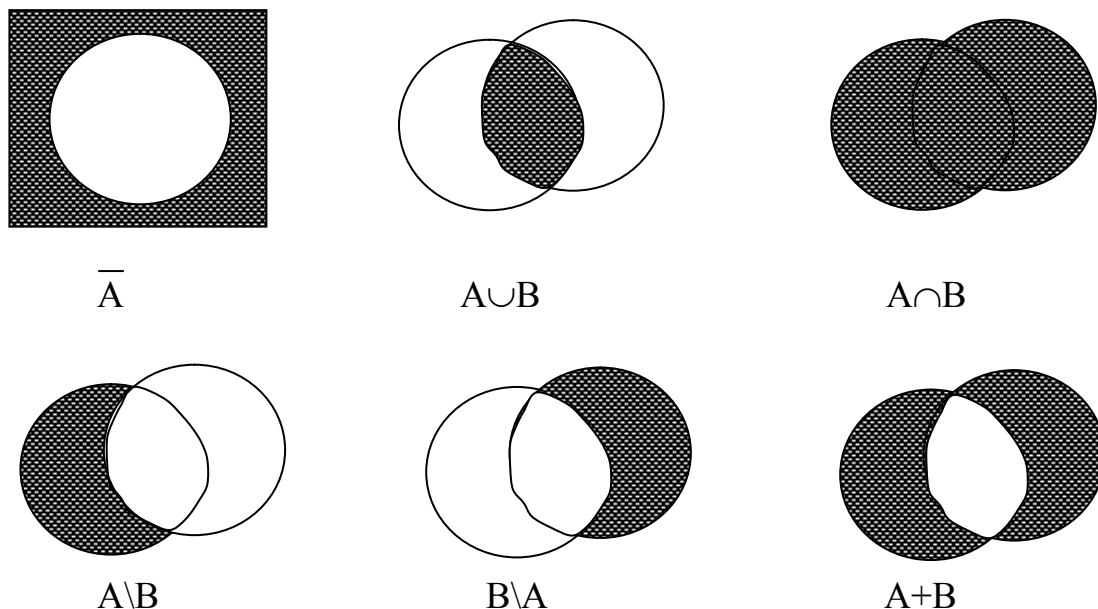


Рис. 1. Диаграммы Венна

Теорема 1.1. Для любых подмножеств A , B и C универсального множества U выполняются следующие тождества (основные тождества алгебры множеств):

- | | |
|---|--|
| 1. $A \cap B = B \cap A$ (коммутативность \cap)
2. $A \cap (B \cap C) = (A \cap B) \cap C$ (ассоциативность \cap)
3. $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$ (дистрибутивность \cap относительно \cup)
4. $A \cap \emptyset = \emptyset$
5. $A \cap \bar{A} = \emptyset$
6. $A \cap A = A$ (закон идемпотентности)
7. $A \cap U = A$
8. $A \cap B = \overline{A \cup B}$ (закон де Моргана)
9. $A \cap (A \cup B) = A$ (закон поглощения) | 1'. $A \cup B = B \cup A$ (коммутативность \cup)
2'. $A \cup (B \cup C) = (A \cup B) \cup C$ (ассоциативность \cup)
3'. $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ (дистрибутивность \cup относительно \cap)
4'. $A \cup \emptyset = A$
5'. $A \cup \bar{A} = U$
6'. $A \cup A = A$ (закон идемпотентности)
7'. $A \cup \emptyset = \emptyset$
8'. $A \cup B = \overline{\bar{A} \cap \bar{B}}$ (закон де Моргана)
9'. $A \cup (A \cap B) = A$ (закон поглощения) |
|---|--|

Докажем тождество 3. Сначала покажем, что $A \cap (B \cup C) \subseteq (A \cap B) \cup (A \cap C)$. Действительно, если $x \in A \cap (B \cup C)$, то $x \in A$ или $x \in B \cup C$. Если $x \in A$, то $x \in A$

$\cap B$ и $x \in A \cap C$. Следовательно, $x \in (A \cap B) \cup (A \cap C)$. Если $x \in B \cup C$, то $x \in B$ и $x \in C$. Отсюда $x \in A \cap B$ и $x \in A \cap C$, а значит $x \in (A \cap B) \cup (A \cap C)$. Теперь покажем, что $(A \cap B) \cup (A \cap C) \subseteq A \cap (B \cup C)$. Если $x \in (A \cap B) \cup (A \cap C)$, то $x \in A \cap B$ и $x \in A \cap C$. Следовательно, $x \in A$ или $x \in B$ и $x \in C$, т. е. $x \in B \cup C$. Отсюда $x \in A \cap (B \cup C)$.

Докажем тождество 8. Пусть $x \in \overline{A \cap B}$. Тогда $x \in U$ и $x \notin A \cap B$. Следовательно, $x \notin A$ и $x \notin B$. Отсюда $x \in \overline{A}$ и $x \in \overline{B}$, а значит, $x \in \overline{A \cup B}$. Итак, $A \cap B \subseteq \overline{\overline{A \cup B}}$. Пусть теперь, $x \in A \cup B$. Тогда $x \in A$ и $x \in B$. Следовательно, $x \in U$ и $x \notin A$ и $x \notin B$. Значит, $x \notin A \cap B$, т. е. $x \in \overline{A \cap B}$. Итак, $A \cup B \subseteq \overline{\overline{A \cap B}}$.

Остальные тождества доказываются аналогично. Рекомендуются сделать это самостоятельно. Справедливость этих тождеств можно наглядно проиллюстрировать с помощью диаграмм Венна, но это, конечно, не является доказательством. С другой стороны, диаграмму вполне можно использовать, чтобы на частном примере опровергнуть какое-нибудь общее утверждение.

Теорема 1.2. Предложения о произвольных множествах A и B попарно эквивалентны:

1) $A \subseteq B$; 2) $A \cup B = A$; 3) $A \cap B = B$.

Докажем, что из первого предложения следует второе. Действительно, так как $A \cup B \subseteq A$, то достаточно показать, что в этом случае $A \subseteq A \cup B$. Но если $x \in A$, то $x \in B$, так как $A \subseteq B$, и, следовательно, $x \in A \cup B$.

Докажем, что из второго предложения следует третье. Так как $A \cup B = A$, то $A \cap B = (A \cup B) \cap B$. По закону поглощения (см. тождество 9) $B \cap (A \cup B) = B$. Отсюда, используя закон коммутативности, получаем $A \cap B = B$.

Докажем, что из третьего предложения следует первое. Так как $A \subseteq A \cap B$, а по условию третьего предложения $A \cap B = B$, то $A \subseteq B$.

§3 Отношения

Определение. Упорядоченная пара $\langle x, y \rangle$ интуитивно определяется как совокупность, состоящая из двух элементов x и y , расположенных в определенном порядке. Две пары $\langle x, y \rangle$ и $\langle u, v \rangle$ считаются равными тогда и только тогда, когда $x = u$ и $y = v$.

Замечание. Предыдущее определение апеллирует к таким неопределенным понятиям как "совокупность" и "расположенные в определенном порядке". Для наших целей это вполне достаточно. Но понятие "упорядоченная пара" можно определить точно, используя понятия "множество", "элемент" и отношение принадлежности.

Упорядоченная n -ка элементов x_1, x_2, \dots, x_n обозначается $\langle x_1, x_2, \dots, x_n \rangle$ и, по определению, есть $\langle \langle x_1, x_2, \dots, x_{n-1} \rangle, x_n \rangle$.

Математическое понятие "отношение", так же, как и понятие "множество", является очень широким и общим. Три специальных типа отношений являются чрезвычайно важными: 1) функции; 2) отношения эквивалентности; 3) отношения порядка.

Определение. *Бинарным (или двуместным) отношением ρ называется множество упорядоченных пар. Если ρ есть некоторое отношение и пара $\langle x, y \rangle$ принадлежит этому отношению, то наряду с записью $\langle x, y \rangle \in \rho$ употребляется запись $x \rho y$. Элементы x и y называются *координатами* или *компонентами* отношения ρ . n -арным отношением называется множество упорядоченных n -ок.*

Областью определения бинарного отношения ρ называется множество $D_\rho = \{x \mid \text{существует такое } y, \text{ что } x \rho y\}$.

Определение. *Областью значений бинарного отношения ρ называется множество $R_\rho = \{y \mid \text{существует такое } x, \text{ что } x \rho y\}$.*

Пример 1.6.

1. Множество $\{\langle 1, 2 \rangle, \langle 2, 4 \rangle, \langle 3, 3 \rangle, \langle 2, 1 \rangle\}$ - бинарное отношение.
2. Отношение равенства на множестве действительных чисел есть множество $\{\langle x, y \rangle \mid x \text{ и } y - \text{действительные числа и } x \text{ равно } y\}$. Для этого отношения существует специальное обозначение $=$. Область определения D_ρ совпадает с областью значений R_ρ и является множеством действительных чисел.
3. Отношение "меньше чем" на множестве целых чисел есть множество $\{\langle x, y \rangle \mid \text{для целых чисел } x \text{ и } y \text{ найдется положительное число } z \text{ такое, что } x + z = y\}$. Для этого отношения существует специальное обозначение $<$. Область определения D_ρ совпадает с областью значений R_ρ и является множеством целых чисел.

Определение. *Прямым произведением множеств X и Y называется множество всех упорядоченных пар $\langle x, y \rangle$ таких, что $x \in X$ и $y \in Y$. Обозначается прямое произведение множеств X и Y через $X \times Y$.*

Каждое отношение ρ есть подмножество прямого произведения некоторых множеств X и Y таких, что $D_\rho \subseteq X$ и $R_\rho \subseteq Y$. Если $X = Y$, то говорят, что ρ есть отношение на множестве X .

Определение. *Прямым произведением множеств X_1, X_2, \dots, X_n называется множество всех упорядоченных n -ок $\langle x_1, x_2, \dots, x_n \rangle$ таких, что $x_i \in X_i$, $i = 1, 2, \dots, n$. Обозначается прямое произведение множеств X_1, X_2, \dots, X_n через $X_1 \times X_2 \times \dots \times X_n$. Если $X_1 = X_2 = \dots = X_n = X$, то пишут $X_1 \times X_2 \times \dots \times X_n = X^n$. Любое n -местное отношение есть подмножество прямого произведения некоторых множеств X_1, X_2, \dots, X_n .*

Пример 1.7.

1. Пусть $X = \{1, 2, 3\}$, $Y = \{0, 1\}$. Тогда $X \times Y = \{\langle 1, 0 \rangle, \langle 1, 1 \rangle, \langle 2, 0 \rangle, \langle 2, 1 \rangle, \langle 3, 0 \rangle, \langle 3, 1 \rangle\}$;
 $Y \times X = \{\langle 0, 1 \rangle, \langle 0, 2 \rangle, \langle 0, 3 \rangle, \langle 1, 1 \rangle, \langle 1, 2 \rangle, \langle 1, 3 \rangle\}$.

Мы указали, кроме того, такие множества X и Y , что $X \times Y \neq Y \times X$.

2. Пусть X - множество точек отрезка $[0, 1]$, а Y - множество точек отрезка $[1, 2]$. Тогда $X \times Y$ - множество точек квадрата $[0, 1] \times [1, 2]$ с вершинами в точках $(0, 1)$, $(0, 2)$, $(1, 1)$ и $(1, 2)$.

3. Пусть $A = \{1, 2, 3, 4, 5\}$. Пусть отношение ρ задано на A : $x \rho y \Leftrightarrow x$ делитель y . (Символ \Leftrightarrow заменяет слова "тогда и только тогда, когда".)

Тогда $\rho = \{ \langle 1, 1 \rangle, \langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 1, 4 \rangle, \langle 1, 5 \rangle, \langle 2, 2 \rangle, \langle 2, 4 \rangle, \langle 2, 6 \rangle, \langle 3, 3 \rangle, \langle 3, 6 \rangle, \langle 4, 4 \rangle, \langle 5, 5 \rangle, \langle 6, 6 \rangle \}$. Очевидно, $\rho \subseteq A^2$.

Для бинарных отношений обычным образом определены теоретико-множественные операции объединения, пересечения и т. д.

Определение. Обратным отношением для ρ называется отношение

$$\rho^{-1} = \{ \langle x, y \rangle \mid \langle y, x \rangle \in \rho \}.$$

Определение. Композицией отношений ρ_1 и ρ_2 называется отношение $\rho_2 \circ \rho_1 = \{ \langle x, z \rangle \mid \text{существует } y \text{ такое, что } \langle x, y \rangle \in \rho_1 \text{ и } \langle y, z \rangle \in \rho_2 \}$.

Для любых бинарных отношений выполняются следующие свойства:

$$(\rho^{-1})^{-1} = \rho;$$

$$(\gamma \circ \varphi)^{-1} = \varphi^{-1} \circ \gamma^{-1}.$$

Первое свойство очевидно. Для доказательства второго свойства покажем, что множества, записанные в левой и правой частях равенства, состоят из одних и тех же элементов. Действительно, $\langle x, y \rangle \in (\gamma \circ \varphi)^{-1} \Leftrightarrow \langle y, x \rangle \in \gamma \circ \varphi \Leftrightarrow \text{существует } z \text{ такое, что } \langle y, z \rangle \in \varphi \text{ и } \langle z, x \rangle \in \gamma \Leftrightarrow \text{существует } z \text{ такое, что } \langle z, y \rangle \in \varphi^{-1} \text{ и } \langle x, z \rangle \in \gamma^{-1} \Leftrightarrow \langle x, y \rangle \in \varphi^{-1} \circ \gamma^{-1}$.

Пример 18.

Пусть ρ и φ - отношения на множестве людей A , определенные следующим образом:

$x \rho y$, если и только если x - мать y ;

$x \varphi y$, если и только если x - отец y .

Имеем $\langle x, y \rangle \in \varphi \circ \rho$, тогда и только тогда, когда x - бабушка по линии отца для y . И $\langle x, y \rangle \in \rho \circ \varphi$, тогда и только тогда, когда x - дедушка по линии матери для y .

Покажем, что композиция отношений является ассоциативной операцией. Пусть даны три отношения $\rho \subseteq A \times B$, $\varphi \subseteq B \times C$ и $\gamma \subseteq C \times D$. Докажем, что $(\gamma \circ \varphi) \circ \rho = \gamma \circ (\varphi \circ \rho)$. Действительно, $\langle a, d \rangle \in (\gamma \circ \varphi) \circ \rho \Leftrightarrow \langle a, b \rangle \in \rho$ и $\langle b, d \rangle \in \gamma \circ \varphi$ для некоторых $b \in B \Leftrightarrow \langle a, b \rangle \in \rho$ и $\langle b, c \rangle \in \varphi$ и $\langle c, d \rangle \in \gamma$ для некоторых $b \in B$ и $c \in C \Leftrightarrow \langle a, c \rangle \in \varphi \circ \rho$ и $\langle c, d \rangle \in \gamma$ для некоторых $c \in C \Leftrightarrow \langle a, d \rangle \in \gamma \circ (\varphi \circ \rho)$.

Определение.

- Отношение ρ на множестве X называется *рефлексивным*, если для любого элемента $x \in X$ выполняется $x \rho x$.

- Отношение ρ на множестве X называется *симметричным*, если для любых $x, y \in X$ из $x \rho y$ следует $y \rho x$.

- Отношение ρ на множестве X называется *транзитивным*, если для любых $x, y, z \in X$ из $x\rho y$ и $y\rho z$ следует $x\rho z$.
- Отношение ρ на множестве X называется *антисимметричным*, если для любых $x, y \in X$ из $x\rho y$ и $y\rho x$ следует $x = y$.

Пример 1.9.

1. Пусть отношение ρ задано на множестве действительных чисел \mathbf{R} и $x\rho y$, если и только если $x \leq y$. Тогда ρ рефлексивно, потому что $x \leq x$ для всех $x \in \mathbf{R}$. Отношение ρ не симметрично, например, $1 \leq 2$, но $2 \leq 1$ не выполнено. Отношение ρ очевидно является транзитивным, ибо если $x \leq y$ и $y \leq z$, то $x \leq z$. Отношение является антисимметричным, поскольку $x \leq y$ и $y \leq x$ влечет $x = y$.

2. Пусть $\rho_1 = \{ \langle 1, 2 \rangle, \langle 2, 3 \rangle \}$, $\rho_2 = \{ \langle 1, 2 \rangle, \langle 1, 3 \rangle \}$. Тогда отношение ρ_1 не транзитивно, так как $\langle 1, 3 \rangle \notin \rho_1$. Но отношение ρ_2 является транзитивным, поскольку нет вообще таких элементов x, y и z , чтобы выполнялось условие $x\rho y$ и $y\rho z$.

3. Пусть A - непустое множество и $\rho = \emptyset$ (пустое отношение на A). Тогда отношение ρ является симметричным, транзитивным, антисимметричным. Если же $A = \emptyset$, то ρ еще и рефлексивно.

§4 Функции

Определим понятие "функция", следуя Дирихле. По сути дела при таком определении мы отождествляем функцию с ее графиком. Это одно из возможных определений. Другое определение, когда функция, рассматривается как закон вычисления некоторого значения, рассматривается в главе 6.

Определение. Бинарное отношение f называется *функцией*, если из $\langle x, y \rangle \in f$ и $\langle x, z \rangle \in f$ следует $y = z$.

Поскольку функции являются бинарными отношениями, то к ним применим интуитивный принцип объемности, т. е. две функции f и g равны, если они состоят из одних и тех же элементов. Область определения функции обозначается D_f , а область её значений - R_f . Определяются они также как и для бинарных отношений. Часто приходится сталкиваться с трудностями при определении области значений функции. Поэтому, если $D_f = X$ и $R_f \subseteq Y$, то говорят, что функция f задана на множестве X со значениями во множестве Y и осуще-



Петер Дирихле

ствяет *отображение* множества X во множество Y (или устанавливает *соответствие* между множествами X и Y). Это отображение обозначается таким образом: $f: X \rightarrow Y$.

Если f - функция, то вместо $\langle x, y \rangle \in f$ пишут $y = f(x)$ и говорят, что y - значение, соответствующее аргументу x , или y - образ элемента x при отображении f . При этом x называют прообразом элемента y .

Пример 1.10. Отношение $\{\langle 1, 2 \rangle, \langle 2, 3 \rangle, \langle \square, 0 \rangle\}$ - функция: отношение $\{\langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 2, 4 \rangle\}$ не является функцией: отношение $\{\langle x, x^2 + 2x + 1 \rangle \mid x - \text{действительное число}\}$ - функция, которую обычно обозначают $y = x^2 + 2x + 1$.

Назовем f n -местной функцией из X в Y , если $f: X^n \rightarrow Y$. Тогда пишем $y = f(x_1, \dots, x_n)$ и говорим, что y - значение функции при значении аргументов x_1, \dots, x_n .

Пусть дана функция $f: X \rightarrow Y$. Подчеркнем еще раз три особенности нашего определения функции (рис. 2):

- несколько элементов из области определения $D_f = X$ могут иметь один и тот же образ в области значений ($f(c) = f(d) = f(e) = 1$);
- не все элементы из Y обязаны быть образом некоторых элементов X (нет элемента $x \in X$ такого, что $f(x) = 4$);
- для любого элемента из X , если существует образ, то он должен быть единственным (для функции недопустимо, чтобы одному элементу $x \in X$ соответствовало два разных значения $f(x)$).

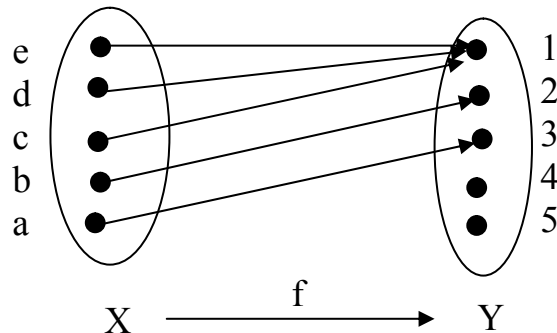


Рис. 2. $f: \{a, b, c, d, e\} \rightarrow \{1, 2, 3, 4, 5\}$

Определение. Пусть $f: X \rightarrow Y$.

- Функция (отображение) f называется *инъективной* (*инъективным*), если для любых x_1, x_2, y из $y = f(x_1)$ и $y = f(x_2)$ следует, что $x_1 = x_2$ (или, иначе, из $\langle x_1, y \rangle \in f$ и $\langle x_2, y \rangle \in f$ следует, что $x_1 = x_2$). Менее формально, функция f - инъективна, если для всех $x_1, x_2 \in X$ выполняется: $x_1 \neq x_2 \Rightarrow f(x_1) \neq f(x_2)$.

- Функция (отображение) f называется *сюрьективной* (*сюрьективным*), если для любого элемента $y \in Y$ существует элемент $x \in X$ такой, что $y = f(x)$.

- Функция (отображение) f называется *биективной* (*биективным*), если f одновременно инъективна и сюрьективна. Если существует биективная функция $f: X \rightarrow Y$, то говорят, что f осуществляет взаимно-однозначное соответствие между множествами X и Y .

Пример 1.11. Рассмотрим три функции, отображающие множество действительных чисел \mathbb{R} во множество действительных чисел $f_i: \mathbb{R} \rightarrow \mathbb{R}$, $i = 1, 2, 3, 4$:

- 1) функция $f_1(x) = e^x$ инъективна, но не сюрьективна;
- 2) функция $f_2(x) = x^3 - x$ сюрьективна, но не инъективна;
- 3) функция $f_3(x) = 2x + 1$ биективна;
- 4) функция $f_4(x) = x^2$ не является ни инъективной, ни сюрьективной.

Композиция двух функций f и g есть отношение $g \circ f = \{ \langle x, z \rangle \mid \text{существует } y \text{ такое, что } xfy \text{ и } ygz \}$.

Теорема 1.3. Композиция двух функций есть функция. При этом, если $f: X \rightarrow Y$, $g: Y \rightarrow Z$, то $g \circ f: X \rightarrow Z$.

Действительно, если $\langle x, y \rangle \in g \circ f$ и $\langle x, z \rangle \in g \circ f$, то существует такое u , что xfu , ugu , и существует такое v , что xfv , vgz . Поскольку f - функция, то $u = v$; поскольку g - функция, то $y = z$ и, следовательно, $g \circ f$ - функция. Вторая часть утверждения очевидна.

Если $f: X \rightarrow Y$, $g: Y \rightarrow Z$, то $g \circ f: X \rightarrow Z$. Тогда используется более привычная запись для композиций функций: $z = g(f(x))$.

Верно также и следующее утверждение.

Теорема 1.4. Композиция двух биективных функций есть биективная функция.

Определение. Тождественным отображением множества X в себя называется отображение $e_X: X \rightarrow X$ такое, что для любого $x \in X$ $e_X(x) = x$. Тогда, если $f: X \rightarrow Y$, то $e_X \circ f = f$, $f \circ e_X = f$.

Пусть F^{-1} - отношение, обратное F . Выясним, при каких условиях отношение F^{-1} будет функцией. Его называют тогда *обратной функцией* или, если F осуществляет отображение множества X во множество Y , *обратным отображением*.

Теорема 1.5. Отображение $F: X \rightarrow Y$ имеет обратное отображение $F^{-1}: Y \rightarrow X$ тогда и только тогда, когда F - биекция.

Если F - биекция, то, поскольку F сюрьективно, F^{-1} определено на множестве Y . Кроме того, F^{-1} - функция, так как если $\langle y, x_1 \rangle \in F^{-1}$ и $\langle y, x_2 \rangle \in F^{-1}$, то $\langle x_1, y \rangle \in F$ и $\langle x_2, y \rangle \in F$, а в силу инъективности F имеем $x_1 = x_2$.

Пусть теперь отображение F имеет обратное отображение F^{-1} , определенное на множестве Y со значениями во множестве X . Тогда F сюрьективно, поскольку любой элемент $y \in Y$ имеет прообраз $x \in X$. При этом F

инъективно, так как если $\langle x_1, y \rangle \in F$ и $\langle x_2, y \rangle \in F$, то $\langle y, x_1 \rangle \in F^{-1}$ и $\langle y, x_2 \rangle \in F^{-1}$, а поскольку F^{-1} - функция, то $x_1 = x_2$.

Заметим, что для того, чтобы обратное отношение F^{-1} было функцией, достаточно инъективности функции F . Поэтому функция $F(x) = x^2 : \mathbf{R} \rightarrow \mathbf{R}$ не будучи биекцией, не имеет обратной функции. Эта функция не имеет обратной, если даже она будет отображением на множество неотрицательных вещественных чисел.

Поскольку функция есть бинарное отношение, то выполняются следующие свойства инъективных функций F и G :

- 1) $(F^{-1})^{-1} = F$;
- 2) $(G \downarrow F)^{-1} = F^{-1} \downarrow G^{-1}$.

Если $F: X \rightarrow Y$ - биекция, то

- 1) $F^{-1} \downarrow F = e_X$;
- 2) $F \downarrow F^{-1} = e_Y$.

§5 Эквивалентность

Одним из самых важных типов отношений является отношение эквивалентности на множестве.

Определение. Рефлексивное, симметричное и транзитивное отношение ρ на множестве X называется *отношением эквивалентности* на множестве X .

Пример 1.12.

1. Отношение равенства на множестве целых чисел есть отношение эквивалентности.

2. Пусть $A = \mathbf{R}^2 \setminus \{ \langle 0, 0 \rangle \}$ - множество точек на плоскости за исключением начала координат. Отношение ρ на A определим так: $\langle a, b \rangle \rho \langle c, d \rangle$ тогда и только тогда, когда точки $\langle a, b \rangle$ и $\langle c, d \rangle$ лежат на одной прямой, проходящей через начало координат. Легко показать, что отношение ρ является отношением эквивалентности.

3. Отношение сравнимости по модулю натурального числа n на множестве целых чисел Z : $x \equiv y \pmod{n}$ тогда и только тогда, когда $x - y$ делится на n . Это отношение рефлексивно на Z , так как для любого $x \in Z$ $x - x = 0$, и, следовательно, делится на n . Это отношение симметрично, так как если $x - y$ делится на n , то $y - x$ делится на n . Это отношение транзитивно, так как если $x - y$ делится на n , то для некоторого целого t_1 имеем $x - y = t_1 n$, а если $y - z$ делится на n , то для некоторого целого t_2 имеем $y - z = t_2 n$. Отсюда $x - z = (t_1 + t_2)n$, т. е. $x - z$ делится на n .

4. Рассмотрим отношение ρ , определенное на множестве неотрицательных целых чисел так: $n \rho m$, если и только если n - делитель m . Отношение ρ не является отношением эквивалентности. Чтобы показать это, достаточно убедиться, что хотя бы одно из трех свойств не выполняется

для ρ . Очевидно, что ρ не является симметричным отношением, так как, например, 2 - делитель 4, но 4 не является делителем 2.

5. На множестве $N \times N$, где N - множество натуральных чисел, определим отношение $\rho : \langle x, y \rangle \rho \langle u, v \rangle \Leftrightarrow xv = uy$. Это отношение рефлексивно: $\langle x, y \rangle \rho \langle x, y \rangle$, так как $xy = yx$; симметрично: если $\langle x, y \rangle \rho \langle u, v \rangle$, то $\langle u, v \rangle \rho \langle x, y \rangle$, так как из $xv = uy$ следует, что и $uy = vx$; транзитивно: если $\langle x, y \rangle \rho \langle u, v \rangle$, $\langle u, v \rangle \rho \langle w, z \rangle$, то $\langle x, y \rangle \rho \langle w, z \rangle$, так как, перемножая левые и правые части равенств $xv = uy$ и $uz = vw$ после сокращения получаем $xz = uw$.

Пусть ρ - отношение эквивалентности на множестве X .

Определение. *Классом эквивалентности*, порожденным элементом x , называется подмножество множества X , состоящее из тех элементов $y \in X$, для которых $x \rho y$. Класс эквивалентности, порожденный элементом x , обозначается $[x]$:

$$[x] = \{y \mid y \in X \text{ и } x \rho y\}.$$

Пример 1.13.

1. Отношение равенства на множестве целых чисел порождает следующие классы эквивалентности: для любого элемента $x \in \mathbb{Z}$ $[x] = \{x\}$, т. е. каждый класс эквивалентности состоит только из одного элемента - числа x .

2. Отношение сравнимости по модулю числа n на множестве целых чисел \mathbb{Z} порождает следующие классы эквивалентности: вместе с любым числом $a \in \mathbb{Z}$ в этом же классе эквивалентности содержатся все числа вида $a + kn$, где k - целое. Очевидно, что все числа $0, 1, 2, \dots, n-1$ порождают различные классы эквивалентности, которые обозначим $[0], [1], [2], \dots, [n-1]$. Они называются классами вычетов по модулю n . Все остальные классы эквивалентности для этого отношения совпадают с ними, так как любое число $a \in \mathbb{Z}$ можно представить в виде $a = qn + r$, где $0 \leq r < n$.

3. Класс эквивалентности, порожденной парой $\langle x, y \rangle$ для отношения ρ из примера 1.12 (5) определяется соотношением $[\langle x, y \rangle] = \{\langle u, v \rangle \mid x/y = u/v\}$. Каждый класс эквивалентности в этом случае определяет одно положительное рациональное число.

Теорема 1.6. Пусть ρ - отношение эквивалентности на множестве X . Тогда: 1) если $x \in X$, то $x \in [x]$; 2) если $x, y \in X$ и $x \rho y$, то $[x] = [y]$ (т. е. класс эквивалентности порождается любым своим элементом).

Для доказательства первой части утверждения достаточно воспользоваться рефлексивностью отношения ρ : $x \rho x$ и, следовательно, $x \in [x]$. Докажем вторую часть утверждения. Пусть $z \in [y]$. Тогда $y \rho z$ и в силу транзитивности отношения ρ $x \rho z$, т. е. $z \in [x]$. Отсюда $[y] \subseteq [x]$. Аналогично, в силу симметричности ρ можно показать, что $[x] \subseteq [y]$, а значит $[y] = [x]$.

Определение. *Разбиением* множества X называется множество попарно непересекающихся подмножеств X таких, что каждый элемент множества X принадлежит одному и только одному из этих подмножеств.

Пример 1.14.

$X = \{1, 2, 3, 4, 5\}$. Тогда $\{\{1, 2\}, \{3, 5\}, \{4\}\}$ - разбиение множества X . Пусть X - множество студентов института. Тогда разбиением этого множества является, например, совокупность студенческих групп.

Теорема 1.7. Всякое разбиение множества X определяет на X отношение эквивалентности ρ : $x\rho y$ тогда и только тогда, когда x и y принадлежат одному подмножеству разбиения.

Рефлексивность и симметричность ρ очевидны. Пусть теперь $x\rho y$ и $y\rho z$. Тогда $x, y \in X_1, y, z \in X_2$, где X_1, X_2 - подмножества из разбиения X . Поскольку $y \in X_1, y \in X_2$, то $X_1 = X_2$. Следовательно, $x, z \in X_1$ и $x\rho z$.

Теорема 1.8. Всякое отношение эквивалентности ρ определяет разбиение множества X на классы эквивалентности относительно этого отношения.

Докажем, что совокупность классов эквивалентности определяет разбиение множества X . В силу теоремы 1.6 $x \in [x]$, а следовательно, каждый элемент множества X принадлежит некоторому классу эквивалентности. Из теоремы 1.6 вытекает также, что два класса эквивалентности либо не пересекаются, либо совпадают, так как если $z \in [x]$ и $z \in [y]$, то $x\rho z$, откуда $[x] = [z]$, и $y\rho z$, откуда $[y] = [z]$. Следовательно, $[x] = [y]$.

Определение. Совокупность классов эквивалентности элементов множества X по отношению эквивалентности ρ называется *фактор-множеством* множества X по отношению ρ и обозначается X/ρ .

§6 Порядок

Элементы многих множеств можно разместить в определенном порядке на основе некоторого, заранее оговоренного соглашения. Например, на любом подмножестве A множества целых положительных чисел можно договориться о таком расположении элементов, при котором меньшие элементы будут находиться левее больших. При этом можно сказать, что на множестве A определено отношение порядка $x \rho y$, где ρ есть отношение "меньше или равно" ($x \leq y$).

Определение. Рефлексивное, транзитивное и антисимметричное отношение называется отношением *частичного порядка* на множестве X и обозначается символом \sqsubseteq .

Пример 1. 15.

1. Отношение $x \leq y$ на множестве действительных чисел есть отношение частичного порядка.

2. Отношение $x < y$ на множестве действительных чисел не является отношением частичного порядка, поскольку не рефлексивно.

3. Во множестве подмножеств некоторого универсального множества U отношение $A \subseteq B$ есть отношение частичного порядка.

4. Схема организации подчинения в учреждении есть отношение частичного порядка на множестве должностей.

5. Отношение на множестве слов, определенное так: "слово w связано отношением ρ со словом v , если $w = v$ или w появляется в словаре перед словом v " является отношением частичного порядка (лексикографический порядок).

Определение. Отношение частичного порядка на множестве X , для которого любые два элемента сравнимы, т. е. для любых $x, y \in X$ $x \sqsubseteq y$ или $y \sqsubseteq x$, называется отношением *линейного порядка*.

Пример 1.15. В примере 1.14 отношение, определенное в пунктах 1 и 5 есть отношение линейного порядка, а отношение, определенное в п. 2, таковым не является.

Пусть на множестве X задано отношение частичного порядка ρ . Как можно задать отношение частичного порядка на множестве $X \times X$, т. е. как сравнивать пары элементов из множества X ? Один из возможных способов состоит в следующем: на множестве $X \times X$ определяем отношение Π условием $\langle a, b \rangle \Pi \langle c, d \rangle \Leftrightarrow a \rho c$ и $b \rho d$. Отношение Π есть отношение частичного порядка. Оно называется отношением Парето.

Определение. Множество X с заданным на нем частичным (линейным) порядком называется *частично (линейно) упорядоченным*.

Логика есть анатомия мышления.

Джон Локк

Глава 2. ЛОГИКА ВЫСКАЗЫВАНИЙ

Было бы крайне нелогично
руководствоваться в жизни только логикой.

Лешек Кумор

§1 Зачем мы изучаем математическую логику?

Логика есть наука о законах и формах познающего мышления. Логика изучает мышление, но не всякое мышление, а лишь те мыслительные процессы, которые направлены на обнаружение и обоснование истины, на решение некоторой задачи, на поиск путей преодоления тех или иных трудностей, встающих перед нами как в профессиональной деятельности, так и в обыденной жизни.

Логику интересует лишь форма наших мыслей, но не их содержание. Разнообразие содержания укладывается в сравнительно небольшое число форм. Грубо говоря, логику интересуют сосуды - бутылки, ведра, бочки, - а не то, что в них налито.

В этом отношении логика сходна с грамматикой, которую мы изучали в школе. Грамматика тоже исследует и описывает формы языковых выражений, отвлекаясь от их содержания. Известное стихотворение "Бармаглот" из "Алисы в Зазеркалье" Льюиса Кэрролла начинается со следующих строк:

"Варкалось. Хливкие шорьки
Пырлялись по наве.
И хрюкотали зелюки,
Как мюмзики в мове."

Знание грамматики позволяет нам обнаружить, что в этих строчках является подлежащим,



Льюис Кэрролл (он же - математик и логик Льюидж Доджсон)

сказуемым и т. д. Мы можем говорить о роде, числе, падеже наших существительных, не имея ни малейшего представления о том, что обозначают соответствующие слова. Более того, как говорит Алиса об этих строках: они "наводят на всякие мысли, хоть и неясно - на какие". Аналогичное знание о формах мысли дает нам логика.

При изучении логики мы вводим различные формальные языки. Дело в том, что формальные языки всегда проще, чем структура естественных языков. Иногда естественный язык может быть очень сложен.

Вот как, например, Марк Твен обыгрывает особенности словообразования в немецком языке [25, с. 59]:

"В одной немецкой газете, - уверяет он, - я сам читал такую весьма занятную историю:

Готтентоты (по-немецки: □хоттентотен□), как известно, ловят в пустынях кенгуру (по-немецки: □бейтельрате□ - сумчатая крыса). Они обычно сажают их в клетки (□коттэр□, снабженные решетчатыми крышками (□латтенгиттер□) для защиты от непогоды (□веттер□).

Благодаря замечательным правилам немецкой грамматики все это вместе - кенгуру и клетки - получают довольно удобное название:

□Латтенгиттерветтеркоттэрбейтельратте□.

Однажды в тех местах, в городе Шраттертроттэле, был схвачен негодяй, убивший готтентотку, мать двоих детей.

Такая женщина по-немецки должна быть названа □хоттентотенмуттер□, а ее убийца сейчас же получил в устах граждан им □щраттертроттэльхоттентотенмуттэрраттэнтэтэр□, ибо убийца - по-немецки □аттэнтэтэр□.

Преступника поймали и за неимением других помещений посадили в одну из клеток для кенгуру, о которых выше было сказано. Он бежал, но снова был изловлен. Счастливый своей удачей, негр-охотник быстро явился к старшине племени.

– Я поймал этого ... Бейтельратте? Кенгуру? - в волнении вскричал он.

– Кенгуру? Какого? - сердито спросил потревоженный начальник.

– Как какого? Этого самого! Латтенгиттерветтеркоттэрбейтельратте.

– Яснее! Таких у нас много... Непонятно, чему ты так радуешься?

– Ах ты, несчастье какое! - возмутился негр, положил на землю лук и стрелы, набрал в грудь воздуха и выпалил:

– Я поймал щраттертроттэльхоттентотенмуттэрраттэнтэтэр-латтенгиттерветтеркоттэрбейтельратте! Вот кого!

Тут начальник подскочил, точно подброшенный пружиной:

– Так что же ты мне сразу не сказал этого так коротко и ясно, как сейчас?!"

Математическая логика - это логика, развиваемая с помощью математических методов. Этот термин имеет и другой смысл: изучать математическую логику - значит изучать логику, используемую в математике.

Математическая логика, возникшая почти 100 лет назад в связи с внутренними потребностями математики, нашла применение в теоретическом и практическом программировании и, судя по всему, взаимодействие этих двух наук в недалеком будущем сможет принести новые плоды.

Почему программисты обратились к математической логике, а логики заинтересовались программированием? Математическая логика, занимается построением формальных языков, предназначенных для представлений таких фундаментальных понятий, как функция, отношение, аксиома, доказательство, и изучение основанных на этих языках логических и логико-математических исчислений.

В недрах математической логики были найдены математически точные понятия алгоритма и вычислимой функции, развита семантика формальных языков и теорий, построены системы логического вывода - и все это, заметим, было сделано в 30-40-х годах, в т. е. "докомпьютерную эру".

Программирование также имеет дело с формальными языками - языками программирования. Чтобы сделать эти языки удобными и естественными для человека полезно воспользоваться опытом математической логики. В результате появились принципиально новые языки функционального (Лисп, Сlean) и логического (Пролог) программирования.

Для формализации семантики программы (это полезно при разработке трансляторов) необходим аппарат математической логики (уже использовались : λ -исчисление Черча, теория областей Дана Скотта).

Другие приложения математической логики в программировании:

- теория логического вывода;
- правильность программ относительно спецификаций;
- "доказательное" программирование - метод построения правильных программ;
- задачи представления и обработки знаний;
- параллельные вычисления;
- проблемы сложности вычислений;
- элементная логическая база компьютеров.

Противоположность правильного высказывания - ложное высказывание. Но противоположность глубокой истины может быть другая глубокая истина.

Нильс Бор

§2 Высказывания

Мы начинаем изучать математическую логику со сравнительно ограниченного и нетрудного ее раздела, чтобы затем иметь возможность продвигаться вширь и вглубь. Этот раздел посвящен изучению связей между *высказываниями* - связей, определяемых исключительно тем, каким образом одни высказывания строятся из других ("элементарных"). Эта часть логики называется *логикой высказываний*.

Под *высказыванием* принято понимать языковое предложение, о котором имеет смысл говорить, что оно истинно или ложно.

Высказываниями являются, например, следующие предложения: "дважды два - четыре", "лошади едят овес и сено", "Волга впадает в Черное море". Первые два предложения истинны, а третье - ложно. Предложения "Москва - столица России", "Венера имеет спутник, сравнимый с Луной", "для всякого целого $n > 2$, уравнение в натуральных числах $x^n + y^n = z^n$ не имеет решения" являются высказываниями, причем первое из них истинно, второе - ложное, а истинность третьего высказывания ("большой теоремы Ферма") была установлена Эндрю Уайлсом только в 1995 г. Высказывания выражаются повествовательными предложениями, поэтому предложения "Шагом марш!" и "Который час?" высказываниями не являются. Предложения "Город стоит на берегу реки", "Мороз и солнце! День чудесный!" и " $x + y = 4$ " тоже не следует относить к высказываниям ввиду их недостаточной точности.

В логике высказываний интересуются не содержанием, а *истинностью* или *ложностью* высказываний (т. е. их *истинностным значением*). Истинностные значения - *истина* и *ложь* - будем обозначать соответственно И и Л соответственно. Множество {И, Л} называется *множеством истинностных значений*. В пределах логики высказываний внутренняя структура предложений, выражающих в каком-то смысле элементарные или атомарные высказывания нас вообще не будет интересовать. Нам надо лишь уметь распознавать и различать их. Мы будем обозначать их прописными буквами латинского алфавита: P, Q, R и т. п.

§3 Логические связи

Грамматическими средствами в разговорном языке из нескольких высказываний можно составить сложное (составное) высказывание. Например, с помощью союзов "и", "или" и отрицательной частицы "не" можно из простых высказываний "Москва - столица США" (ложного) и "Берлин - столица Германии" (истинного) составить следующие сложные высказывания: "Москва - не столица США", "Москва - столица США или Берлин - столица Германии", "Москва - столица США и Берлин - столица Германии". Первые два высказывания истинны, а последнее ложное.

В математической логике используются специальные операции (конструкции), позволяющие из исходных высказываний получать более сложные высказывания. Эти операции обозначаются символами: "&", " \vee ", " \supset ", " \sim ", " \neg ". Первые четыре операции - бинарные (двуместные), пятая - унарная (одноместная).

Эти логические операции (связки) над высказываниями таковы, что истинностные значения составных высказываний определяются только истинностными значениями составляющих высказываний, а не их смыслом.

Отрицанием высказывания P называется высказывание, истинное тогда и только тогда, когда высказывание P ложно. Отрицание P обозначается через $\neg P$ и читается как "не P ". Отрицание высказывания определяется также таблицей истинности (см. табл. 1).

В разговорной речи отрицание соответствует составлению из высказывания P нового высказывания, которое передается словами "неверно, что P " или "не P ".

Таблица 1

P	$\neg P$
И	Л
Л	И

Конъюнкцией двух высказываний P и Q называется высказывание, истинное тогда и только тогда, когда истинны оба высказывания. Конъюнкция высказываний P и Q обозначается через $P \& Q$ и читается как " P и Q ". Конъюнкция определяется также таблицей истинности (см. табл. 2).

В разговорной речи конъюнкция соответствует обычно соединению высказываний союзом "и". Кроме того, следующий список выражений в словесных рассуждениях часто может истолковываться как конъюнкция: "не только A , но и B ", "как A , так и B ", " A вместе с B ", " A , в то время как B ", " B , хотя и A ".

Таблица 2

P	Q	$P \& Q$
И	И	И
И	Л	Л
Л	И	Л
Л	Л	Л

Дизъюнкцией двух высказываний P и Q называется высказывание, ложное тогда и только тогда, когда оба высказывания ложны. Дизъюнкция высказываний P и Q обозначается через $P \vee Q$ и читается как " P или Q ". Дизъюнкция определяется также таблицей истинности (см. табл. 3).

В разговорной речи дизъюнкция соответствует соединению высказываний союзом "или" в "неразделительном смысле". Дизъюнкция передается также часто выражениями: "... или ... или оба", "и/или".

Таблица 3

P	Q	$P \vee Q$
И	И	И
И	Л	И
Л	И	И
Л	Л	Л

Из двух высказываний P и Q можно составить высказывание " P влечет Q " (или, иначе, "если P , то Q "). Не математик может признать утверждение типа "если $2 \times 2 = 5$, то Москва - столица России" ложным, поскольку для него истинность высказывания " P влечет Q " означает, что P по смыслу должно влечь за собой Q . Но тогда связка "влечет" зависит от смысла самих этих высказываний. Однако практика показывает, что можно обороты типа " P влечет Q " и "из P следует Q " использовать таким образом, чтобы под ними каждый раз подразумевалась некоторая операция, не зависящая от смысла высказываний. Рассмотрим следующие высказывания [24, с. 25]:

- 1) если $0 = 0$, то $1 = 1$; 2) если $0 = 1$, то $0 = 0$;
- 3) если $0 = 0$, то $0 = 1$; 4) если $0 = 1$, то $1 = 2$.

Первое утверждение естественно считать истинным, поскольку, используя равенство $0 = 0$, а также другие свойства чисел, можно вывести равенство $1 = 1$ (например, прибавляя 1 к обеим частям равенства $0 = 0$).

Второе утверждение также естественно считать истинным: умножая на 0 обе части равенства $0 = 1$, получаем равенство $0 = 0$.

Третье утверждение приходится считать ложным, ибо, исходя из верного равенства, мы с помощью умозаключений никогда не придем к ложному.

Четвертое рассуждение естественно считать истинным: прибавляя 1 к обеим частям равенства $0 = 1$, получаем равенство $1 = 2$.

Таким образом, используя оборот "если P , то Q " как логическую операцию (связку), определим её следующим образом.

Импликацией двух высказываний P и Q называется высказывание, ложное тогда и только тогда, когда P - истинно, а Q - ложно. Импликация высказываний P и Q обозначается через $P \supset Q$ (или $P \Rightarrow Q$) и читается как " P влечет Q " (или, иначе, "если P , то Q ", "из P следует Q ", " Q только, если P ", "коль скоро A , то B ", "в случае A имеет место B ", "для B достаточно A ", "для A необходимо B "). Высказывание P называется *посылкой* импликации, а высказывание Q - *заключением* импликации. Импликация определяется также таблицей истинности (см. табл. 4).

Таблица 4

P	Q	$P \supset Q$
И	И	И
И	Л	Л

Л	И	И
Л	Л	И

Эквиваленцией двух высказываний P и Q называется высказывание, истинное тогда и только тогда, когда истинностные значения P и Q совпадают. Эквиваленция высказываний P и Q обозначается через $P \sim Q$ и читается как " P эквивалентно Q " (используются также слова "равносильно", "тогда и только тогда", "если A , то B , и обратно", " A , если и только если B ", "для A необходимо и достаточно B "). Эквиваленция определяется также таблицей истинности (см. табл. 5).

Таблица 5

P	Q	$P \sim Q$
И	И	И
И	Л	Л
Л	И	Л
Л	Л	И

§4 Формулы логики высказываний

Мы определим формальный язык для описания логики высказываний. Это описание чисто синтаксическое и оно не требует, чтобы формулы логики высказывания имели какую-то семантику (смысл).

Алфавитом называется любое непустое множество. Элементы этого множества называются *символами* данного алфавита. *Словом* в данном алфавите называется произвольная конечная последовательность символов (возможно, пустая). Слово a называется *подсловом* слова b , если $b = b_1ab_2$ для некоторых слов b_1 и b_2 (мы используем обозначение $\alpha\beta$ для конкатенации (соединения) двух слов α и β в одно слово).

Алфавит логики высказываний содержит следующие символы: высказывательные переменные X_1, X_2, \dots ; логические символы $\&, \vee, \neg, \supset, \sim$; символы скобок $(,)$.

Слово в алфавите логики высказываний называется *формулой*, если оно удовлетворяет следующему определению:

- 1) любая высказывательная переменная - формула;
- 2) если A и B - формулы, то $(\neg A)$, $(A \& B)$, $(A \vee B)$, $(A \supset B)$, $(A \sim B)$ - формулы.
- 3) только те слова являются формулами, для которых это следует из 1) и 2).

Подформулой формулы A называется любое подслово A , само являющееся формулой.

Для упрощения записи будем в формуле опускать внешние скобки и те пары скобок, без которых можно восстановить эту формулу, если пользоваться следующим правилом: каждое вхождение знака \neg относится к наикратчайшей подформуле, следующей за ним.

Пример 2.1. Слово $(X_1 \& X_2) \supset X_3 \neg X_1$ не является формулой, а слова $(\neg X_1 \supset X_2) \vee X_1$, $(X_1 \sim X_2) \supset \neg X_3$ - формулы. Слова $X_1 \sim X_2$, $\neg X_3$, X_1 , X_2 , X_3 - подформулы последней формулы.

Принцип математической индукции, который будем использовать в рассуждениях, формулируется следующим образом: если какое-то высказывание $P(t)$, зависящее от натурального параметра t , доказано для $t = 0$ и при произвольном t удастся из истинности $P(t)$ обосновать истинность $P(t+1)$, то $P(t)$ истинно для всех t .

Будем применять также и другую формулировку этого принципа: если $P(t)$ истинно при $t = 0$ и для любого t из истинности $P(s)$ при всех $s \leq t$ следует истинность $P(t+1)$, то $P(t)$ истинно для всех t .

Применительно к высказывательным формулам принцип математической индукции можно сформулировать следующим образом: если какое-то утверждение $P(F)$, зависящее от параметра F , который пробегает все множество высказывательных формул,

- истинно для всех формул, не содержащих логических символов (т.е. формул вида X_i);
- и при любом натуральном n из того, что $P(F)$ истинно для всех формул F с числом логических символов, меньших n , следует, что $P(F)$ истинно для всех формул с n логическими символами, то $P(F)$ истинно для всех формул F .

Пример 2.2. Докажем методом математической индукции, что S_n - сумма натуральных чисел от 1 до n - равна $1/2 (n+1)n$.

Базис индукции. $S_1 = 1 = (2 \cdot 1)/2$.

Индуктивный переход. Пусть $S_n = 1/2 (n+1)n$. Тогда $S_{n+1} = S_n + n+1 = (n+1)n/2 + n+1 = (n^2 + n + 2n + 2)/2 = (n+1)(n+2)/2$, ч. и т. д.

Каждому распределению истинностных значений высказывательных переменных, входящих в ту или иную формулу, соответствует согласно истинностным таблицам для логических связок, некоторое истинностное значение этой формулы логики высказываний.

Таким образом, если $\{X_1, X_2, \dots, X_n\}$ - множество высказывательных переменных, входящих в формулу F , то формула F определяет *истинностную функцию* $\{И, Л\}^n \rightarrow \{И, Л\}$, которая графически может быть представлена истинностной таблицей для этой формулы.

Пример 2.3.

1) Таблица 6 - таблица истинности для формулы $(X_1 \supset X_2) \vee (X_1 \supset (X_1 \& X_2))$. Здесь каждая строка содержит некоторое распределение истинностных значений для переменных X_1 и X_2 и соответствующие истинностные зна-

чения, принимаемые различными подформулами, которые возникают при построении формулы $(X_1 \supset X_2) \vee (X_1 \supset (X_1 \& X_2))$.

Таблица 6

X_1	X_2	$X_1 \supset X_2$	$X_1 \& X_2$	$X_1 \supset (X_1 \& X_2)$	$(X_1 \supset X_2) \vee (X_1 \supset (X_1 \& X_2))$
И	И	И	И	И	И
И	Л	Л	Л	Л	Л
Л	И	И	Л	Л	И
Л	Л	И	Л	И	И

2) Таблица 7 - таблица истинности для формулы $(X_1 \supset X_2) \vee \neg X_3$.

Таблица 7

X_1	X_2	X_3	$\neg X_3$	$X_1 \supset X_2$	$(X_1 \supset X_2) \vee \neg X_3$
И	И	И	Л	И	И
И	И	Л	И	И	И
И	Л	И	Л	Л	Л
И	Л	Л	И	Л	И
Л	И	И	Л	И	И
Л	И	Л	И	И	И
Л	Л	И	Л	И	И
Л	Л	Л	И	И	И

Если в формуле имеется k различных переменных, то имеем 2^k различных распределений истинностных значений для переменных и, следовательно, истинностная таблица для такой формулы содержит 2^k строк.

§5 Равносильность формул

Пусть A и B - две формулы и $\{X_1, X_2, \dots, X_n\}$ - множество всех высказывательных переменных, входящих в формулу A и/или в формулу B . Будем называть эти формулы равносильными, если при любом распределении истинностных значений для переменных $\{X_1, X_2, \dots, X_n\}$, они принимают одинаковые значения. Равносильность формул A и B будем обозначать $A \equiv B$.

Нужно различать символы \sim и \equiv . Так, \sim является символом формального языка, с по-



Огастес де Морган

мощью которого строятся формулы, а символ \equiv обозначает отношение на множестве формул.

Отношение равносильности есть отношение эквивалентности. Действительно, оно рефлексивно, так как для любой формулы A $A \equiv A$; симметрично, так как для любых формул A и B , если $A \equiv B$, то $B \equiv A$; транзитивно, так как для любых формул A , B , C , если $A \equiv B$ и $B \equiv C$, то $A \equiv C$.

Основные равносильности. Для любых формул A , B , C справедливы следующие равносильности:

1. $A \& B \equiv B \& A$ (коммутативность $\&$);
2. $A \& A \equiv A$ (идемпотентность $\&$);
3. $A \& (B \& C) \equiv (A \& B) \& C$ (ассоциативность $\&$);
4. $A \vee B \equiv B \vee A$ (коммутативность \vee);
5. $A \vee A \equiv A$ (идемпотентность \vee);
6. $A \vee (B \vee C) \equiv (A \vee B) \vee C$ (ассоциативность \vee);
7. $A \vee (B \& C) \equiv (A \vee B) \& (A \vee C)$ (дистрибутивность \vee относительно $\&$);
8. $A \& (B \vee C) \equiv (A \& B) \vee (A \& C)$ (дистрибутивность $\&$ относительно \vee);
9. $A \& (A \vee B) \equiv A$ (первый закон поглощения);
10. $A \vee (A \& B) \equiv A$ (второй закон поглощения);
11. $\neg \neg A \equiv A$ (снятия двойного отрицания);
12. $\neg (A \& B) \equiv \neg A \vee \neg B$ (первый закон де Моргана);
13. $\neg (A \vee B) \equiv \neg A \& \neg B$ (второй закон де Моргана);
14. $A \equiv (A \& B) \vee (A \& \neg B)$ (первый закон расщепления);
15. $A \equiv (A \vee B) \& (A \vee \neg B)$ (второй закон расщепления);
16. $A \supset B \equiv (A \supset B) \& (B \supset A) \equiv (A \& B) \vee (\neg A \& \neg B)$;
17. $A \supset B \equiv \neg A \vee B \equiv \neg (A \& \neg B)$;
18. $A \vee B \equiv \neg A \supset B \equiv \neg (\neg A \& \neg B)$;
19. $A \& B \equiv \neg (A \supset \neg B) \equiv \neg (\neg A \vee \neg B)$.

Равносильности 16-19 показывают, что одни связки могут быть выражены через другие.

Все эти равносильности легко доказываются либо с помощью таблиц истинности либо без них. В качестве примера, докажем 7 с помощью таблицы истинности.

Таблица 8

A	B	C	$B \& C$	$A \vee (B \& C)$	$A \vee B$	$A \vee C$	$(A \vee B) \& (A \vee C)$
И	И	И	И	И	И	И	И
И	И	Л	Л	И	И	И	И
И	Л	И	Л	И	И	И	И
И	Л	Л	Л	И	И	И	И
Л	И	И	И	И	И	И	И
Л	И	Л	Л	Л	И	Л	Л

Л	Л	И	Л	Л	Л	И	Л
Л	Л	Л	Л	Л	Л	Л	Л

Доказательство 12 без таблицы истинности.

Пусть на некотором наборе истинностных значений переменных формула $\neg(A \& B)$ принимает значение Л. Тогда формула $A \& B$ принимает значение И, а поэтому обе формулы A и B принимают значение И. Но в этом случае, очевидно, и правая часть равносильности 12 принимает значение Л. И наоборот, пусть формула $\neg A \vee \neg B$ принимает значение Л. Тогда формулы $\neg A$, $\neg B$ принимают значение Л, а формулы A , B - значение И. Очевидно, что и левая часть равносильности 12 принимает значение Л.

В силу транзитивности отношения равносильности, если $A_1 \equiv A_2$, $A_2 \equiv A_3, \dots, A_{k-1} \equiv A_k$, то $A_1 \equiv A_k$. В таком случае для простоты будем записывать цепочку $A_1 \equiv A_2 \equiv A_3 \equiv \dots \equiv A_{k-1} \equiv A_k$.

Приведем правило, с помощью которого можно переходить от одних равносильностей к другим.

Лемма 2.1. Пусть $A \equiv B$ и C - произвольная формула. Тогда $\neg A \equiv \neg B$, $A \& C \equiv B \& C$, $C \& A \equiv C \& B$, $A \vee C \equiv B \vee C$, $C \vee A \equiv C \vee B$, $A \supset C \equiv B \supset C$, $C \supset A \equiv C \supset B$, $A \sim C \equiv B \sim C$, $C \sim A \equiv C \sim B$.

Докажем например равносильность $A \supset C \equiv B \supset C$. Пусть на произвольном наборе высказывательных переменных формулы A и B принимают одинаковое значение (скажем, s). Пусть t - значение C на этом распределении истинностных значений. Обе части рассматриваемой равносильности принимают одно и то же значение $s \supset t$.

Лемма 2.2. Пусть $A \equiv B$ и C - формула, в которой выделено одно вхождение переменной X_i . Пусть C_A получается из C заменой этого вхождения X_i на A , а C_B - из C заменой того же вхождения X_i на B . Тогда $C_A \equiv C_B$.

Докажем это индукцией по числу n логических символов в C .

Базис индукции. Если $n=0$, то формула C должна совпадать с X_i (так как в ней имеется вхождение переменной X_i). В этом случае C_A есть A , C_B есть B , $C_A \equiv C_B$ - не что иное, как $A \equiv B$.

Индуктивный переход. Пусть лемма доказана для числа логических символов меньше n и пусть C - формула с n логическими символами. Она имеет вид $\neg D$, или $D \& E$, или $D \vee E$, или $D \supset E$, или $D \sim E$, причем в первом случае выделенное вхождение X_i содержится в D , а в остальных случаях - либо в D , либо в E , но не в D и E сразу. Рассмотрим, например, случай, когда C имеет вид $D \supset E$ и выделенное вхождение X_i содержится в D . Заменяя X_i в этом вхождении в D на A и B , получаем соответственно формулы D_A и D_B . Ясно, что C_A есть $D_A \supset E$, а C_B есть $D_B \supset E$. Так как в формуле D , меньше логических символов, чем в C , то $D_A \equiv D_B$. Применим теперь лемму 2.1 в случае $A \supset C \equiv B \supset C$, где в роли A выступает D_A и в роли B - D_B , в роли C - E . В результате получаем $C_A \equiv C_B$. Другие случаи рассматриваются аналогично.

Теорема 2.1. (правило равносильных преобразований). Пусть C_A - формула, содержащая A в качестве своей подформулы. Пусть C_B получается из C_A заменой A в этом вхождении на B . Тогда, если $A \equiv B$, то $C_A \equiv C_B$.

Рассмотрим произвольную переменную X_i и получим формулу C из C_A заменой A на X_i . Будем считать это вхождение X_i в C выделенным. Тогда C , A , B , C_A , C_B удовлетворяют условиям леммы 2.2, а значит, $C_A \equiv C_B$.

Теорема 2.2 (правило устранения логических символов \supset и \sim). Для каждой формулы можно указать равносильную ей формулу, не содержащую логических символов \supset и \sim .

В самом деле, опираясь на правило равносильных преобразований, можно в исходной формуле каждую подформулу вида $A \sim B$ заменить на $(A \& B) \vee (\neg A \& \neg B)$, а каждую подформулу вида $A \supset B$ на $\neg A \vee B$ (см. равносильности 16 и 17).

Пример 2.4. Формула $(X_1 \supset (X_2 \supset X_3)) \sim \neg (X_2 \supset X_1)$ преобразуется следующим образом: $(X_1 \supset (X_2 \supset X_3)) \sim \neg (X_2 \supset X_1) \equiv (X_1 \supset (\neg X_2 \vee X_3)) \sim (\neg (\neg X_2 \vee X_1)) \equiv (\neg X_1 \vee (\neg X_2 \vee X_3)) \sim (\neg (\neg X_2 \vee X_1)) \equiv ((\neg X_1 \vee (\neg X_2 \vee X_3)) \& \neg (\neg X_2 \vee X_1)) \vee (\neg (\neg X_1 \vee (\neg X_2 \vee X_3)) \& \neg \neg (\neg X_2 \vee X_1)).$

§6 Тавтологично-истинные формулы

Определение. Формула, которая истинна независимо от того, какие значения принимают встречающиеся в ней высказывательные переменные, называется *тавтологией* (или *тождественно-истинной* формулой).

Формула является тавтологией тогда и только тогда, когда соответствующая истинностная функция принимает только значение И, или, что тоже, если в ее таблице истинности столбец под самой формулой состоит только из букв И.

Определение.

- Формула называется *выполнимой*, если на некотором наборе распределения истинностных значений переменных она принимает значение И.
- Формула называется *тождественно-ложной* или *противоречием*, если она ложна независимо от того, какие значения принимают встречающиеся в ней высказывательные переменные.
- Формула называется *опровержимой*, если при некотором распределении истинностных значений переменных она принимает значение Л.

Приведем утверждения, которые являются очевидными следствиями данных определений:

- A - тавтология тогда и только тогда, когда A не является опровержимой;
- A - тождественно-ложна тогда и только тогда, когда A не является выполнимой;

- A - тавтология тогда и только тогда, когда $\neg A$ - тождественно-ложна;

- A - тождественно-ложна тогда и только тогда, когда $\neg A$ - тавтология;

- $A \sim B$ - тавтология тогда и только тогда, когда A и B - равносильны.

С точки зрения логики тавтологии суть не что, иное, как логические законы, ибо при любой подстановке вместо переменных тавтологии конкретных высказываний в результате получим истинное высказывание.

Перечислим наиболее важные тавтологии (A , B , C - произвольные формулы):

1. $A \vee \neg A$ (закон исключенного третьего или *tertium nondatur*);
2. $A \supset A$;
3. $A \supset (B \supset A)$;
4. $(A \supset B) \supset ((B \supset C) \supset (A \supset C))$ (цепное рассуждение);
5. $(A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))$;
6. $(A \& B) \supset A$; $(A \& B) \supset B$;
7. $A \supset (B \supset (A \& B))$;
8. $A \supset (A \vee B)$; $B \supset (A \vee B)$;
9. $(\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B)$;
10. $((A \supset B) \supset A) \supset A$ (закон Пирса).

Каждую из этих тавтологий можно обосновать, например, составив таблицу и вычислив по ней значение формулы при произвольных значениях A , B и C .

§7 Нормальные формы формул

Содержание этого параграфа изложим, следуя [24].

Будем рассматривать формулы, содержащие только логические операции $\&$, \vee , \neg . Символы $\&$ и \vee называются *двойственными* друг другу.

Формула A^* называется *двойственной* формуле A , если она получена из A одновременной заменой всех символов $\&$, \vee на двойственные. Например, формула $X_1 \& (X_2 \vee \neg X_1)$ двойственна формуле $X_1 \vee (X_2 \& \neg X_1)$. Заметим, что A^* содержит те же высказывательные переменные, что и A .

Очевидно, что $(A^*)^*$ совпадает с A .

Пусть $\langle X_1, X_2, \dots, X_n \rangle$ - некоторый упорядоченный список высказывательных переменных. Пусть $\langle s_1, s_2, \dots, s_n \rangle$ - соответствующий список распределения истинностных значений для этих переменных (каждое s_i , $i=1, 2, \dots, n$, есть И или Л). Список истинностных значений $\langle t_1, t_2, \dots, t_n \rangle$ назовем двойственным к списку $\langle s_1, s_2, \dots, s_n \rangle$, если $\langle t_1, t_2, \dots, t_n \rangle$ получается из $\langle s_1, s_2, \dots, s_n \rangle$ заменой всех И на Л и всех Л на И.

Лемма 2.3. Пусть A - формула и $\langle X_1, X_2, \dots, X_n \rangle$ - высказывательные переменные этой формулы. Пусть $\langle s_1, s_2, \dots, s_n \rangle$ - соответствующий список распределения истинностных значений для этих переменных. Тогда A

принимает значение И при этих значениях переменных тогда и только тогда, когда A^* принимает значение Л на списке истинностных значений $\langle t_1, t_2, \dots, t_n \rangle$, двойственном к списку $\langle s_1, s_2, \dots, s_n \rangle$.

Доказательство. Проведем математическую индукцию по числу k логических связок в формуле A .

Базис индукции. При $k=0$ формула A совпадает с одной из высказывательных переменных X_i и A^* также совпадает с этой переменной. Поэтому формулы A и A^* имеют противоположные истинностные значения s_i и t_i и утверждение леммы выполнено.

Индуктивный переход. Пусть утверждение леммы справедливо при числе логических операций, меньшем k . Докажем, что оно остается справедливым и при числе операций, равном k . Рассмотрим три различных случая, в зависимости от того какой вид имеет формула A .

1. Пусть формула A совпадает с формулой $\neg B$. Тогда A^* совпадает с формулой $\neg (B^*)$. Пусть $\langle s_1, s_2, \dots, s_n \rangle$ - некоторый список истинностных значений. Истинностное значение формулы B на этом списке противоположно истинностному значению формулы A на этом же списке. Так как количество логических операций в формуле B меньше k , то значение формулы B^* на двойственном списке $\langle t_1, t_2, \dots, t_n \rangle$ противоположно значению B на списке $\langle s_1, s_2, \dots, s_n \rangle$. Следовательно, значение $\neg (B^*)$ на списке $\langle t_1, t_2, \dots, t_n \rangle$ совпадает со значением формулы B на списке распределения истинностных значений $\langle s_1, s_2, \dots, s_n \rangle$. Отсюда сразу получаем искомое утверждение для формулы A .

2. Пусть формула A имеет вид $B \& C$. Из определения операции двойственности сразу следует, что $A^* = (B \& C)^* = B^* \vee C^*$. Для распределения истинностных значений $\langle s_1, s_2, \dots, s_n \rangle$ формула A истинна тогда и только тогда, когда формулы B и C также истинны. Поскольку B и C имеют логических операций меньше k , то значения формул B^* и C^* ложны на двойственном наборе истинностных значений $\langle t_1, t_2, \dots, t_n \rangle$. На этом же наборе ложна и формула $B^* \vee C^* = A^*$. Эти рассуждения показывают справедливость искомого утверждения в данном случае.

3. Пусть формула A имеет вид $B \vee C$. Из определения операции двойственности сразу следует, что $A^* = (B \vee C)^* = B^* \& C^*$. Для распределения истинностных значений $\langle s_1, s_2, \dots, s_n \rangle$ формула A ложна тогда и только тогда, когда формулы B и C также ложны. Поскольку B и C имеют логических операций меньше k , то значения формул B^* и C^* истинны на двойственном наборе истинностных значений $\langle t_1, t_2, \dots, t_n \rangle$. На этом же наборе истинна и формула $B^* \& C^* = A^*$. Тем самым мы получаем справедливость искомого утверждения и в данном случае.

Разбор всех случаев позволил нам закончить индуктивный переход и убедиться в справедливости леммы.

Теорема 2.3. (принцип двойственности) Если $A \equiv B$, то $A^* \equiv B^*$.

Доказательство. Пусть $\langle t_1, t_2, \dots, t_n \rangle$ - произвольный набор распределения истинностных значений для истинностных переменных в формуле A^* . Тогда истинностное значение A^* противоположно истинностному значению A (а, следовательно, и B) на двойственном списке $\langle s_1, s_2, \dots, s_n \rangle$. С другой стороны, значение формулы B^* на списке истинностных значений $\langle t_1, t_2, \dots, t_n \rangle$ противоположно истинностному значению B на списке $\langle s_1, s_2, \dots, s_n \rangle$. Следовательно, значения формул A^* и B^* при произвольном распределении истинностных значений переменных совпадают. Что и требовалось доказать.

Принцип двойственности можно использовать для нахождения новых равносильностей. Например, используя следующий частный случай дистрибутивности $\&$ относительно \vee

$$X \& (Y \vee Z) \equiv (X \& Y) \vee (X \& Z),$$

получаем равносильность

$$X \vee (Y \& Z) \equiv (X \vee Y) \& (X \vee Z).$$

Заметим, что в силу ассоциативности операций $\&$ и \vee , как бы мы не расставляли скобки в выражениях $A_1 \& A_2 \& \dots \& A_n$ и $A_1 \vee A_2 \vee \dots \vee A_n$ ($n > 3$), всегда будем приходить к равносильным формулам. Допуская некоторую вольность речи, каждое из этих выражений будем считать формулами и называть соответственно многочленной конъюнкцией и дизъюнкцией формул A_1, A_2, \dots, A_n . Для этих формул, используя, например, индукцию по $\max(k, n)$, нетрудно получить равносильности, выражающие обобщенную дистрибутивность (для простоты записи, положим $k=2, n=3$):

$$\begin{aligned} (A_1 \vee A_2) \& (B_1 \vee B_2 \vee B_3) &\equiv (A_1 \& B_1) \vee (A_1 \& B_2) \vee (A_1 \& B_3) \vee \\ &\quad (A_2 \& B_1) \vee (A_2 \& B_2) \vee (A_2 \& B_3), \\ (A_1 \& A_2) \vee (B_1 \& B_2 \& B_3) &\equiv (A_1 \vee B_1) \& (A_1 \vee B_2) \& (A_1 \vee B_3) \& \\ &\quad (A_2 \vee B_1) \& (A_2 \vee B_2) \& (A_2 \vee B_3). \end{aligned}$$

Точно также получаем обобщенные законы де Моргана:

$$\neg(A_1 \& A_2 \& \dots \& A_n) \equiv \neg A_1 \vee \neg A_2 \vee \dots \vee \neg A_n,$$

$$\neg(A_1 \vee A_2 \vee \dots \vee A_n) \equiv \neg A_1 \& \neg A_2 \& \dots \& \neg A_n.$$

Определим теперь некоторые канонические виды формул.

Формула называется *элементарной конъюнкцией*, если она является конъюнкцией (быть может, одночленной) переменных и отрицаний переменных. Например, формулы $\neg X_2, X_1, X_1 \& X_2, X_1 \& \neg X_2 \& \neg X_1 \& X_4$ являются элементарными конъюнкциями.

Говорят, что формула находится в *дизъюнктивной нормальной форме* (ДНФ), если она является дизъюнкцией (быть может, одночленной) элементарных конъюнкций. Например, формулы $\neg X_2, X_1, X_1 \& X_2, X_1 \& \neg X_2 \& \neg X_1 \& X_4, (X_1 \& X_2) \vee (X_1 \& \neg X_2 \& \neg X_1 \& X_4) \& X_1$ находятся в ДНФ.

Теорема 2.4 (о приведении к ДНФ). Для любой формулы A можно найти такую формулу B , находящуюся в ДНФ, что $A \equiv B$. Формула B называется *дизъюнктивной нормальной формой формулы A* .

Доказательство теоремы распадается на три этапа:

1-ый этап. Для формулы A строим такую формулу A_1 , что $A \equiv A_1$ и в A_1 не содержатся символы \sim и \supset (теорема 2.2).

2-ой этап. Докажем теперь, что для формулы A_1 можно найти формулу A_2 такую, что $A_1 \equiv A_2$ и в A_2 отрицание находится только перед переменными. Такая формула называется формулой с "тесными" отрицаниями. Докажем это утверждение математической индукцией по числу n логических операций формулы A_1 .

Базис индукции. Если $n=0$, то A_1 есть какая-то переменная X_i . В качестве A_2 нужно взять X_i .

Индуктивный переход. Пусть утверждение выполняется для всех формул A_1 с числом связок меньше n . Пусть в формуле A_1 содержится точно n логических связок. Рассмотрим следующие случаи:

1) A_1 имеет вид $B_1 \vee C_1$. Тогда в B_1 , C_1 логических символов меньше, чем n . Поэтому существуют формулы B_2 , C_2 такие, что $B_1 \equiv B_2$, $C_1 \equiv C_2$ и в B_2 и C_2 отрицание встречается только перед переменными. Ясно, что $B_2 \vee C_2$ равносильна A_1 и является формулой с "тесными" отрицаниями;

2) A_1 имеет вид $B_1 \& C_1$. Доказательство аналогично предыдущему случаю;

3) A_1 имеет вид $\neg \neg B_1$. Тогда $A_1 \equiv B_1$ в B_1 логических символов меньше, чем n . Поэтому к B_1 применимо индуктивное предположение;

4) A_1 имеет вид $\neg(B_1 \vee C_1)$. Тогда $A_1 \equiv \neg B_1 \& \neg C_1$ и в $\neg B_1$, $\neg C_1$ логических символов меньше, чем n . Поэтому существуют такие формулы B_2 , C_2 , что $\neg B_1 \equiv B_2$, $\neg C_1 \equiv C_2$ и в B_2 и C_2 отрицание встречается только перед переменными. Ясно, что $A_1 \equiv B_2 \& C_2$ и $B_2 \& C_2$ является формулой с "тесными" отрицаниями;

5) A_1 имеет вид $\neg(B_1 \& C_1)$. Тогда $A_1 \equiv \neg B_1 \vee \neg C_1$, и далее поступаем как в предыдущем случае.

При практическом преобразовании встречающиеся в формуле отрицания просто передвигают к переменным, используя законы де Моргана и уничтожая пары стоящих рядом отрицаний, если таковые встречаются.

Пример 2.5. Преобразуем к формуле с "тесными" отрицаниями:

$$\begin{aligned} \neg(\neg \neg(X_1 \& \neg X_2) \vee (X_2 \& \neg X_1)) &\equiv \neg \neg \neg(X_1 \& \neg X_2) \& \neg(X_2 \& \neg X_1) \equiv \\ \neg(X_1 \& \neg X_2) \& (\neg X_2 \vee \neg \neg X_1) &\equiv (\neg X_1 \vee \neg \neg X_2) \& (\neg X_2 \vee X_1) \equiv \\ (\neg X_1 \vee X_2) \& (\neg X_2 \vee X_1). \end{aligned}$$

3-ий этап. Полученная формула A_2 построена из переменных и их отрицаний с помощью многочленных конъюнкций и дизъюнкций. Применив теперь обобщенную дистрибутивность $\&$ относительно \vee , последовательно преобразуем формулу (аналогично приведению алгебраического выражения, составленного из переменных, с помощью сложений и умножений, к виду многочлена). Заметим, что при этом \vee аналогично сложе-

нию, а $\&$ - умножению. Полученная в результате преобразований формула В будет удовлетворять требованиям теоремы.

Пример 2.6. Применим преобразования третьего этапа к формуле с "тесными" отрицаниями, полученной в примере 2.5:

$$(\neg X_1 \vee X_2) \& (\neg X_2 \vee X_1) \equiv (\neg X_1 \& \neg X_2) \vee (\neg X_1 \& X_1) \vee (X_2 \& \neg X_2) \vee (X_2 \& X_1).$$

Говорят, что формула А находится в *конъюнктивной нормальной форме* (КНФ), если формула A^* определена (т. е. в А нет символов \sim и \supset) и находится в ДНФ.

КНФ можно дать и другое равносильное определение. Формулу называют *элементарной дизъюнкцией*, если она является дизъюнкцией (возможно, одночленной) переменных и отрицаний переменных. Формула находится в КНФ, если она является конъюнкцией (возможно, одночленной) элементарных дизъюнкций.

Теорема 2.5 (о приведении к КНФ). Для любой формулы А можно найти такую формулу В, что В находится в КНФ и $A \equiv B$. Формула В называется *конъюнктивной нормальной формой формулы А*.

Первое доказательство. Пусть $A \equiv A_1$ и A_1 не содержит символов \sim , \supset . Пусть B_1 - дизъюнктивная нормальная форма формулы A_1^* . Тогда B_1^* находится в КНФ и, кроме того, по принципу двойственности $B_1^* \equiv (A_1^*)^* \equiv A_1 \equiv A$. Значит, B_1^* удовлетворяет требованиям теоремы.

Второе доказательство. Применив первые два этапа из доказательства теоремы 2.6 о ДНФ, получим формулу A_2 , равносильную А, не содержащую символов \sim , \supset и содержащую отрицания только перед переменными. Преобразуем теперь A_2 как алгебраическое выражение, считая на этот раз $\&$ аналогом сложения, а \vee - аналогом умножения и применяя дистрибутивность \vee относительно $\&$. Приведение формулы A_2 к виду многочлена даст на этот раз КНФ.

Пример 2.7. Приведем к КНФ формулу:

$$\begin{aligned} (X_1 \& X_2) \sim (\neg X_1 \& X_3) &\equiv \\ ((X_1 \& X_2) \& (\neg X_1 \& X_3)) \vee (\neg (X_1 \& X_2) \& \neg (\neg X_1 \& X_3)) &\equiv \\ (X_1 \& X_2 \& \neg X_1 \& X_3) \vee ((\neg X_1 \vee \neg X_2) \& (\neg \neg X_1 \vee \neg X_3)) &\equiv \\ (X_1 \& X_2 \& \neg X_1 \& X_3) \vee ((\neg X_1 \vee \neg X_2) \& (X_1 \vee \neg X_3)) &\equiv \\ (X_1 \vee \neg X_1 \vee \neg X_2) \& (X_1 \vee X_1 \vee \neg X_3) \& (X_2 \vee \neg X_1 \vee \neg X_2) \& (X_2 \vee X_1 \vee \neg X_3) \& \\ (\neg X_1 \vee \neg X_1 \vee \neg X_2) \& (X_1 \vee X_1 \vee \neg X_3) \& (X_3 \vee \neg X_1 \vee \neg X_2) \& (X_3 \vee X_1 \vee \neg X_3). \end{aligned}$$

Заметим, что первое преобразование основано на равносильности 16. Нетрудно видеть, что ДНФ и КНФ не являются однозначно определенными. Формула может иметь несколько равносильных друг другу ДНФ и КНФ.

§8 Разрешимость для логики высказываний

Проблемой разрешимости для логики высказываний называют следующую проблему: существует ли алгоритм, который позволил бы для произвольной формулы в конечном числе шагов определить, является ли она тавтологией?

Ясно, что эта проблема разрешима, поскольку всегда можно перебрать все оценки списка переменных и вычислить на них значения формулы. Опишем другую более эффективную процедуру распознавания, связанную с приведением формулы к КНФ.

Теорема 2.6. Формула является тавтологией в том и только том случае, если в ее КНФ в любую из элементарных дизъюнкций в качестве дизъюнктивных членов входит какая-нибудь переменная и ее отрицание.

Доказательство. Достаточность. Пусть в каждую элементарную дизъюнкцию в качестве дизъюнктивных членов входят некоторые переменные вместе со своими отрицаниями, тогда каждая такая элементарная дизъюнкция является тавтологией и, следовательно, КНФ является также тавтологией.

Необходимость. Нам понадобится следующее вспомогательное утверждение: "Если элементарная дизъюнкция является тавтологией, то она содержит какую-нибудь переменную вместе с отрицанием".

От противного. Пусть элементарная дизъюнкция не содержит никакой переменной вместе с ее отрицанием. Выберем такое распределение истинностных значений, при котором эта элементарная дизъюнкция будет иметь ложное значение. Для этого любой переменной, входящей без отрицания, положим значение Л, а любой переменной, входящей с отрицанием, положим значение И. Тогда элементарная дизъюнкция есть дизъюнкция ложных значений и, следовательно, будет ложной.

Вернемся к доказательству необходимости. Для того, чтобы конъюнкция формул была тавтологией, необходимо, чтобы каждый конъюнктивный член был тавтологией. Иначе, мы сразу приходим к противоречию.

Отсюда получаем, что КНФ исходной формулы, будучи конъюнкцией элементарных дизъюнкций, необходимо в каждой элементарной дизъюнкции содержит какую-нибудь переменную вместе с отрицанием.

Двойственное утверждение справедливо и для противоречия (тождественно-ложной) формулы.

Теорема 2.7. Формула является противоречием в том и только том случае, если в ее ДНФ каждая элементарная конъюнкция одновременно содержит в качестве конъюнктивных членов какую-нибудь переменную и ее отрицание.

Доказательство. Пусть формула А равносильна формуле В, находящейся в ДНФ. Тогда легко показать, что $\neg В$, используя законы де Моргана и снятие двойного отрицания, преобразуется в формулу С, находящуюся в КНФ. При этом преобразовании любая элементарная конъюнкция переходит в элементарную дизъюнкцию. Имеем $\neg А \equiv С$ и по предыдущей теореме

2.6 формула $\neg A$ - тавтология тогда и только тогда, когда каждая элементарная дизъюнкция в S содержит какую-нибудь переменную и ее отрицание. Переходя от формул $\neg A$ и S к формулам A и B , мы получаем справедливость теоремы.

Пора чудес прошла, и нам
Подыскивать приходится причины
Всему, что совершается на свете.
В. Шекспир

Глава 3. БУЛЕВЫ АЛГЕБРЫ

Характерная черта современных компьютеров - сведение всех вычислительных структур (чисел, символов, массивов и т. п.) к двоичным словам и алгоритмам их обработки. С математической точки зрения мы имеем дело с частным случаем булевых алгебр.

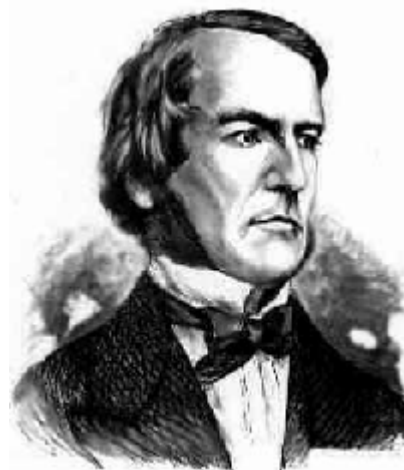
Аксиома - это истина, на которую не хватило доказательств.

В. Хмурый

§1 Абстрактное определение булевых алгебр

Определение. Множество элементов B с заданным на нем двуместными операциями \wedge и \vee (конъюнкцией и дизъюнкцией) и одноместной операцией \neg (отрицанием) называется *булевой алгеброй*, если выполнены следующие аксиомы (f, g, h - произвольные элементы множества):

$$\begin{aligned} f \wedge g &= g \wedge f, f \vee g = g \vee f \text{ (коммутативность);} \\ (f \wedge g) \wedge h &= f \wedge (g \wedge h), (f \vee g) \vee h = f \vee (g \vee h) \text{ (ассоциативность);} \\ f \wedge f &= f, f \vee f = f \text{ (идемпотентность);} \end{aligned}$$



$f \wedge (g \vee f) = f$, $f \vee (g \wedge f) = f$ (законы поглощения);
 $f \wedge (g \vee h) = (f \wedge g) \vee (f \wedge h)$, $f \vee (g \wedge h) = (f \vee g) \wedge (f \vee h)$ (дистрибутивность);
 $\neg(\neg f) = f$ (закон инволюции);
 $\neg(f \wedge g) = \neg f \vee \neg g$, $\neg(f \vee g) = \neg f \wedge \neg g$ (законы де Моргана);
 $f \wedge (g \vee \neg g) = f$, $f \vee (g \wedge \neg g) = f$ (законы нейтральности).

Из перечисленных законов можно вывести, что для произвольных f , g справедливы равенства $f \wedge \neg f = g \wedge \neg g$ и $f \vee \neg f = g \vee \neg g$. Например, в силу законов нейтральности и коммутативности имеем

$$f \wedge \neg f = (f \wedge \neg f) \vee (g \wedge \neg g) = (g \wedge \neg g) \vee (f \wedge \neg f) = g \wedge \neg g.$$

Если обозначить $f \wedge \neg f$ через \in и $f \vee \neg f$ через \notin , то выполняются равенства

$$\begin{aligned}
 \neg \notin &= \in, & \neg \in &= \notin, \\
 f \wedge \notin &= f, & f \wedge \in &= \in, \\
 f \vee \notin &= \notin, & f \vee \in &= f.
 \end{aligned}$$

Булева алгебра называется *вырожденной*, если \in и \notin совпадают; в таком случае ввиду равенств $f = f \wedge \notin = f \wedge \in = \in$ она не содержит никаких других элементов, а значит состоит ровно из одного элемента. Всякая невырожденная булева алгебра - а только такие и будут рассматриваться в дальнейшем - содержит два *нейтральных элемента*: \in (*нулевой элемент*) и \notin (*единичный элемент*).

Из $f \wedge g = \notin$ следует, что $f = g = \notin$. Действительно, $f = f \vee (f \wedge g) = f \vee \notin = \notin$. Этот факт называют неразложимостью нейтрального элемента \notin .

Данное выше определение булевых алгебр ничего не говорит о том, а существуют ли вообще булевы алгебры? Может быть определение булевых алгебр является противоречивым и поэтому нет ни одного множества, на котором можно было бы ввести булевы операции, удовлетворяющие указанным аксиомам?

К счастью, это не так. Сейчас укажем некоторые модели - конкретные примеры булевых алгебр.

Двоичная модель

Наиболее простая из булевых алгебр (и вместе с тем одна из наиболее важных для компьютерной науки) содержит только два (нейтральных) элемента и операции на ней вводятся с помощью следующих таблиц значений.

\wedge	\notin	\in
\notin	\notin	\in
\in	\in	\in

\vee	\notin	\in
--------	----------	-------

\varnothing	\varnothing	\varnothing
\in	\varnothing	\in

	\varnothing	\in
\neg	\in	\varnothing

Модель исчисления высказываний

Пусть B - множество высказываний с обычными логическими операциями конъюнкции, дизъюнкции и отрицания и равенство высказываний интерпретируется как их равносильность.

Во второй главе показано, что операции \vee и \wedge ассоциативны, коммутативны, и каждая из них дистрибутивна относительно другой. Если мы обозначим любое противоречие через L , а любую тавтологию через I , то мы можем считать, что I и L являются элементами B (так как все тавтологии равносильны, и все противоречия также равносильны). Легко проверить, что множество B с логическими операциями является булевой алгеброй, а элементы I и L будут нейтральными элементами.

Теоретико-множественная модель

Пусть A - непустое множество, тогда множество-степень $P(A)$ является моделью булевой алгебры, если условиться о следующем.

Элементы этой булевой алгебры - это различные подмножества множества A .

Операция \wedge определяется как пересечение множеств, операция \vee обозначает объединение множеств и операция \neg является абсолютным дополнением (до A). Нетрудно убедиться, что все аксиомы булевой алгебры при таких определениях выполнены и множества A и \emptyset являются соответственно единичным и нулевым элементом алгебры.

Кроме конъюнкции и дизъюнкции особенно важны с точки зрения технической реализации переключательных функций следующие операции:

$f \downarrow g = \neg(f \vee g)$ (функция Пирса);
 $f \mid g = \neg(f \wedge g)$ (штрих Шеффера);
 $f \supset g = \neg f \vee g$ (импликация);
 $f \setminus g = f \wedge \neg g$ (разность, теоретико-множественная разность в $P(A)$);
 $f \sim g = (f \wedge g) \vee (\neg f \wedge \neg g)$ (эквивалентность);
 $f + g = (f \wedge \neg g) \vee (\neg f \wedge g)$ (симметрическая разность, исключаящее "или", неэквивалентность).

Для произвольного элемента f булевой алгебры имеем:

$\in \supset f = \varnothing$, $\varnothing \supset f = f$,
 $f \supset \varnothing = \varnothing$, $f \supset \in = \neg f$.

Кроме того, $f \setminus g = \neg(f \supset g)$.

Для симметрической разности имеем также

$$f + g = (f \setminus g) \vee (g \setminus f)$$

$$f + g = \neg(f \sim g)$$

$$\in + \in = \in \quad \in + \not\in = \not\in$$

$$\not\in + \in = \not\in \quad \not\in + \not\in = \in$$

При отождествлении \in с 0, а $\not\in$ с 1, получаем операцию сложения по модулю 2:

$$0+0=0 \quad 1+0=1$$

$$1+1=0 \quad 0+1=1$$

Возможно, теперь стало понятным, почему основные равносильности логики высказываний и основные тождества алгебры множеств так похожи - они просто законы булевой алгебры.

На основании законов булевой алгебры можно доказывать утверждения о тождественности (эквивалентности) выражений с булевыми операциями. Например, $(f \wedge g) \vee (\neg f \wedge b) \vee (f \wedge \neg g) = g \vee f$. Действительно,

$$\begin{aligned} (f \wedge g) \vee (\neg f \wedge b) \vee (f \wedge \neg g) &= \\ ((f \wedge g) \vee (\neg f \wedge g)) \vee (f \wedge \neg g) &= \\ ((f \vee \neg f) \wedge g) \vee (f \wedge \neg g) &= \\ (\top \wedge g) \vee (f \wedge \neg g) &= \\ g \vee (f \wedge \neg g) &= \\ (g \vee f) \wedge (g \vee \neg g) &= \\ (g \vee f) \wedge \top &= \\ g \vee f. \end{aligned}$$

Докажем, что все булевы выражения можно выразить через функцию Пирса. Для этого достаточно показать представимость с помощью функции Пирса базисных операций \wedge , \vee , \neg .

Имеем

$$\neg f = \neg(f \vee f) = f \downarrow f.$$

Далее, по определению,

$$f \downarrow g = \neg(f \vee g) \Rightarrow f \vee g = \neg(f \downarrow g) \Rightarrow f \vee g = (f \downarrow g) \downarrow (f \downarrow g).$$

И, наконец,

$$f \wedge g = \neg(\neg f \vee \neg g) = (\neg f) \downarrow (\neg g) = (f \downarrow f) \downarrow (g \downarrow g).$$

§2 Булевы функции. Теорема о нормальной булевой форме

Рассмотрим еще одну модель булевой алгебры.

Определение. Пусть M - произвольная булева алгебра с базисными операциями \wedge , \vee , \neg . Рассмотрим множество n -местных функций

$$f: M^n \rightarrow M$$

с поточечно определенными операциями \wedge , \vee и \neg . А именно, пусть

$$f : (x_1, x_2, \dots, x_n) \rightarrow f(x_1, x_2, \dots, x_n).$$

Тогда, по определению,

$$f_1 \wedge f_2 : (x_1, x_2, \dots, x_n) \rightarrow f_1(x_1, x_2, \dots, x_n) \wedge f_2(x_1, x_2, \dots, x_n),$$

$$f_1 \vee f_2 : (x_1, x_2, \dots, x_n) \rightarrow f_1(x_1, x_2, \dots, x_n) \vee f_2(x_1, x_2, \dots, x_n),$$

$$\neg f : (x_1, x_2, \dots, x_n) \rightarrow \neg f(x_1, x_2, \dots, x_n).$$

Множество таким образом определенных функций вместе с введенными операциями является булевой алгеброй и *называются булевыми функциями*.

Две постоянные функции

$$0 : (x_1, x_2, \dots, x_n) \rightarrow \in$$

$$1 : (x_1, x_2, \dots, x_n) \rightarrow \not\in$$

являются соответственно нулевым и единичным нейтральным элементом.

Определение.

Если булева алгебра M - двухэлементна (т. е. содержит только $\not\in$ и \in), то булевы функции называются *двоичными функциями*.

Если в двухэлементной булевой алгебре элементы $\not\in$ и \in интерпретировать как "включено" и "выключено", то двоичные функции называются *переключаемыми функциями*. При такой интерпретации $\not\in$ и \in обозначаются соответственно через 1 и 0.

Если $M = \{И, Л\}$ - булева алгебра значений истинности, то булевы функции являются функциями истинности или функциями логики высказываний.

Переключаемые функции одной переменной имеют вид

$$f: \{0,1\} \rightarrow \{0,1\},$$

и может быть только четыре различных одноместных переключаемых функций:

$$0: x \rightarrow 0;$$

$$1: x \rightarrow 1;$$

$$\text{id} : x \rightarrow x, \text{ тождественная функция};$$

$$\text{neg} : x \rightarrow \neg x, \text{ функция отрицания}.$$

Всякую переключаемую функцию от n переменных можно задать таблицей из 2^n строк, в которой в каждой строке записывают одну из оценок списка переменных, принимающих значение 0 или 1. Например, для $n=3$ переключаемую функцию можно задать табл. 9

Таблица 9

X_1	X_2	X_3	$f(X_1, X_2, X_3)$
1	1	1	$f(1, 1, 1)$
1	1	0	$f(1, 1, 0)$

1	0	1	$f(1,0,1)$
1	0	0	$f(1,0,0)$
0	1	1	$f(0,1,1)$
0	1	0	$f(0,1,0)$
0	0	1	$f(0,0,1)$
0	0	0	$f(0,0,0)$

Так как длина каждого столбца равна 2^n , а различных столбцов из 0 и 1 длины 2^n имеется $2^{(2^n)}$, то существует точно $2^{(2^n)}$ переключательных функций от n переменных. В частности, при $n=2$ имеем 16 различных переключательных функций.

Вопрос: Нельзя ли свести все переключательные функции к какому-нибудь меньшему числу "базисных" переключательных функций?

Ответ: это возможно. Например, можно все переключательные функции представить как композицию только трех функций:

(двуместная конъюнкция) $\wedge : (x_1, x_2) \rightarrow x_1 \wedge x_2$;

(двуместная дизъюнкция) $\vee : (x_1, x_2) \rightarrow x_1 \vee x_2$;

(одноместная функция отрицания) $\neg : x \rightarrow \neg x$.

Лемма 3.1. Для всякой n -местной переключательной функции f выполняется соотношение

$$f(a_1, a_2, \dots, a_{i-1}, a_i, a_{i+1}, \dots, a_n) = (a_i \wedge f(a_1, a_2, \dots, a_{i-1}, 1, a_{i+1}, \dots, a_n)) \vee (\neg a_i \wedge f(a_1, a_2, \dots, a_{i-1}, 0, a_{i+1}, \dots, a_n)).$$

Для доказательства рассмотрим два случая.

Пусть $a_i=1$, тогда $\neg a_i=0$. Правая часть доказываемого соотношения равна $(1 \wedge f(a_1, a_2, \dots, a_{i-1}, 1, a_{i+1}, \dots, a_n)) \vee (0 \wedge f(a_1, a_2, \dots, a_{i-1}, 0, a_{i+1}, \dots, a_n))$. Первый член в дизъюнкции равен $f(a_1, a_2, \dots, a_{i-1}, 1, a_{i+1}, \dots, a_n)$, а второй 0. Следовательно, правая часть равна $f(a_1, a_2, \dots, a_{i-1}, 1, a_{i+1}, \dots, a_n)$, но точно такое же значение имеет левая часть.

Пусть $a_i=0$. Совершенно аналогично получаем, что правая часть равна $f(a_1, a_2, \dots, a_{i-1}, 0, a_{i+1}, \dots, a_n)$.

Эта лемма позволяет "выносить" переменную a_i за знак переключательной функции. Последовательным применением леммы к a_1, a_2, \dots, a_n устанавливается

Теорема 3.1 (о булевой нормальной форме). Каждую переключательную функцию можно однозначно представить в следующей (дизъюнктивной) нормальной форме:

$$\begin{aligned} f(a_1, a_2, \dots, a_n) = & (a_1 \wedge a_2 \wedge \dots \wedge a_{n-1} \wedge a_n \wedge f(1, 1, \dots, 1, 1)) \\ & \vee (\neg a_1 \wedge a_2 \wedge \dots \wedge a_{n-1} \wedge a_n \wedge f(0, 1, \dots, 1, 1)) \\ & \vee (a_1 \wedge \neg a_2 \wedge \dots \wedge a_{n-1} \wedge a_n \wedge f(1, 0, \dots, 1, 1)) \\ & \dots \end{aligned}$$

$$\vee(\neg a_1 \wedge \neg a_2 \wedge \dots \wedge \neg a_{n-1} \wedge a_n \wedge f(0, 0, \dots, 0, 1)) \\ \vee(\neg a_1 \wedge \neg a_2 \wedge \dots \wedge \neg a_{n-1} \wedge \neg a_n \wedge f(0, 0, \dots, 0, 0)).$$

Если $f(a_1, a_2, \dots, a_{n-1}, a_n) = 0$, то соответствующий член, разумеется, выпадает из представления. Таким образом, всякая переключательная функция представима в виде дизъюнкции k , $0 \leq k \leq 2^k$, членов - так называемых *совершенных конъюнкций*, каждая совершенная конъюнкция - это n -местная конъюнкция, у которых все аргументы - либо сами переменные, либо их отрицания.

Пример 3.1. Переключательную функцию с таблицей значений

a_1	1	0	1	0	1	0	1	0
a_2	1	1	0	0	1	1	0	0
a_3	1	1	1	1	0	0	0	0
$f(a_1, a_2, a_3)$	1	0	0	1	0	1	1	0

можно представить в виде

$$f(a_1, a_2, a_3) = (a_1 \wedge a_2 \wedge a_3) \vee (\neg a_1 \wedge \neg a_2 \wedge a_3) \vee (\neg a_1 \wedge a_2 \wedge \neg a_3) \vee (a_1 \wedge \neg a_2 \wedge \neg a_3).$$

Всего имеется 16 двуместных переключательных функций. Они распадаются на следующие группы:

Функция без совершенных конъюнкций

$$f(x_1, x_2) = 0$$

Функция со всеми четырьмя совершенными конъюнкциями

$$f(x_1, x_2) = (x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2) \vee (\neg x_1 \wedge \neg x_2) = 1$$

Четыре функции по одной совершенной конъюнкции

$x_1 \wedge x_2$ - конъюнкция

$\neg x_1 \wedge \neg x_2$ - функция Пирса

$$x_1 \wedge \neg x_2 = x_1 \setminus x_2$$

$$\neg x_1 \wedge x_2 = x_2 \setminus x_1$$

Четыре функции по три совершенных конъюнкции

$$x_1 \vee x_2 = (x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2) - \text{дизъюнкция}$$

$$x_1 | x_2 = (x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2) \vee (\neg x_1 \wedge \neg x_2) - \text{штрих Шеффера}$$

$$x_1 \supset x_2 = (x_1 \wedge x_2) \vee (\neg x_1 \wedge x_2) \vee (\neg x_1 \wedge \neg x_2)$$

$$x_2 \supset x_1 = (x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge \neg x_2)$$

Шесть функций по две совершенных конъюнкции

$$x_1 \sim x_2 = (x_1 \wedge x_2) \vee (\neg x_1 \wedge \neg x_2) - \text{эквивалентность}$$

$$x_1 + x_2 = (x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2) - \text{симметрическая разность}$$

$$(x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2)$$

$$(\neg x_1 \wedge x_2) \vee (\neg x_1 \wedge \neg x_2)$$

$$(x_1 \wedge x_2) \vee (\neg x_1 \wedge x_2)$$

$$(x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge \neg x_2)$$

Вопрос о тождественности двух переключательных функций можно решить, приведя их обоих к совершенной дизъюнктивной нормальной форме или преобразуя булевы выражения по законам булевой алгебры.

§3 Полные системы булевых функций

Определение. Система функций $\{f_1, f_2, \dots, f_n\}$ называется *полной*, если любая булева функция может быть выражена через функции f_1, f_2, \dots, f_n с помощью композиции (т. е. составления сложных функций).

Теорема 3.2. Следующие системы булевых функций полны:

$\{\neg, \wedge, \vee\}$; $\{\neg, \vee\}$; $\{\neg, \wedge\}$; $\{\neg, \supset\}$; $\{+, \wedge, 1\}$; $\{\supset, \neg\}$; $\{\downarrow\}$; $\{\}$.

Доказательство. То, что система функций $\{\neg, \wedge, \vee\}$ является полной, доказано в теореме 3.1. Конъюнкцию можно выразить через операции \neg и \vee , а дизъюнкцию можно выразить через операции \neg и \wedge , поэтому полными системами являются $\{\neg, \vee\}$ и $\{\neg, \wedge\}$. Полнота системы $\{+, \wedge, 1\}$ следует из следующих тождеств: $\neg X = X + 1$, $X \vee Y = X \wedge Y + X + Y$. Для доказательства полноты системы $\{\supset, \neg\}$ воспользуемся равенством $f \supset g = \neg f \vee g$. Отсюда получаем $f \vee g = \neg f \supset g$, $f \wedge g = \neg f \vee \neg g = f \supset \neg g$. Полнота системы $\{\downarrow\}$ доказана в §1 данной главы. Полноту системы, состоящей из одного штриха Шеффера, предлагаем доказать читателю.

Рассмотрим систему $\{+, \wedge, 1\}$. Будем вместо \wedge писать знак умножения \square , или вообще опускать, т. е. вместо $X \wedge Y$ писать XY .

Таблица 10

X	Y	X+Y	XY
1	1	0	1
1	0	1	0
0	1	1	0
0	0	0	0

С помощью табл. 10 можно доказать тождества:

- 1) $X+Y = Y+X$, $XY = YX$;
- 2) $(X+Y)+Z = X+(Y+Z)$, $(XY)Z = X(YZ)$;
- 3) $X+X = 0$, $XX = X$;
- 4) $X(Y+Z) = XY + XZ$;
- 5) $0+X = X$;
- 6) $0X = 0$;
- 7) $1X = X$.

Все правила, за исключением 3, выражают свойства, аналогичные обычным свойствам арифметики сложения и умножения.

Поскольку система $\{+, \wedge, 1\}$ полная, то любую переключательную функцию можно представить в виде многочлена с единичными коэффици-

ентами и с переменными, входящими только в первой степени. Такие многочлены называются *многочленами Жегалкина*.

Пример 3.2.

$$X \vee Y \vee Z = XYZ + XY + XZ + YZ + X + Y + Z.$$

Переключательную функцию из примера 3.1 $f(a_1, a_2, a_3) = (a_1 \wedge a_2 \wedge a_3) \vee (\neg a_1 \wedge \neg a_2 \wedge a_3) \vee (\neg a_1 \wedge a_2 \wedge \neg a_3) \vee (a_1 \wedge \neg a_2 \wedge \neg a_3)$ можно представить в виде следующего многочлена Жегалкина: $a_1 a_2 + a_1 a_3 + a_1 + a_2 + a_3$.

§4 Переключательные элементы

Пусть имеется "черный ящик" - некоторое устройство, внутренняя структура которого нас не интересует, а известно лишь, что оно имеет n упорядоченных "входов" (например, занумерованных числами от 1 до n) и один "выход" (рис. 3).

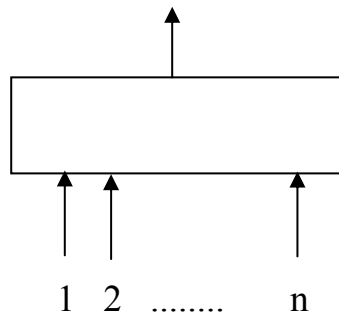


Рис. 3 "Черный ящик"

На каждый из входов могут подаваться два сигнала (например, отсутствие электрического тока или наличие его), которые мы условимся обозначать символами 0 и 1, и при каждом наборе сигналов на входах однозначно определяется сигнал на выходе. Такое устройство назовем *переключательным элементом*. Ясно, что каждому переключательному элементу соответствует переключательная функция $f(x_1, x_2, \dots, x_n)$, которая строится следующим образом: входу с номером i ($1 \leq i \leq n$) ставится в соответствие переменная x_i и каждому (двоичному) набору значений этих переменных отвечает величина $f(x_1, x_2, \dots, x_n)$, равная 0 или 1 в зависимости от того, какой сигнал возникает на выходе при подаче на входы переключательного элемента.

Если у нас имеется несколько переключательных элементов, то из них можно получать новые сложные переключательные элементы следующим образом. Один из входов одного переключательного элемента можно соединить с выходом другого переключательного элемента (рис. 4).

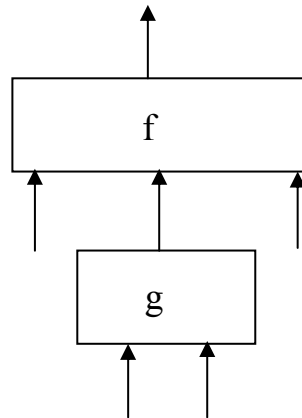


Рис. 4 Соединение "выход-вход"

Возникающее при этом устройство можно считать новым переключательным элементом, выходом которого является выход первого элемента (f), а входами все оставшиеся свободными входы первого элемента и входы второго элемента (g).

При таком соединении новому сложному переключательному элементу соответствует переключательная функция, полученная в результате суперпозиции исходных функций. Так, например, для элемента на рис. 4 имеем

$$h(x_1, y_1, y_2, x_3) = f(x_1, g(y_1, y_2), x_3).$$

Кроме этой операции можно отождествлять входы функционального элемента (рис. 5). При этом возникает новый переключательный элемент, у которого тот же выход и те же входы, за исключением отождествленных, которые теперь считаются одним входом. Для этого элемента соответствующая переключательная функция строится из первоначальной:

$$h(x_1, x_2, x_3) = f(x_1, x_2, x_2, x_3).$$

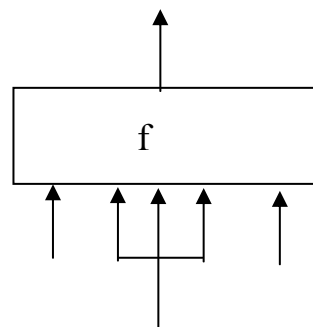


Рис. 5 Соединение "отождествление входов"

Можно использовать несколько соединений указанных выше двух типов. В этом случае получается сложный переключательный элемент или может получиться более сложное соединение, когда отождествляются некоторые входы различных переключательных элементов (рис. 6). В последнем случае такое соединение называется *переключательной (комбинационной схемой)*.

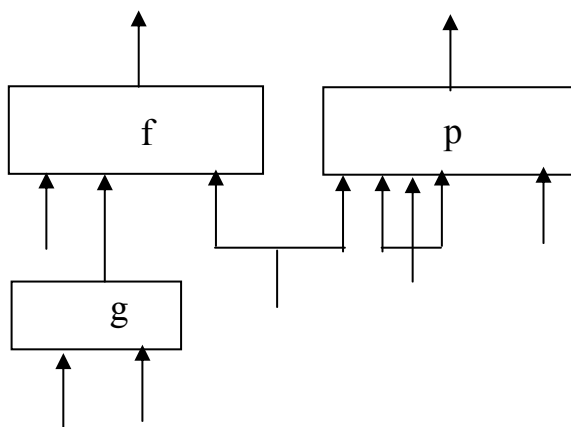


Рис. 6 Комбинационная схема

Комбинационным схемам соответствует уже несколько переключательных функций (своя для каждого внешнего выхода), составленных с помощью суперпозиции. Для рис.6 получаем две переключательные функции $\{h, t\}$:

$$h(x_1, x_2, x_3, x_4, x_5, x_6) = f(x_1, g(x_2, x_3), x_4),$$

$$t(x_1, x_2, x_3, x_4, x_5, x_6) = p(x_4, x_5, x_5, x_5, x_6).$$

Комбинационная схема изображается уже черным ящиком с несколькими выходами (рис. 7)

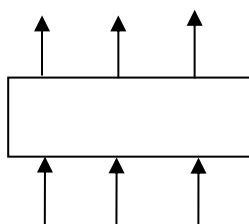


Рис. 7 Комбинационная схема

Символические изображения переключательных функций

Для трех переключательных функций: отрицания, конъюнкции и дизъюнкции - вместо прямоугольника используют три особых символических изображения (рис.8). Соответствующие переключательные элементы называются *НЕ-элементом*, *И-элементом* и *ИЛИ-элементом*; о них говорят как о *логических элементах*.

Символическое изображение			
---------------------------	--	--	--

Формула	$c = a \wedge b$	$c = a \vee b$	$c = \neg a$
Название	И-элемент	ИЛИ-элемент	НЕ-элемент

Рис.8. Символические изображения для конъюнкции, дизъюнкции и отрицания

Применение основных операций к переключательным функциям реализуется посредством соответствующего соединения символических изображений. Таким образом, возникают комбинационные схемы (в том числе и со многими выходами), построенные лишь из конъюнкции, дизъюнкции и отрицания. Вместо отрицания на входе конъюнкции или дизъюнкции ставят просто жирную "точку отрицания"; это дает такие картинки, как, например, на рис. 10.

Закон ассоциативности отражает возможность снабжать символические изображения логических элементов более чем двумя входами (рис. 9).

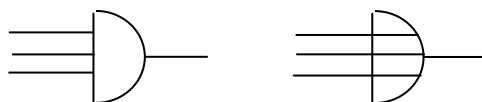


Рис.9. И- и ИЛИ-элементы со многими входами

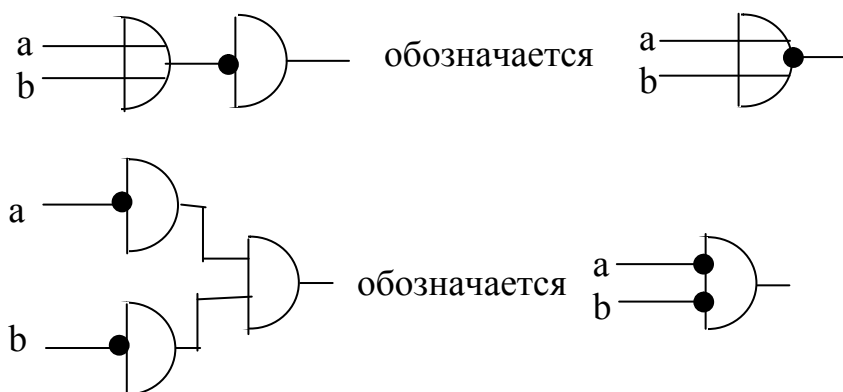


Рис.10. Символическое изображение функции Пирса: $\neg(a \vee b) = \neg a \wedge \neg b$

Пример: полусумматор

Для поразрядного сложения двух двоичных закодированных чисел применяется комбинационная схема, называемая полусумматором, с двумя перестановочными входами a , b и двумя выходами s и u

$$s = a + b, u = a \wedge b.$$

На рис. 11 представлены две реализации полусумматора, для правой реализации использовано преобразование $a + b = \neg((a \wedge b) \vee \neg(a \vee b))$.

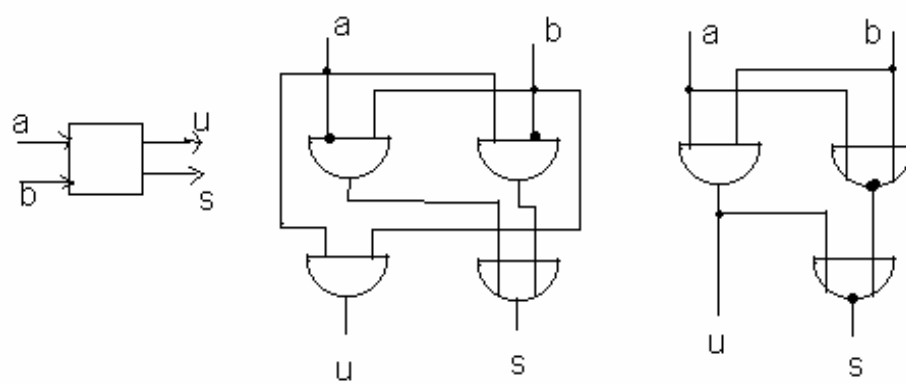


Рис. 11. Реализации полусумматора

Не то, что мните вы, природа:
 Не слепок, не бездушный лик -
 В ней есть душа, в ней есть свобода,
 В ней есть любовь, в ней есть язык...

Ф. И. Тютчев

Глава 4. ЛОГИКА ПРЕДИКАТОВ

§1 Формулы логики предикатов

Существуют такие виды логических рассуждений, которые нельзя формализовать на языке логики высказываний. Вот примеры таких рассуждений:

1. Каждый любит сам себя. Значит, кого-то кто-нибудь любит.
2. Ни одно животное не бессмертно. Собаки - животные. Значит некоторые собаки не бессмертны.
3. Всякий, кто может решить эту задачу, - математик. Ни один математик не может решить эту задачу. Значит, она неразрешима.
4. Перья есть только у птиц. Ни одно млекопитающее не является птицей. Значит, все млекопитающие лишены перьев.

Чтобы выяснить корректность этих рассуждений недостаточно установить только истинностно-функциональные отношения между входящими в них предложениями, но необходимо также исследовать внутреннюю структуру предложений, а также понять такие слова как "каждый", "всякий" и "некоторые".

В этих приведенных выше предложениях в каждом рассуждении рассматриваются объекты (сущности) из различных предметных областях. Эти объекты обладают различными свойствами и связаны друг с другом различными отношениями. Удобно ввести специальные обозначения. Во-первых, введем специальные переменные, значениями которых будут объекты из соответствующих предметных областей. Обозначим такие переменные символами x и y . Во-вторых, свойства объектов и бинарные отношения между объектами будем обозначать, например, $P(x)$ и $Q(x, y)$, соответственно. И, в-третьих, фразы вида "все x обладают свойством P " и "некоторые x обладают свойством P " будем обозначать символически $\forall x P(x)$ и $\exists x P(x)$, соответственно.

Используя приведенные обозначения, можно переписать предыдущие рассуждения в таком виде, чтобы лучше увидеть его структуру.

1. Если $\forall x Q(x, x)$, то $\forall x \exists y Q(x, y)$. В данном случае $Q(x, y)$ обозначает бинарное отношение "x любит y".

2. Если $\forall x (A(x) \supset \neg M(x))$ и $\forall x (C(x) \supset A(x))$, то $\exists x (C(x) \supset \neg M(x))$. В данном случае $A(x)$ обозначает свойство "x - животное", $M(x)$ обозначает свойство "x - бессмертно" и $C(x)$ обозначает свойство "x - собака".

3. Если $\forall x(S(x) \supset M(x))$ и $\forall x(M(x) \supset \neg S(x))$, то $\neg \exists x S(x)$. Здесь $S(x)$ обозначает свойство "х может решить задачу" и $M(x)$ обозначает свойство "х - математик".

4. Если $\forall x(P(x) \supset B(x))$ и $\forall x(M(x) \supset \neg B(x))$, то $\forall x(M(x) \supset \neg P(x))$. Здесь $P(x)$ обозначает свойство "х имеет перья", $B(x)$ обозначает свойство "х - птица" и $M(x)$ обозначает свойство "х - млекопитающее".

Введенные обозначения принадлежат языку логики предикатов. Рассмотрим предложения, зависящие от параметров:

"х - четное число";

"х меньше у";

" $x + y = z$ ";

"х - отец Иванова Олега";

"х и у - братья";

"х - является смертным".

Все эти предложения становятся истинными или ложными (т. е. становятся высказываниями) при замене предметных переменных на соответствующие предметные константы.

Определение.

Предикатом $P(x_1, x_2, \dots, x_n)$ называется функция

$P : M^n \rightarrow \{И, Л\}$.

Множество M определяется контекстом, элементы этого множества *предметные (индивидные) константы*.

Предикат от n аргументов называется *n-местным* или *n-арным*. Любое высказывание можно рассматривать как нульместный предикат (постоянная функция).

Некоторые примеры предикатов мы уже видели:

$Q(x, y)$ - бинарное отношение "х любит у";

$M(x)$ - "х - является математиком";

$P(x)$ - "х имеет перья".

Над предикатами можно проводить обычные логические операции и, тем самым, создавать новые предикаты.

Пример 4.1.

1. $P(x) \equiv$ "х делится на 2",

$Q(x) \equiv$ "х делится на 3",

$P(x) \& Q(x) \equiv$ "х делится на 6".

2. $S(x, y) \equiv$ " $x = y$ ",

$\neg S(x, x) \supset S(x, y)$ - предикат истинен при любых x и y .

Наряду с применением логических связок с предикатами можно выполнять и операции квантификации или "связывания квантором переменную".

Определение. Квантор общности. Пусть $P(x)$ - предикат, тогда формула $\forall x P(x)$ обозначает высказывание, которое истинно тогда и только тогда, когда для всех $x \in M$ предикат $P(x)$ - истинен. $\forall x P(x)$ читается как

"для всех x $P(x)$ ". Формула $\forall x P(x)$ от x не зависит - вместо x нельзя подставить никакую предметную константу. Символ \forall называется *квантором общности* или *универсальным квантором*.

Формула $\forall x P(x)$ на обычном языке передается также и следующими способами:

$P(x)$ при произвольном x ;
 $P(x)$, каково бы ни было x ;
 для каждого x (верно) $P(x)$;
 всегда имеет место $P(x)$;
 каждый обладает свойством P ;
 все удовлетворяет P .

Определение. Квантор существования. Пусть $P(x)$ - предикат, тогда формула $\exists x P(x)$ обозначает высказывание, которое истинно тогда и только тогда, когда найдется такой $x \in M$, что предикат $P(x)$ - истинен. $\exists x P(x)$ читается как "существует такой x , что $P(x)$ ". Формула $\exists x P(x)$ от x не зависит - вместо x нельзя подставить никакую предметную константу. Символ \exists называется *квантором существования* или *экзистенциальным квантором*.

Формула $\exists x P(x)$ на обычном языке передается также и следующими способами:

для некоторых x (имеет место) $P(x)$;
 для подходящего x (верно) $P(x)$;
 имеется x , для которого $P(x)$;
 у некоторых вещей есть признак P ;
 кто-нибудь относится к (есть) P .

Пример 4.2.

$\exists x (P(x) \& Q(x)) \equiv$ "существует x , который делится на 6" - истинное высказывание.

$\forall x (P(x) \& Q(x)) \equiv$ "все x делятся на 6" - ложное высказывание.

Определение. Алфавит языка логики предикатов содержит:

- 1) символы предметных переменных x, y, z, x_1, y_1, z_1 и т. д.;
- 2) символы предикатов P, Q, P_1, Q_1 и т. д.;
- 3) символы логических операций $\&, \vee, \neg, \supset, \sim$;
- 4) символы кванторов \forall, \exists .

Определение. Слово в алфавите логики предикатов называется *формулой*, если выполнены следующие условия (одновременно определяют понятия *свободной* и *связанной переменной* формулы).

1. $P(x_1, x_2, \dots, x_n)$ - формула, если P - n -местный предикат (Такая формула называется атомарной). Все переменные x_1, x_2, \dots, x_n - свободные переменные, связанных переменных в этой формуле нет.

2. Пусть A - формула, тогда $\neg A$ - формула с теми же свободными и связанными переменными, что и в формуле A .

3. Пусть A и B - формулы, причем нет таких переменных, которые были бы связаны в одной формуле и свободные - в другой. Тогда $(A \& B)$, $(A \vee B)$, $(A \supset B)$, $(A \sim B)$ суть формулы, в которых свободные переменные формул A и B остаются свободными, а связанные переменные формул A и B остаются связанными.

4. Пусть A - формула, содержащая свободную переменную x . Тогда $\forall x A$, $\exists x A$ тоже формулы. Переменная x в них связана. Остальные же переменные, которые в формуле A свободны, остаются свободными. Переменные, которые в формуле A были связаны, остаются связанными. В этих вновь полученных формулах формула A называется *областью действия квантора* $\forall x$ и $\exists x$ соответственно.

5. Слово в алфавите логики предикатов называется формулой только в том случае, если это следует из правил 1-4.

Пример 4.3.

Следующие выражения являются формулами логики предикатов: $R(x_1, x_2, x_7)$ - атомарная формула, в которой x_1, x_2, x_7 - свободные переменные; $(\forall x \exists y R(x, y, z)) \supset \forall x Q(x, w)$ - формула, в которой переменные x, y связаны, а переменные z, w свободны.

Выражение $(\forall x \exists y R(x, y, z)) \supset \forall x Q(x, y)$ не является формулой.

Факты не существуют - есть только интерпретации.

Фридрих Ницше

§2 Интерпретации

Формулы имеют смысл только тогда, когда имеется какая-нибудь интерпретация входящих в нее символов.

Определение. Под *интерпретацией* мы будем понимать всякую пару, состоящую из непустого множества M , называемого *областью интерпретации*, и какого-либо отображения, относящему каждому предикатному символу арности n некоторое n -местное отношение на M . При заданной интерпретации предметные переменные мыслятся пробегающими область M этой интерпретации, а связкам и кванторам придается их обычный смысл.

Для данной интерпретации всякая формула без свободных переменных представляет собой высказывание, которое истинно или ложно, а всякая формула со свободными переменными выражает некоторое отношение на области интерпретации; это отношение может быть выполнено (истинно) для одних значений переменных из области интерпретации и не выполнено (ложно) для других.

Дадим индуктивное определение *значения формулы* (в данной интерпретации).

Значение формулы F на n -ке $\langle a_1, a_2, \dots, a_n \rangle$, $a_i \in M$, своих свободных переменных $\langle x_1, x_2, \dots, x_n \rangle$ будем обозначать $F|\langle a_1, a_2, \dots, a_n \rangle$.

Формула F есть атомарная переменная $A(x_1, x_2, \dots, x_n)$, где x_1, x_2, \dots, x_n - свободные переменные, тогда $F|\langle a_1, a_2, \dots, a_n \rangle$ есть значение n -местного предиката, сопоставленного предикатному символу A , при соответствующем замещении его переменных элементами a_1, a_2, \dots, a_n .

Формула F имеет вид $\neg A$. Пусть $A|\langle a_1, a_2, \dots, a_n \rangle = \varepsilon$, тогда $F|\langle a_1, a_2, \dots, a_n \rangle = \neg \varepsilon$.

Формула F имеет вид $A \vee B$, $A \& B$, $A \supset B$ или $A \sim B$. Пусть $A|\langle a_1, a_2, \dots, a_n \rangle = \alpha$, $B|\langle a_1, a_2, \dots, a_n \rangle = \beta$, тогда $F|\langle a_1, a_2, \dots, a_n \rangle$ равно соответственно $\alpha \vee \beta$, $\alpha \& \beta$, $\alpha \supset \beta$, $\alpha \sim \beta$.

Формула F имеет вид $\forall x A$. Если x_1, x_2, \dots, x_n - все свободные переменные формулы F , то x, x_1, x_2, \dots, x_n - все свободные переменные формулы A . Значение $\forall x A|\langle a_1, a_2, \dots, a_n \rangle = I$ тогда и только тогда, когда для любого $a \in M$ имеем $A|\langle a, a_1, a_2, \dots, a_n \rangle = I$.

Формула F имеет вид $\exists x A$. Если x_1, x_2, \dots, x_n - все свободные переменные формулы F , то x, x_1, x_2, \dots, x_n - все свободные переменные формулы A . Значение $\exists x A|\langle a_1, a_2, \dots, a_n \rangle = I$ тогда и только тогда, когда существует такое $a \in M$, что $A|\langle a, a_1, a_2, \dots, a_n \rangle = I$.

Пример 4.4. Рассмотрим три формулы:

- 1) $A(x, y)$;
- 2) $\forall y A(x, y)$;
- 3) $\exists x \forall y A(x, y)$.

Возьмем в качестве области интерпретации множество целых положительных чисел и интерпретируем $A(x, y)$ как $x \leq y$. Тогда первая формула - это предикат $x \leq y$, который принимает значение I для всех пар a, b целых положительных чисел таких, что $a \leq b$. Вторая формула выражает свойство: "для каждого целого положительного числа y $x \leq y$ ", которое выполняется только при $x=1$. Наконец, третья формула - это истинное высказывание о существовании наименьшего целого положительного числа. Если бы в качестве области интерпретации мы рассматривали множество целых чисел, то третья формула была бы ложным высказыванием.

Пример 4.5. Пусть задана следующая интерпретация. M - множество натуральных чисел $(0, 1, 2, \dots)$, предикатные символы $S(x, y, z)$ и $P(x, y, z)$ обозначают следующие предикаты " $x + y = z$ " и " $x \times y = z$ " соответственно.

Запишем формулы, истинные на M тогда и только тогда, когда выполнены следующие условия:

- 1) $x = 0$;
- 2) $x = 1$;
- 3) x - четное число;
- 4) x - простое число;
- 5) $x = y$;

- 6) $x \leq y$;
- 7) x делит y ;
- 8) коммутативность сложения.

Ответы:

- 1) $F_1(x) = \forall y S(x, y, y)$, так как $x+y=y$ для любого y тогда и только тогда, когда $x = 0$;
- 2) $F_2(x) = \forall y P(x, y, y)$;
- 3) $F_3(x) = \exists y S(y, y, x)$;
- 4) $F_4(x) = \neg F_1(x) \& \neg F_2(x) \& (\forall y \forall z (P(y, z, x) \supset (F_2(y) \vee F_2(z))))$, где F_1, F_2 - формулы, определенные в пп. 1 и 2;
- 5) $F_5(x, y) = \forall z \forall u (S(x, z, u) \supset S(y, z, u))$;
- 6) $F_6(x, y) = \exists z S(x, z, y)$;
- 7) $F_7(x, y) = \exists z P(x, z, y)$;
- 8) $\forall x \forall y \forall z (S(x, y, z) \supset S(y, x, z))$.

Пример 4.6. Пусть $f(x)$ - произвольная фиксированная функция, заданная на отрезке $[a, b]$.

1. Рассмотрим интерпретацию: M - множество действительных чисел, $P(x, \delta)$ обозначает $|x - x_0| < \delta$, $Q(x, \varepsilon)$ обозначает $|f(x) - A| < \varepsilon$, $R(\varepsilon)$ обозначает $\varepsilon > 0$. Здесь x_0 - фиксированный элемент отрезка $[a, b]$; A - некоторое фиксированное действительное число. Тогда утверждение о том, что A - предел функции $f(x)$ при $x \rightarrow x_0$, записывается формулой

$$\forall \varepsilon \exists \delta \forall x ((R(\varepsilon) \& P(x, \delta)) \supset Q(x, \varepsilon)).$$

2. Рассмотрим интерпретацию: M - множество действительных чисел, $P(x, \delta)$ обозначает $|x - x_0| < \delta$, $S(x, \varepsilon)$ обозначает $|f(x) - f(x_0)| < \varepsilon$, $R(\varepsilon)$ обозначает $\varepsilon > 0$. Здесь x_0 - фиксированный элемент отрезка $[a, b]$. Тогда утверждение о том, что функция f - непрерывна в точке x_0 записывается в виде формулы

$$\forall \varepsilon \exists \delta \forall x ((R(\varepsilon) \& P(x, \delta)) \supset S(x, \varepsilon)).$$

3. Рассмотрим интерпретацию: M - множество действительных чисел, $P(x, x_1, \delta)$ обозначает $|x - x_1| < \delta$, $S(x, x_1, \varepsilon)$ обозначает $|f(x) - f(x_1)| < \varepsilon$, $R(\varepsilon)$ обозначает $\varepsilon > 0$, $D(x)$ обозначает $x \in [a, b]$. Тогда утверждение о том, что функция f - непрерывна на отрезке $[a, b]$ записывается в виде формулы

$$\forall x_1 \forall \varepsilon \exists \delta \forall x ((D(x_1) \& R(\varepsilon) \& P(x, x_1, \delta)) \supset S(x, x_1, \varepsilon)).$$

§3 Выполнимость и общезначимость

Определение. Формула A выполнима в данной интерпретации, если существует такой набор $\langle a_1, a_2, \dots, a_n \rangle$, $a_i \in M$, значений свободных переменных x_1, x_2, \dots, x_n формулы A , что $A|_{\langle a_1, a_2, \dots, a_n \rangle} = И$.

Формула A истинна в данной интерпретации, если она принимает значение И на любом наборе значений своих свободных переменных.

Формула A называется *ложной в данной интерпретации*, если она не является выполнимой ни на одном наборе значений своих свободных переменных.

Формула A *общезначима* (в логике предикатов), если она истинна в любой интерпретации.

Формула A *выполнима* (в логике предикатов), если существует интерпретация, в которой она выполнима.

Формула A называется *противоречием* (в логике предикатов), если она ложна в любой интерпретации.

Данная интерпретация называется *моделью* для данного множества формул S , если каждая формула из S истинна в данной интерпретации.

Приведенные ниже утверждения являются просто простыми следствиями определений.

Формула A является ложной в данной интерпретации тогда и только тогда, когда $\neg A$ истинно в той же интерпретации, и A истинно тогда и только тогда, когда $\neg A$ ложно.

Никакая формула не может быть одновременно истинной и ложной в одной и той же интерпретации.

Если в данной интерпретации истинны A и $A \supset B$, то истинно и B .

Формула $A \supset B$ является ложной в данной интерпретации тогда и только тогда, когда A в этой интерпретации истинно, а B ложно.

Формула A общезначима тогда и только тогда, когда формула $\neg A$ не является выполнимой, а формула A выполнима тогда и только тогда, когда формула $\neg A$ не является общезначимой.

Теорема 4.1. Формула $\forall x A(x) \supset A(y)$, где y не входит в формулу $A(x)$, общезначима.

Доказательство. Пусть $\{x, x_1, x_2, \dots, x_n\}$ - множество всех свободных переменных формулы A . Тогда, очевидно, $\{y, x_1, x_2, \dots, x_n\}$ будет множеством всех свободных переменных формулы $\forall x A(x) \supset A(y)$. Возьмем произвольную интерпретацию этой формулы на произвольном непустом множестве M и пусть $\langle b, a_1, a_2, \dots, a_n \rangle$ произвольный набор значений переменных $\{y, x_1, x_2, \dots, x_n\}$ на множестве M . Требуется доказать, что значение формулы $\forall x A(x) \supset A(y)$ на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$ есть истина.

Рассмотрим два случая.

1. Значение формулы $A(x)$ на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$ есть истина для любого $b \in M$. Тогда формула $\forall x A(x)$ имеет истинное значение на наборе $\langle a_1, a_2, \dots, a_n \rangle$. Формула $A(y)$, как и формула $A(x)$ является истинной, когда ее свободные переменные принимают значения $\langle b, a_1, a_2, \dots, a_n \rangle$. Отсюда получаем, что формула $\forall x A(x) \supset A(y)$ имеет истинное значение на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$.

2. Значение формулы $A(x)$ на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$ есть ложь. Следовательно, формула $\forall x A(x)$ имеет ложное значение на наборе

$\langle a_1, a_2, \dots, a_n \rangle$. Формула $A(y)$, как и формула $A(x)$ является ложной, когда ее свободные переменные принимают значения $\langle b, a_1, a_2, \dots, a_n \rangle$. Отсюда получаем, что формула $\forall x A(x) \supset A(y)$ имеет истинное значение на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$.

В обоих случаях значение формулы $\forall x A(x) \supset A(y)$ является истинным, чем и заканчивается доказательство.

Теорема 4.2. Формула $A(y) \supset \exists x A(x)$, где y не входит в формулу $A(x)$, общезначима.

Доказательство. Пусть $\{x, x_1, x_2, \dots, x_n\}$ - множество всех свободных переменных формулы A . Тогда, очевидно, $\{y, x_1, x_2, \dots, x_n\}$ будет множеством всех свободных переменных формулы $A(y) \supset \exists x A(x)$. Возьмем произвольную интерпретацию этой формулы на произвольном непустом множестве M и пусть $\langle b, a_1, a_2, \dots, a_n \rangle$ произвольный набор значений переменных $\{y, x_1, x_2, \dots, x_n\}$ на множестве M . Требуется доказать, что значение формулы $A(y) \supset \exists x A(x)$ на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$ есть истина.

Рассмотрим два случая.

1. Значение формулы $A(x)$ на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$ есть ложь. Формула $A(y)$, как и формула $A(x)$ является ложной, когда ее свободные переменные принимают значения $\langle b, a_1, a_2, \dots, a_n \rangle$. Отсюда получаем, что формула $A(y) \supset \exists x A(x)$ имеет истинное значение на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$.

2. Значение формулы $A(x)$ на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$ есть истина. Следовательно, формула $A(y)$, как и формула $A(x)$ является истинной на этом наборе. Формула $\exists x A(x)$ имеет истинное значение на наборе $\langle a_1, a_2, \dots, a_n \rangle$. Отсюда получаем, что формула $A(y) \supset \exists x A(x)$ имеет истинное значение на наборе $\langle b, a_1, a_2, \dots, a_n \rangle$.

В обоих случаях значение формулы $A(y) \supset \exists x A(x)$ является истинным, чем и заканчивается доказательство.

Задача распознавания общезначимости формул логики предикатов существенно сложнее, чем формул логики высказываний. Так же, как и в логике высказываний, она называется *проблемой разрешимости* и ставится следующим образом: указать алгоритм распознавания общезначимости формул (т. е. является ли данная формула общезначимой или нет). Как мы увидим в главе 5, такой алгоритм отсутствует.

Знания забудутся, пробелы в них -
никогда.

Михаил Генин

Два мира есть у человека:
Один, который нас творил,
Другой, который мы от века
Творим по мере наших сил.

Н. Заболоцкий

Глава 5. ИСЧИСЛЕНИЯ

Как было сказано выше, в логике предикатов, в отличие от логики высказываний, нет эффективного способа для распознавания общезначимости формул. Выходом из этой неприятной ситуации является следующий подход: выделяем небольшое множество общезначимых формул и указываем правила, с помощью которых из известных общезначимых формул создаются новые общезначимые формулы. Если окажется, что таким образом мы можем получить любую общезначимую формулу, то это нас вполне устроит.

Указанный путь реализуется в рамках так называемых формальных аксиоматических теорий.

§1 Формальные аксиоматические теории

Формальная теория представляет собой множество чисто абстрактных объектов (не связанных с внешним миром), в которой представлены правила оперирования множеством символов в чисто синтаксической трактовке без учета смыслового содержания (или семантики).

Формальную теорию иногда называют *аксиоматикой* или *формальной аксиоматической теорией*. Родоначальником аксиоматических теорий можно считать "Начала" Евклида.

Определение. *Формальная теория* T считается определенной, если:



Евклид

1) задано некоторое счетное множество символов - символов теории T ; конечные последовательности символов теории T называются *выражениями* теории T ;

2) имеется подмножество выражений теории T , называемых *формулами* теории T ;

3) выделено некоторое множество формул, называемых *аксиомами* теории T ;

4) имеется конечное множество R_1, R_2, \dots, R_m отношений между формулами, называемых *правилами вывода*. Правила вывода позволяют получать из некоторого конечного множества формул другое множество формул.

Обычно для формальной теории имеется алгоритм, позволяющий по данному выражению определить, является ли оно формулой. Точно также, чаще всего существует алгоритм, выясняющий, является ли данная формула теории T аксиомой; в таком случае T называется *эффективно аксиоматизированной* теорией.

Определение. Если формула A и формулы A_1, A_2, \dots, A_j находятся в некотором отношении R_i , то A называют *непосредственным следствием* из формул A_1, A_2, \dots, A_j , полученных по правилу R_i .

Выводом в теории T называется всякая последовательность A_1, A_2, \dots, A_n формул такая, что для любого i формула A_i есть либо аксиома теории T , либо непосредственное следствие каких-либо предыдущих формул по одному из правил вывода.

Формула A называется *теоремой* теории T , если в ней существует вывод, в котором последней формулой является A . Этот вывод называется выводом (доказательством) формулы A . Иными словами, теоремы аксиоматической теории - это формулы, которые могут выведены (доказаны) по определенным правилам.

Отметим, что в соотношении $\{\text{теоремы}\} \subset \{\text{формулы}\} \subset \{\text{выражения}\}$ включение множеств является строгим.

Определение. Формула A называется *следствием* множества формул Γ в теории T тогда и только тогда, когда существует такая последовательность формул A_1, A_2, \dots, A_n , что A_n есть A , и для любого i формула A_i есть либо аксиома, либо элемент Γ , либо непосредственное следствие некоторых предыдущих формул по одному из правил вывода. Такая последовательность называется *выводом A из Γ* . Элементы Γ называются *гипотезами* или *посылками вывода*.

Для сокращения утверждения " A есть следствие Γ " употребляется запись $\Gamma \vdash A$. Если множество Γ конечно: $\Gamma = \{B_1, B_2, \dots, B_n\}$, то вместо $\{B_1, B_2, \dots, B_n\} \vdash A$ пишут $B_1, B_2, \dots, B_n \vdash A$. Если Γ есть пустое множество \emptyset , то $\Gamma \vdash A$ имеет место тогда и только тогда, когда A является теоремой и в этом случае используют сокращенную запись $\vdash A$ (" A есть теорема").

Приведем несколько простых свойств понятия выводимости из посылок.

1. Если $\Gamma \subseteq \Sigma$ и $\Gamma \vdash A$, то $\Sigma \vdash A$.

Это свойство выражает тот факт, что если A выводимо из множества посылок Γ , то оно остается выводимым, если мы добавим к Γ новые посылки.

2. $\Gamma \vdash A$ тогда и только тогда, когда в Γ существует конечное подмножество Σ , для которого $\Sigma \vdash A$.

Часть "тогда" утверждения 2 вытекает из утверждения 1. Часть "только тогда" этого утверждения очевидно, поскольку всякий вывод A из Γ использует лишь конечное число посылок из Γ .

3. Если $\Sigma \vdash A$ и $\Gamma \vdash B$ для любого B из множества Σ , то $\Gamma \vdash A$.

Смысл этого утверждения прост: если A выводимо из Σ и любая формула из Σ выводима из Γ , то A выводима из Γ .

Первым вопросом, который возникает при задании формальной теории, является вопрос о том, возможно ли, рассматривая какую-нибудь формулу формальной теории, определить является ли она доказуемой или нет. Другими словами, речь идет о том, чтобы определить, является ли данная формула теоремой или *не-теоремой* и как это доказать. В математике предполагается, что при задании формальной теории существует алгоритм, который позволит получить ответ на данный вопрос. Такой алгоритм, если он существует, называется *процедурой решения*, а соответствующую теорию называют *разрешимой*. Однако даже такие простые и фундаментальные теории, как исчисление предикатов (см. §3), являются неразрешимыми. Причина этого состоит в следующем. Даже если применить правила словообразования (т. е. правила построения формул, правила вывода) последовательно ко всем возможным объектам формальной теории и формальная теория такова, что имеется принципиальная возможность полного перечисления ее теорем (даже при бесконечном их числе), то все же *не существует никакого подходящего способа в общем случае, чтобы перечислить все не-теоремы*.

Формальные теории являются не просто игрой ума, а всегда представляют собой модель какой-то реальности (либо конкретной, либо математической). Вначале математик изучает реальность, конструируя некоторое абстрактное представление о ней, т. е. некоторую формальную теорию. Затем он доказывает теоремы этой формальной теории. Вся польза и удобство формальных теорий как раз и заключается в их абстрагировании от конкретной реальности. Благодаря этому одна и та же формальная теория может служить моделью многочисленных различных конкретных ситуаций. Наконец, он возвращается к исходной точке всего построения и дает интерпретацию теорем, полученных при формализации.

§2 Исчисление высказываний

Оказывается множество тавтологий логики высказываний можно описать в рамках простой формальной аксиоматической теории - исчисления высказываний.

Определим исчисление высказываний следующим образом:

1. Символы исчисления высказываний: \neg , \supset , $($, $)$ и буквы X_i с целыми положительными числами в качестве индексов: X_1, X_2, \dots . Символы \neg и \supset - логические символы, символы X_1, X_2, \dots - переменные.

2. Формулы исчисления высказываний: а) все переменные X_i - формулы; б) если A и B - формулы, то $(\neg A)$ и $(A \supset B)$ тоже формулы.

Пример 5.1. Последовательность символов $\neg X_1 \supset X_2 \supset X_1$ - выражение, но не формула.

Пример 5.2. Пусть A, B, C - формулы. Тогда $(C \supset (A \supset B))$, $((\neg A) \supset B) \supset (\neg C)$ тоже формулы.

Для сокращения записи опустим в формуле внешние скобки и те пары скобок, без которых можно восстановить формулы по следующему правилу: каждое вхождение знака \neg относится к наикратчайшей подформуле, следующей за этим знаком. Тогда две предыдущие формулы примут вид

$$C \supset (A \supset B), (\neg A \supset B) \supset \neg C.$$

Аксиомы исчисления высказываний. Каковы бы ни были формулы A, B и C , следующие формулы являются аксиомами:

A1. $A \supset (B \supset A)$;

A2. $(A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))$;

A3. $(\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B)$.

Выражения A1-A3 называются схемами аксиом, поскольку каждое из них порождает бесконечное множество формул, являющихся аксиомами исчисления высказываний. Например, формула $X_1 \supset (X_2 \supset X_1)$ есть аксиома, полученная по схеме A1, формула $(\neg A \supset \neg A) \supset ((\neg A \supset A) \supset A)$ (где A - любая формула) - аксиома полученная по схеме A3.

Единственным правилом вывода формулы служит правило *modus ponens* (правило отделения, утверждающий модус). Пусть имеется три формулы: $A, A \supset B$ и B . Про формулу B будем говорить, что она получается по правилу *modus ponens* из формул A и $A \supset B$. Формально, это правило вывода записывается в виде: $A, A \supset B \Rightarrow B$.

Хотя для исчисления высказываний мы выбрали только два логических символа \neg и \supset , с помощью подходящих определений можно ввести и остальные операции \vee , $\&$, \sim , например, $A \sim B$ означает $\neg((B \supset A) \supset \neg(A \supset B))$.

Пример 5.3. Для любой формулы A построим вывод формулы $A \supset A$, т. е. $A \supset A$ - теорема.

Подставляем в схему аксиом A2 вместо В формулу $A \supset A$ и вместо С формулу А, получаем аксиому

$$(A \supset ((A \supset A) \supset A)) \supset ((A \supset (A \supset A)) \supset (A \supset A)). \quad (1)$$

Подставляем в A1 вместо формулы В формулу $A \supset A$, получаем аксиому $A \supset ((A \supset A) \supset A)$. (2)

Из формул (1) и (2) по правилу *modus ponens* получаем

$$(A \supset (A \supset A)) \supset (A \supset A). \quad (3)$$

Подставляем в A1 вместо формулы В формулу А, получаем аксиому $A \supset (A \supset A)$. (4)

Из формул (3) и (4) по правилу *modus ponens* получаем $A \supset A$.

Теорема 5.1. Если $\Gamma \vdash A \supset B$ и $\Gamma \vdash A$, то $\Gamma \vdash B$.

Доказательство. Пусть A_1, A_2, \dots, A_n - вывод формулы А из Γ , где A_n совпадает с А. Пусть B_1, B_2, \dots, B_m - вывод формулы $A \supset B$ из Γ , где B_n совпадает с $A \supset B$. Тогда $A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m, B$ - вывод формулы В из Γ . Последняя формула в этом выводе получена применением правила *modus ponens* к формулам A_n и B_m .

Наша цель показать, что формула исчисления высказываний является тавтологией тогда и только тогда, когда она есть теорема. В одну сторону это совсем просто.

Теорема 5.2

1. Любая аксиома в исчислении высказываний является тавтологией.
2. Любая теорема в исчислении высказываний является тавтологией.

Доказательство. То, что каждая аксиома A1-A3 является тавтологией легко проверить с помощью таблиц истинности. Для доказательства пункта 2 теоремы, предварительно докажем, что правило *modus ponens*, примененное к тавтологиям приводит к тавтологиям.

Действительно, пусть при произвольном распределении истинностных значений формулы А и $A \supset B$ являются тавтологиями. Тогда формула А истинна и, по свойствам импликации, В истинно. Следовательно, В - тавтология.

Для доказательства обратного утверждения нам потребуется несколько лемм.

Лемма 5.1. (теорема о дедукции - Эрбран, 1930 г.). Если Γ - множество формул, А и В - формулы и $\Gamma, A \vdash B$, то $\Gamma \vdash A \supset B$. В частности, если $A \vdash B$, то $\vdash A \supset B$.

Доказательство. Пусть B_1, B_2, \dots, B_n есть вывод из $\Gamma \cup \{A\}$, где $B_n = B$. Индукцией по i ($1 \leq i \leq n$) докажем, что $\Gamma \vdash A \supset B_i$.

Базис индукции. B_1 должно быть элементом Γ , либо быть аксиомой, либо совпадать с А. По схеме аксиом A1 формула $B_1 \supset (A \supset B_1)$ является аксиомой. Поэтому в первых двух случаях $\Gamma \vdash A \supset B$ по *modus ponens*. В

третьем случае, когда B_i совпадает с A , мы имеем $\vdash A \supset B_1$ (см. пример 5.3) и, следовательно, $\Gamma \vdash A \supset B_1$.

Индуктивный переход. Допустим теперь, что $\Gamma \vdash A \supset B_k$ для любого $k < i$. Для B_i имеем четыре возможности:

- 1) B_i есть аксиома;
- 2) $B_i \in \Gamma$;
- 3) B_i совпадает с A ;
- 4) B_i следует по modus ponens из некоторых B_s и B_m , где $s < i$ и $m < i$ и B_m имеет вид $B_s \supset B_i$.

В первых трех случаях $\Gamma \vdash A \supset B$ доказывается также как при $i=1$. В последнем случае применим индуктивное предположение, согласно которому $\Gamma \vdash A \supset B_s$ и $\Gamma \vdash A \supset (B_s \supset B_i)$. По схеме аксиом A2, $\vdash (A \supset (B_s \supset B_i)) \supset ((A \supset B_s) \supset (A \supset B_i))$. Следовательно, по modus ponens, $\Gamma \vdash (A \supset B_s) \supset (A \supset B_i)$ и, снова по modus ponens, $\Gamma \vdash A \supset B_i$.

Следствие.

1. $A \supset B, B \supset C \vdash A \supset C$.
2. $A \supset (B \supset C), B \vdash A \supset C$.

Доказательство 1.

- (a) $A \supset B$ гипотеза
- (b) $B \supset C$ гипотеза
- (c) A гипотеза
- (d) B применяя modus ponens из (a) и (c)
- (e) C применяя modus ponens из (b) и (d)

Таким образом, $A \supset B, B \supset C, A \vdash C$. Отсюда по теореме дедукции, $A \supset B, B \supset C \vdash A \supset C$.

Доказательство 2.

- (a) $A \supset (B \supset C)$ гипотеза
- (b) B гипотеза
- (c) A гипотеза
- (d) $B \supset C$ применяя modus ponens из (a) и (c)
- (e) C применяя modus ponens из (b) и (d)

Таким образом, $A \supset (B \supset C), B, A \vdash C$. Отсюда по теореме дедукции, $A \supset (B \supset C), B \vdash A \supset C$.

Теорема о дедукции служит обоснованием для исчисления высказываний следующего приема, который часто используют в математических доказательствах. Для того, чтобы доказать утверждение "Если A , то B " предполагают, что справедливо A и доказывают справедливость B .

Лемма 5.2. [23, стр. 41-43]. Для любых формул A, B следующие формулы являются теоремами:

- (a) $\neg\neg B \supset B$;
- (b) $B \supset \neg\neg B$;
- (c) $\neg A \supset (A \supset B)$;

- (d) $(\neg B \supset \neg A) \supset (A \supset B)$;
 (e) $(A \supset B) \supset (\neg B \supset \neg A)$;
 (f) $A \supset (\neg B \supset \neg (A \supset B))$;
 (g) $(A \supset B) \supset ((\neg A \supset B) \supset B)$.

Доказательство.

(a) $\vdash \neg \neg \neg B \supset B$.

1. $(\neg B \supset \neg \neg B) \supset ((\neg B \supset \neg B) \supset B)$
2. $\neg B \supset \neg B$
3. $(\neg B \supset \neg \neg B) \supset B$
4. $\neg \neg B \supset (\neg B \supset \neg \neg B)$
5. $\neg \neg B \supset B$

схема аксиом A3

пример 5.3

1, 2 и следствие (2) из леммы 5.1

схема аксиом A1

3, 4 и следствие (1) из леммы 5.1

(b) $\vdash \neg B \supset \neg \neg B$.

1. $(\neg \neg \neg B \supset \neg B) \supset ((\neg \neg \neg B \supset B) \supset \neg \neg B)$
2. $\neg \neg \neg B \supset \neg B$
3. $(\neg \neg \neg B \supset B) \supset \neg \neg B$
4. $B \supset (\neg \neg \neg B \supset B)$
5. $B \supset \neg \neg B$

схема аксиом A3

пункт (a), доказанный выше

modus ponens из (1) и (2)

схема аксиом A1

3, 4 и следствие (1) из леммы 5.1

(c) $\vdash \neg A \supset (A \supset B)$.

1. $\neg A$
2. A
3. $A \supset (\neg B \supset A)$
4. $\neg A \supset (\neg B \supset \neg A)$
5. $\neg B \supset A$
6. $\neg B \supset \neg A$
7. $(\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B)$
8. $(\neg B \supset A) \supset B$
9. B

гипотеза

гипотеза

схема аксиом A1

схема аксиом A1

2, 3, modus ponens

1, 4, modus ponens

схема аксиом A3

6, 7, modus ponens

5, 8, modus ponens

Итак, в силу 1-9, $\neg A, A \vdash B$. Поэтому по теореме дедукции, $\neg A \vdash A \supset B$ и, снова по той же теореме, $\vdash \neg A \supset (A \supset B)$.

(d) $\vdash (\neg B \supset \neg A) \supset (A \supset B)$.

1. $\neg B \supset \neg A$
2. A
3. $(\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B)$
4. $A \supset (\neg B \supset A)$
5. $(\neg B \supset A) \supset B$
6. $A \supset B$
7. B

гипотеза

гипотеза

схема аксиом A3

схема аксиом A1

1, 3, modus ponens

4, 5, следствие (1) из леммы 5.1

2, 6, modus ponens

В силу 1-7, $\neg B \supset \neg A, A \vdash B$, после чего, дважды применяя теорему дедукции, получим требуемый результат.

(e) $\vdash (A \supset B) \supset (\neg B \supset \neg A)$.

1. $A \supset B$

гипотеза

- | | |
|--|----------------------------------|
| 2. $\neg\neg A \supset A$ | пункт (a) |
| 3. $\neg\neg A \supset B$ | 1, 2, следствие (1) из леммы 5.1 |
| 4. $B \supset \neg\neg B$ | пункт (b) |
| 5. $\neg\neg A \supset \neg\neg B$ | 3, 4, следствие (1) из леммы 5.1 |
| 6. $(\neg\neg A \supset \neg\neg B) \supset (\neg B \supset \neg A)$ | пункт (d) |
| 7. $\neg B \supset \neg A$ | 5, 6, modus ponens |

В силу 1-7, $A \supset B \mid\!-\! \neg B \supset \neg A$, откуда (e) получаем по теореме дедукции.

(f) $\mid\!-\! A \supset (\neg B \supset \neg(A \supset B))$.

Очевидно, $A, A \supset B \mid\!-\! B$. Применив дважды теорему дедукции, получаем $\mid\!-\! A \supset ((A \supset B) \supset B)$. По пункту (e) имеем $\mid\!-\! ((A \supset B) \supset B) \supset (\neg B \supset \neg(A \supset B))$. Наконец, применив следствие (1) из леммы 5.1, получаем $\mid\!-\! (A \supset B) \supset (\neg B \supset \neg A)$.

(g) $\mid\!-\! (A \supset B) \supset ((\neg A \supset B) \supset B)$.

- | | |
|--|--------------------|
| 1. $A \supset B$ | гипотеза |
| 2. $\neg A \supset B$ | гипотеза |
| 3. $(A \supset B) \supset (\neg B \supset \neg A)$ | пункт (e) |
| 4. $\neg B \supset \neg A$ | 1, 3, modus ponens |
| 5. $(\neg A \supset B) \supset (\neg B \supset \neg\neg A)$ | пункт (e) |
| 6. $\neg B \supset \neg\neg A$ | 2, 5, modus ponens |
| 7. $(\neg B \supset \neg\neg A) \supset ((\neg B \supset \neg A) \supset B)$ | схема аксиом A3 |
| 8. $(\neg B \supset \neg A) \supset B$ | 6, 7, modus ponens |
| 9. B | 4, 8, modus ponens |

Итак, $A \supset B, \neg A \supset B \mid\!-\! B$. Применив два раза теорему дедукции, получаем (g).

Лемма 5.3. Если $\Gamma, A \mid\!-\! B$ и $\Gamma, \neg A \mid\!-\! B$, то $\Gamma \mid\!-\! B$.

Доказательство. По лемме 5.2(g) формула $(A \supset B) \supset ((\neg A \supset B) \supset B)$ является теоремой. Таким образом, $\Gamma \mid\!-\! (A \supset B) \supset ((\neg A \supset B) \supset B)$. Кроме того, по теореме о дедукции $\Gamma \mid\!-\! A \supset B$ и $\Gamma \mid\!-\! \neg A \supset B$. Теперь дважды воспользуемся теоремой 5.1.

Лемма 5.4. [23, стр. 43] Пусть A с высказывательными переменными X_1, X_2, \dots, X_m и пусть задано некоторое распределение истинностных значений для X_1, X_2, \dots, X_m . Пусть тогда X'_i есть X_i , если X_i принимает значение И, и $\neg X_i$, если X_i принимает значение Л, и пусть, наконец, A' есть A , если при этом распределении A принимает значение И, и $\neg A$, если A принимает значение Л. Тогда $X'_1, X'_2, \dots, X'_m \mid\!-\! A'$.

Если, например, A обозначает $\neg(\neg X_1 \supset X_2)$, то для каждой строки истинностной таблицы

X_1	X_2	$\neg(\neg X_1 \supset X_2)$
И	И	Л

Л	И	Л
И	Л	Л
Л	Л	И

лемма 5.3. утверждает факт соответствующей выводимости. Так, в частности, третьей строке соответствует утверждение $X_1, \neg X_2 \vdash \neg\neg(\neg X_1 \supset X_2)$, а четвертой строке $\neg X_1, \neg X_2 \vdash \neg(\neg X_1 \supset X_2)$.

Доказательство. Проведем доказательство с помощью математической индукции по числу n вхождений в A логических связок.

Базис индукции. Если $n=0$, то A представляет из себя просто высказывательную переменную X_1 , и утверждение леммы сводится к $X_1 \vdash X_1$ и $\neg X_1 \vdash \neg X_1$.

Индуктивный переход. Допустим теперь, что лемма верна при любом $j < n$.

Случай 1. A имеет вид отрицания: $\neg B$. Число вхождений логических связок в B , очевидно, меньше n .

Случай 1а. Пусть при заданном распределении истинностных значений B принимает значение И. Тогда A принимает значение Л. Таким образом, B' есть B , а A' есть $\neg A$. По индуктивному предположению, примененному к B , мы имеем $X'_1, X'_2, \dots, X'_m \vdash B$. Следовательно, по лемме 5.2(b) и *modus ponens*, $X'_1, X'_2, \dots, X'_m \vdash \neg\neg B$. Но $\neg\neg B$ и есть A' .

Случай 1б. Пусть B принимает значение Л; тогда B' есть не $\neg B$, а A' совпадает с A . По индуктивному предположению, $X'_1, X'_2, \dots, X'_m \vdash \neg B$, что и требовалось получить, ибо $\neg B$ есть A' .

Случай 2. Формула A имеет вид $B \supset C$. Тогда число вхождений логических связок в B и C меньше, чем в A . Поэтому, в силу индуктивного предположения, $X'_1, X'_2, \dots, X'_m \vdash B$, $X'_1, X'_2, \dots, X'_m \vdash C$.

Случай 2а. Формула B принимает значение Л. Тогда A принимает значение И, и B' есть $\neg B$, а A' есть A . Таким образом, $X'_1, X'_2, \dots, X'_m \vdash \neg B$ и, по лемме 5.2 (с), $X'_1, X'_2, \dots, X'_m \vdash B \supset C$, но $B \supset C$ и есть A .

Случай 2б. Формула C принимает значение И. Следовательно, A принимает значение И и C' есть C , а A' есть A . Имеем $X'_1, X'_2, \dots, X'_m \vdash C$, и тогда, по схеме аксиом $A1$, $X'_1, X'_2, \dots, X'_m \vdash B \supset C$, где $B \supset C$ совпадает с A .

Случай 2с. Формула B принимает значение И и C принимает значение Л. Тогда A' есть $\neg A$, ибо A принимает значение Л, B' есть B и C' есть $\neg C$. Имеем $X'_1, X'_2, \dots, X'_m \vdash B$ и $X'_1, X'_2, \dots, X'_m \vdash \neg C$. Отсюда, по лемме 5.2 (f) получаем $X'_1, X'_2, \dots, X'_m \vdash \neg(B \supset C)$, где $\neg(B \supset C)$ есть A' .

Теорема 5.3. (Пост, 1921) Формула A в исчислении высказываний является теоремой тогда и только тогда, когда A - тавтология.

Доказательство. Нам осталось доказать только половину теоремы, другая половина доказана в теореме 5.2. Итак пусть A есть тавтология, докажем, что она доказуема. Предположим, что X_1, X_2, \dots, X_m - все высказывательные переменные, содержащиеся в A . При каждом распределении ис-

тинностных значений для переменных X_1, X_2, \dots, X_m мы имеем, в силу леммы 5.4, $X'_1, X'_2, \dots, X'_m \vdash A$ (A' совпадает с A , так как A есть тавтология). Поэтому в случае, когда X_m принимает значение И, мы применив лемму 5.4, получаем $X'_1, X'_2, \dots, X'_m \vdash A$; когда же X_m принимает значение Л, то по той же лемме получаем $X'_1, X'_2, \dots, \neg X'_m \vdash A$. Отсюда, по лемме 5.3, $X'_1, X'_2, \dots, X'_{m-1} \vdash A$. Точно таким же образом, рассмотрев два случая, когда X_{m-1} принимает значение И и Л, и снова применив леммы 5.4 и 5.3, мы исключим X_{m-1} и так далее; после m таких шагов мы придем к $\vdash A$.

Исчисление высказываний можно описать системами аксиом, отличными от $A1, A2$ и $A3$. Например, если использовать штрих Шеффера $x|y = \neg(x \& y)$, то достаточно одной схемы аксиом

$$((P|(Q|R))(S|(S|S)))(T|Q)((P|T)|(P|T)))$$

и единственного правила вывода "из A и $A|(B|C)$ следует C ".

§3 Исчисление предикатов

Исчисление предикатов - это аксиоматическая теория, символами которой являются, по существу, те же символы, что и в логике предикатов:

- 1) символы предметных переменных: x_1, x_2, \dots ;
- 2) символы предикатов P, Q, R, A, \dots ;
- 3) логические символы \neg, \supset ;
- 4) символы кванторов; \forall, \exists ;
- 5) скобки и запятая.

Сформулированное в разделе 4.1 определение формулы остается в силе и для исчисления предикатов (с той лишь разницей, что мы употребляем только два логических символа: \neg и \supset ; остальные связки можно определить через эти).

Аксиомы исчисления предикатов. Каковы бы ни были формулы A и B , следующие формулы являются аксиомами (при этом не должно нарушаться определение формулы):

- A1. $A \supset (B \supset A)$;
- A2. $(A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))$;
- A3. $(\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B)$.
- A4. $\forall x A(x) \supset A(y)$, где формула $A(x)$ не содержит переменной y .
- A5. $A(x) \supset \exists y A(y)$, где формула $A(x)$ не содержит переменной y .

Правила вывода исчисления предикатов:

1. Правило *modus ponens*;

2. Правило связывания квантором общности: если $B \supset A(x)$, то $B \supset \forall x A(x)$, где формула B не содержит переменной x .

3. Правило связывания квантором существования: если $A(x) \supset B$, то $\exists x A(x) \supset B$, где формула B не содержит переменной x .

4. Правило переименования связанной переменной. Связанную переменную формулы A можно заменить (в кванторе и во всех вхождениях в области действия квантора) другой переменной, не являющейся свободной в A .

Всякая формальная теория, которая имеет в качестве своих аксиом приведенные выше пять аксиом исчисления предикатов, называется формальной теорией первого порядка.

Многочисленные обычные теории такого рода являются по существу вариантами исчисления предикатов первого порядка, дополненного одной или несколькими аксиомами и/или правилами вывода.

Термин "теория первого порядка" означает, что в их формулах действие квантора может распространяться только на предметные переменные. К теориям второго порядка относятся такие формальные теории, в которых действия кванторов может распространяться и на сами предикаты. В формальных теориях более высокого порядка действие кванторов может распространяться и на предикаты от предикатов и т. д.

В исчислении предикатов первого порядка любая теорема является общезначимой формулой. Нам понадобятся несколько лемм.

Лемма 5.5 (ослабленная теорема о дедукции). Если $\Gamma, A \vdash B$ и существует вывод в исчислении предикатов, построенный с применением только правила *modus ponens*, то $\Gamma \vdash A \supset B$.

Доказательство этой леммы опустим [23, стр. 70].

Лемма 5.6. Если $B \supset A(x)$, где формула B не содержит переменной x , является общезначимой, то формула $B \supset \forall x A(x)$ так же общезначима.

Доказательство. Пусть x_1, x_2, \dots, x_n - все свободные переменные формулы $B \supset \forall x A(x)$. Пусть задана произвольная интерпретация с множеством M и пусть $\langle a_1, a_2, \dots, a_n \rangle$, $a_i \in M$, $1 \leq i \leq n$, - произвольный набор значений свободных переменных формулы $B \supset \forall x A(x)$.

Случай 1. Формула B при этих значениях предметных переменных имеет значение И. Поскольку формула $B \supset A(x)$ - общезначима, то для любого элемента $a \in M$ формула $A(x)$ на наборе $\langle a, a_1, a_2, \dots, a_n \rangle$ имеет значение И (переменная x принимает значение a). Тогда $\forall x A(x)$ - общезначимая формула и, следовательно, $B \supset \forall x A(x)$ - так же общезначимая формула.

Случай 2. Формула B при значениях предметных переменных $\langle a_1, a_2, \dots, a_n \rangle$ имеет значение Л. В силу определения импликации тогда формула $B \supset \forall x A(x)$ имеет значение И.

Лемма 5.7. Если $A(x) \supset B$, где формула B не содержит переменной x , - общезначима, то формула $\exists x A(x) \supset B$ - так же общезначима.

Доказательство. Пусть x_1, x_2, \dots, x_n - все свободные переменные формулы $\exists x A(x) \supset B$. Тогда x, x_1, x_2, \dots, x_n - список всех свободных переменных формулы $A(x) \supset B$. Пусть задана произвольная интерпретация с множеством M и пусть $\langle a_1, a_2, \dots, a_n \rangle, a_i \in M, 1 \leq i \leq n$, - произвольный набор значений свободных переменных формулы $\exists x A(x) \supset B$.

Случай 1. Формула B при значениях предметных переменных $\langle a_1, a_2, \dots, a_n \rangle$ имеет значение I . В силу определения импликации, тогда формула $\exists x A(x) \supset B$ имеет значение I .

Случай 2. Формула B при значениях предметных переменных $\langle a_1, a_2, \dots, a_n \rangle$ имеет значение L . Поскольку формула $A(x) \supset B$ - общезначима, то для любого элемента $a \in M$ формула $A(x)$ на наборе $\langle a, a_1, a_2, \dots, a_n \rangle$ имеет значение L (переменная x принимает значение a). Тогда $\exists x A(x)$ имеет значение L на наборе $\langle a_1, a_2, \dots, a_n \rangle$ и, следовательно, $\exists x A(x) \supset B$ имеет значение I .

Теорема 5.4

1. Аксиомы исчисления предикатов - общезначимые формулы.
2. Формула, получающаяся из общезначимых формулы по любому из правил вывода 1-4, является общезначимой.
3. Любая доказуемая в исчислении предикатов формула общезначима.

Доказательство.

1. Для аксиом A1-A3 утверждение следует из леммы 5.5, для аксиом A4-A5 это следует из теорем 4.1 и 4.2.

2. Для правила *modus ponens* утверждение следует из свойств импликации. Справедливость утверждения для правил вывода 2 и 3 следует из лемм 5.6 и 5.7. Справедливость утверждения для четвертого правила вывода легко доказывается.

3. Эта часть теоремы следует из утверждений 1 и 2.

Доказательство обратной теоремы значительно сложнее.

Теорема 5.6 (теорема Геделя о полноте исчисления предикатов, 1930). Всякая общезначимая формула выводима в исчислении предикатов.

В отличие от случая исчисления высказываний в исчислении предикатов первого порядка у нас нет эффективного способа для распознавания общезначимости.

Теорема 5.7 (теорема Черча о неразрешимости исчисления предикатов, 1936). Не существует алгоритма, который для любой формулы исчисления предикатов первого порядка устанавливает, общезначима она или нет.

§4 Логический вывод

Терпеть не могу логики. Она всегда банальна и нередко убедительна.

Оскар Уайльд

Формальная математика основывается на аксиоматическом методе. Вначале вводятся неопределяемые понятия и аксиомы, далее, на основании логических правил и заключений появляются теоремы. Проблема доказательства в математической логике состоит в установлении истинности формулы B (*заключения*), если предполагается истинность формул A_1, \dots, A_n (*посылок*). Мы записываем это в виде

$$A_1, A_2, \dots, A_n \models B$$

и говорим, что B является *логическим следствием* A_1, A_2, \dots, A_n .

Основной метод решения этой проблемы следующий. Записываем посылки и применяем правила вывода, чтобы получить из них другие истинные формулы. Из этих формул и исходных посылок выводим последующие формулы и продолжаем этот процесс до тех пор, пока не будет получено нужное заключение. Мы называем это логическим выводом или доказательством; именно такой метод обычно применяется в математических доказательствах.

Два классических правила вывода были открыты очень давно. Одно из них носит латинское название *modus ponens* (модус поненс или *сокращение посылки*). Его можно записать следующим образом:

$$A, A \supset B \models B.$$

Второе правило (*цепное*) позволяет вывести новую импликацию из двух данных импликаций. Можно записать его следующим образом:

$$A \supset B, B \supset C \models A \supset C.$$

Во всей своей полноте понятие доказательства несомненно обладает и психологическими признаками. Надо обладать красноречием и умением убеждать, чтобы слушатели (или читатели) приняли ваше доказательство. Рассмотрим основные логические особенности доказательства и выделим некоторые из методов доказательства.

Доказательство с помощью математической индукции

Мы рассмотрели этот способ в разделе 2.4. Подчеркнем, что принцип математической индукции является одним из способов логического вывода и в математике принимается без доказательства.

Доказательство с введением допущения

Многие математические утверждения могут быть представлены в виде импликаций $P \supset Q$. Доказательство такого утверждения может быть представлено в виде

$$A_1, A_2, \dots, A_n \models P \supset Q.$$

Формулы A_1, A_2, \dots, A_n - суть посылки, истинность которых предполагается. Поэтому в данном случае доказательство логически эквивалентно доказательству того, что формула

$$(A_1 \& A_2 \& \dots \& A_n) \supset (P \supset Q)$$

является тавтологией. Последняя формула равносильна формуле

$$(A_1 \& A_2 \& \dots \& A_n \& P) \supset Q.$$

Доказательство истинности этой формулы мы можем записать в виде

$$A_1 \& A_2 \& \dots \& A_n \& P \models Q.$$

Таким образом, для непосредственного доказательства теоремы вида $P \supset Q$, мы предполагаем истинность аксиом и ранее доказанных теорем и допускаем истинность высказывания P и далее показываем, что из всего этого с неизбежностью вытекает истинность Q .

Этот метод часто применяется в геометрии. Например, при доказательстве равенства боковых сторон треугольника, у которого углы при основании равны, допускается, что эти углы равны, а затем это используется в доказательстве равенства сторон.

Косвенные методы доказательства

Рассмотрим условное высказывание вида $A \supset B$, где A - конъюнкция посылок, B - заключение. Иногда удобнее вместо доказательства истинности этого условного высказывания установить логическую истинность некоего другого высказывания, равносильного исходному. Такие формы доказательства называются косвенными методами доказательства.

Еще одной формой косвенного метода доказательства, является доказательство по закону контрапозиции, основанное на равносильности

$$A \supset B \equiv \neg B \supset \neg A,$$

когда вместо истинности $A \supset B$ мы доказываем истинность $\neg B \supset \neg A$.

Преимущества этого метода доказательства проявляются при автоматизированном способе доказательства, т. е. когда доказательство совершают компьютер с помощью специальных программных систем доказательства теорем.

При построении выводов не всегда целесообразно ждать появления искомого заключения, просто применяя правила вывода. Именно такое часто случается, когда мы делаем допущение A для доказательства импликации $A \supset B$. Мы применяем цепное правило и модус поненс к A и другим посылкам, чтобы в конце получить B . Однако можно пойти по неправильному пути, и тогда будет доказано много предложений, большинство из которых не имеет отношения к нашей цели. Этот метод носит название *прямой волны* и имеет тенденцию порождать лавину промежуточных результатов, если его запрограммировать для компьютера и не ограничить глубину.

Другая возможность - использовать контрапозицию и попытаться, например, доказать $\neg B \supset \neg A$ вместо $A \supset B$. Тогда мы допустим $\neg B$ и по-

пробуем доказать $\neg A$. Иными словами, допускается, что заключение В (правая часть исходной импликации) неверно, и делается попытка опровергнуть посылку А. Это позволяет двигаться как бы назад от конца к началу, применяя правила так, что старое заключение играет роль посылки. Такая организация поиска может лучше показать, какие результаты имеют отношение к делу. Она называется *поиском от цели*.

Доказательство приведением к противоречию

Частным случаем косвенных методов доказательства является приведение к противоречию (от противного). В этом методе используются следующие равносильности:

$$A \supset B \equiv \neg(A \supset B) \supset (C \& \neg C) \equiv (A \& \neg B) \supset (C \& \neg C),$$

$$A \supset B \equiv (A \& \neg B) \supset \neg A,$$

$$A \supset B \equiv (A \& \neg B) \supset B.$$

Используя вторую из приведенных равносильностей для доказательства $A \supset B$ мы допускаем одновременно А и $\neg B$, т.е. предполагаем, что заключение ложно:

$$\neg(A \supset B) \equiv \neg(\neg A \vee B) \equiv A \& \neg B.$$

Теперь мы можем двигаться и вперед от А, и назад от $\neg B$. Если В выводимо из А, то, допустив А, мы доказали бы В. Поэтому, допустив $\neg B$, мы получим противоречие. Если же мы выведем $\neg A$ из $\neg B$, то тем самым получим противоречие с А. В общем случае мы можем действовать с обоих концов, выводя некоторое предложение С, двигаясь вперед, и его отрицание $\neg C$, двигаясь назад. В случае удачи это доказывает, что наши посылки *несовместимы* или *противоречивы*. Отсюда мы выводим, что дополнительная посылка $A \& \neg B$ должна быть ложна, а значит противоположное ей утверждение $A \supset B$ истинно. Метод приведения к противоречию часто используется в математике. Например, в геометрии мы можем допустить, что углы при основании некоторого треугольника равны, а противолежащие стороны не равны, и попробовать показать, что при этом и углы должны быть не равны, или получить еще какое-то противоречие.

Доказательство контрпримером

Многие математические гипотезы имеют в своей основе форму: "Все объекты со свойством А обладают свойством В". Мы можем записать это в виде формулы

$$\forall x (A(x) \supset B(x)),$$

где $A(x)$ обозначает предикат "х обладает свойством А", $B(x)$ - "х обладает свойством В". Если число возможных значений х является конечным, то в принципе доказательство может быть проведено с помощью разбора случаев, то есть непосредственной проверкой выполнимости гипотезы для каждого объекта. В случае, если число объектов не является конечным, то такой возможности не существует даже в принципе. Однако для доказа-

тельства ложности гипотезы достаточно привести хотя бы один пример (контрпример), для которого гипотеза не выполняема.

Нет ничего практичнее хорошей теории.

Роберт Кирхгоф

§5 Метод резолюций

Логическое программирование является, пожалуй, наиболее впечатляющим примером применения идей и методов математической логики (точнее, одного из ее разделов - теории логического вывода) в программировании.

Идея использования языка логики предикатов первого порядка в качестве языка программирования возникла еще в 60-ые годы, когда создавались многочисленные системы автоматического доказательства теорем и основанные на них вопросно-ответные системы. Суть этой идеи заключается в том, чтобы программист не указывал машине последовательность шагов, ведущих к решению задачи, как это делается во всех процедурных языках программирования, а описывал на логическом языке свойства интересующей его области, иначе говоря, описывал мир своей задачи. Другие свойства и удовлетворяющие им объекты машина находила бы сама путем построения логического вывода.

Первые компьютерные реализации систем автоматического доказательства теорем появились в конце 50-х годов, а в 1965г. Робинсон предложил свой метод резолюций, который и по сей день лежит в основе большинства систем поиска логического вывода.

Робинсон пришел к заключению, что правила вывода, которые следует применять при автоматизации процесса доказательства с помощью компьютера, не обязательно должны совпадать с правилами вывода, используемыми человеком. Он обнаружил, что общепринятые правила вывода, например, правило *modus ponens*, специально сделаны “слабыми”, чтобы человек мог интуитивно проследить за каждым шагом процедуры доказательства. Правило резолюции более сильное, оно трудно поддается восприятию человеком, но эффективно реализуется на компьютере.

Вскоре метод резолюций был использован в качестве основы нового языка программирования. Так в 1972 году родился язык Пролог (“ПРО-граммирование в терминах ЛОГики”), быстро завоевавший популярность во всем мире.

Сформулируем сначала правило резолюций в рамках исчисления высказываний. Пусть A , B и X - формулы. Предположим, что две формулы $A \vee X$ и $B \vee \neg X$ - истинны. Если X тоже истина, то отсюда можно заключить, что B истинна. Наоборот, если X ложна, то можно заключить, что A ис-

тинна. В обоих случаях формула $A \vee B$ истина. Это логическое следствие мы можем записать в виде правила

$$A \vee X, B \vee \neg X \models A \vee B,$$

которое можно записать также в виде

$$\neg X \supset A, X \supset B \models A \vee B.$$

В том частном случае, когда X - высказывание, а A и B - элементарные дизъюнкции, то это правило называется *правилом резолюций*. Сравним это правило с уже известными нам:

$$\text{цепное правило: } \neg A \supset X, X \supset B \models \neg A \supset B,$$

$$\text{модус поненс: } X, X \supset B \models B.$$

Правило резолюции можно рассматривать как аналог цепного правила в применении к формулам, находящимся в конъюнктивной нормальной форме. Правило модус поненс также можно считать частным случаем правила резолюции при ложном A .

Доказательства, основанные на принципе резолюций, выделяются среди прочих тем, что они дают возможность использовать средства автоматического доказательства, применяемые в логическом программировании.

Чтобы применить правило резолюции, будем действовать следующим образом. Используем доказательство от противного и допускаем отрицание заключения.

1. Приводим все посылки и отрицание заключения, принятое в качестве дополнительной посылки, к конъюнктивной нормальной форме:

а) устраним символы \supset и \sim с помощью эквивалентностей

$$A \sim B = (A \supset B) \& (B \supset A),$$

$$A \supset B = \neg A \vee B;$$

б) продвигаем отрицания внутрь с помощью закона де Моргана;

в) применяем дистрибутивность $A \vee (B \& C) = (A \vee B) \& (A \vee C)$.

2. Теперь каждая посылка превратилась в конъюнкцию элементарных дизъюнктов (будем их в дальнейшем называть просто дизъюнктами), может быть, одночленную. Выписываем каждый дизъюнкт с новой строки; все дизъюнкты истинны, так как конъюнкция истинна по предположению.

3. Каждый дизъюнкт - это дизъюнкция (возможно, одночленная), состоящая из переменных и отрицаний переменных. Именно к ним применим метод резолюций. Берем любые два дизъюнкта, содержащие одну и ту же переменную, но с противоположными знаками, например,

$$X \vee Y \vee Z \vee \neg P,$$

$$X \vee P \vee W.$$

Применяем правило резолюции и получаем $X \vee Y \vee Z \vee W$.

4. Продолжаем этот процесс, пока не получится P и $\neg P$ для некоторой переменной P . Применяя резолюцию и к ним, получим пустой дизъ-

юнкт, выражающий противоречие, что завершает доказательство от противного.

В качестве примера рассмотрим доказательство соотношения

$$P \vee Q, P \supset R, Q \supset S \models R \vee S.$$

Приводим посылки к нормальной форме и выписываем их на отдельных строках.

$$P \vee Q \quad (1)$$

$$\neg P \vee R \quad (2)$$

$$\neg Q \vee S \quad (3)$$

Записываем отрицание заключения и приводим его к нормальной форме.

$$\neg(R \vee S) \equiv \neg R \& \neg S$$

$$\neg R \quad (4)$$

$$\neg S \quad (5)$$

Выводим пустой дизъюнкт с помощью резолюции.

$$\neg P \quad \text{из (2) и (4)} \quad (6)$$

$$Q \quad \text{из (1) и (6)} \quad (7)$$

$$\neg Q \quad \text{из (3) и (5)} \quad (8)$$

$$\text{пустой} \quad \text{из (7) и (8)}$$

Для того, чтобы сформулировать правило резолюций для исчисления предикатов первого порядка, нам потребуется обобщить понятие формулы в исчислении предикатов, введенное в 4.1. Мы допускаем, что аргументами предикатов могут быть не только переменные из предметной области, но и константы этой области, а также составные объекты, сконструированные из переменных и констант предметной области.

Точнее, пусть нам дано множество *функциональных символов* f_1, f_2, f_3, \dots , каждый функциональный символ имеет вполне определенную местность, т.е. требует определенное количество аргументов.

Множество *термов* - это наименьшее подмножество языка логики предикатов, удовлетворяющее двум условиям:

а) переменные и константы суть (атомарные) термы;

в) если f_i - функциональный символ местности r , а t_1, t_2, \dots, t_r - термы, то $f_i(t_1, t_2, \dots, t_r)$ - терм.

Мы будем называть формулу *атомарной*, если она выражается каким-то n -местным предикатом, аргументами которого служат термы. Используя атомарные формулы как первоначальные, понятие формулы общего вида определяем как в 4.1.

Фразовая форма логики предикатов - это способ записи формул, при котором употребляются только связки $\&$, \vee и \neg . *Литерал* - это позитивная или негативная атомарная формула. Каждая *фраза* (или *клауза*) - это множество литералов, соединенных символом \vee . Фразу можно рассматривать как обобщение понятия импликации. Если A и B - атомарные формулы, то формула

$$A \supset B$$

может также быть записана как

$$B \vee \neg A.$$

Простейшая фраза содержит только один литерал, позитивный или негативный.

Фраза с одним позитивным литералом называется *фразой* (или *клаузой*) *Хорна*. Любая фраза Хорна представляет импликацию: так, например,

$$D \vee \neg F \vee \neg E \text{ равносильно } (F \& E) \supset D.$$

Правило резолюции действует следующим образом. Две фразы могут быть резольвированы друг с другом, если одна из них содержит позитивный литерал, а другая - соответствующий негативный литерал с одним и тем же обозначением предиката и одинаковым количеством аргументов, и если аргументы обоих литералов могут быть *унифицированы* (т.е. согласованы) друг с другом. Рассмотрим две фразы:

$$P(a) \vee \neg Q(b,d), \quad (1)$$

$$Q(b,d) \vee \neg R(b,d). \quad (2)$$

Поскольку во фразе (1) содержится негативный литерал $\neg Q(b,d)$, а во фразе (2) - соответствующий позитивный литерал $Q(b,d)$ и аргументы обоих литералов могут быть унифицированы (т.е. b унифицируется с b , а d унифицируется с d), то фраза (1) может быть резольвирована с фразой (2). В результате этого получается фраза (3), которая называется *резольвентой*:

$$P(a) \vee \neg R(b,d). \quad (3)$$

Фразы (4) и (5) не резольвируются друг с другом, так как аргументы литералов Q не поддаются унификации:

$$P(a) \vee \neg Q(b,d), \quad (4)$$

$$Q(d,d) \vee \neg R(b,d). \quad (5)$$

Унификация переменных. Во фразовой форме не употребляется явная квантификация переменных. Неявно, однако, все переменные квантифицированы кванторами всеобщности. Так, во фразе $Q(x,y) \vee \neg R(x,y)$ подразумевается наличие кванторов:

$$\forall x \forall y (Q(x,y) \vee \neg R(x,y)).$$

Если в качестве аргумента выступает переменная, то она унифицируется с любой константой. Если в одной и той же фразе переменная встречается более одного раза и эта переменная в процессе резолюции унифицируется с константой, то резольвента будет содержать данную константу

на тех местах, где рассматриваемая переменная располагалась в исходной фразе. Например, фразы

$$P(a) \vee \neg Q(a,b), \quad (6)$$

$$Q(x,y) \vee R(x,y) \quad (7)$$

резольвируемы, поскольку аргументы литерала Q унифицируются. При этом переменная x унифицируется с константой a , а переменная y - с константой b . Обратите внимание, что во фразе (8), т.е. в резольvente

$$P(a) \vee \neg R(a,b), \quad (8)$$

переменные, служившие аргументами R во фразе (7), теперь заменены константами.

Следующий рекурсивный алгоритм выясняет унифицируемы ли два терма S и T .

1. Если S и T - константы, то S и T - унифицируемы т. и т. т., когда они являются одним и тем же объектом.

2. Если S - переменная, а T - произвольный терм, то они унифицируемы и S приписывается значение T . Наоборот, если T - переменная, а S - произвольный терм, то T получает в качестве значения S .

3. Если S и T - не атомарные термы, то они унифицируемы т. и т. т., когда а) S и T имеют одинаковый главный функциональный символ и б) все их соответствующие компоненты (подтермы) унифицируемы.

Результирующая конкретизация определяется унификацией компонент.

Применение метода резолюций для ответов на вопросы

Рассмотрим следующий пример из [11]. Предположим, что у нас есть предикат $F(x,y)$, означающий, что x отец y , и истинна следующая формула $F(\text{john},\text{harry}) \& F(\text{john},\text{sid}) \& F(\text{sid},\text{liz})$.

Таким образом, у нас есть три дизъюнкта. Они не содержат переменных или импликаций, а просто представляют базисные факты.

Введем еще три предиката $M(x)$, $S(x,y)$ и $B(x,y)$, означающие соответственно, что x - мужчина, что он единокровен с y , что он брат y . Мы можем записать следующие аксиомы о семейных отношениях:

$$\forall x,y (F(x,y) \supset M(x));$$

$$\forall x,y,w ((F(x,y) \& F(x,w)) \supset S(y,w));$$

$$\forall x,y ((S(x,y) \& M(x)) \supset B(x,y)).$$

Они утверждают следующее: 1) все отцы - мужчины; 2) если у детей один отец, то они единокровны; 3) брат - это единокровный мужчина.

Пусть мы задали вопрос $\exists z B(z,\text{harry})$? Чтобы найти ответ с помощью метода резолюции, мы записываем отрицание вопроса $\forall z \neg B(z,\text{harry})$. Затем приводим аксиомы к нормальной форме и записываем каждый дизъюнкт в отдельной строке (так как каждый дизъюнкт истинен сам по себе):

$$\neg F(x,y) \vee M(x); \quad (1)$$

$$\neg F(x,y) \vee \neg F(x,w) \vee S(y,w); \quad (2)$$

$$\neg S(x,y) \vee \neg M(x) \vee B(x,y); \quad (3)$$

$$F(\text{john}, \text{harry}); \quad (4)$$

$$F(\text{john}, \text{sid}); \quad (5)$$

$$F(\text{sid}, \text{liz}); \quad (6)$$

$$\neg B(z, \text{harry}). \quad (7)$$

Мы не пишем внешних кванторов всеобщности, так как подразумевается, что каждая переменная связана таким квантором.

Для применения резолюции необходимо найти для данной пары дизъюнктов такую подстановку термов вместо переменных, чтобы после нее некоторый литерал одного из дизъюнктов стал отличаться от некоторого литерала другого дизъюнкта лишь отрицанием. Мы можем делать подстановки, так как все переменные связаны кванторами всеобщности. Если, например, мы подставим *john* вместо *x* и *sid* вместо *y*, то получим следующее:

$$\neg F(\text{john}, \text{sid}) \vee \neg F(\text{john}, w) \vee S(\text{sid}, w).$$

Мы можем применить правило резолюции к этому дизъюнкту и к (5), что дает новый дизъюнкт

$$\neg F(\text{john}, w) \vee S(\text{sid}, w) \quad (8)$$

из (5) и (2) $\{x \rightarrow \text{john}, y \rightarrow \text{sid}\}$.

Продолжая, получим

$$S(\text{sid}, \text{harry}) \quad (9)$$

из (4) и (8) $\{w \rightarrow \text{harry}\}$,

$$M(\text{sid}) \quad (10)$$

из (6) и (1) $\{x \rightarrow \text{sid}, y \rightarrow \text{liz}\}$,

$$\neg S(\text{sid}, y) \vee B(\text{sid}, y) \quad (11)$$

из (10) и (3) $\{x \rightarrow \text{sid}\}$,

$$B(\text{sid}, \text{harry}) \quad (12)$$

из (9) и (11) $\{y \rightarrow \text{harry}\}$,

пустой дизъюнкт

из (12) и (7) $\{z \rightarrow \text{sid}\}$.

Таким образом, мы вывели дизъюнкт (12), выражающий, что *sid* брат *harry*, используя аксиомы и факты (4), (5) и (6). Это противоречит отрицанию нашего вопроса, которое утверждает, что *harry* не имеет братьев.

Этот пример демонстрирует возможности метода резолюций.

У логики один недостаток: она не останавливается на полпути.

Д. Уиндем "День триффидов".

Некоторые вещи недоступны человеческому уму, но мы не знаем какие.
"Закон Джеффи"

§6 Неполнота математики

Таким образом, показано, что класс всех теорем исчисления предикатов совпадает с классом общезначимых формул. На этом примере мы видим силу формального аксиоматического метода. Но насколько этот метод действительно силен? Из школьной программы вы знаете, что аксиоматическую теорию еще использовал Евклид при изложении геометрии. Может быть возможно аксиоматизировать всю математику?

В 30-е годы анонимная группа французских математиков под псевдонимом Никола Бурбаки начала писать трактат "Начала математики" (он не окончен и по сей день). Используя аксиомы теории множеств и исчисления предикатов (в несколько большем варианте, чем в предыдущем изложении), Н. Бурбаки излагает всю современную математику с единой (формально-аксиоматической) точки зрения. Они начали не на пустом месте.

В 1889 г. Пеано предложил свои аксиомы для аксиоматизации понятия натурального числа и, после этого была создана формальная теория, известная под названием *формальная арифметика*. Это теория является расширением исчисления предикатов. Аксиомы Пеано следующие:



Джузеппе Пеано

- 1) 1 есть натуральное число;
- 2) следующее за натуральным числом есть натуральное число;
- 3) 1 не следует ни за каким натуральным числом;
- 4) если натуральное число a следует за натуральным числом b и за натуральным числом c , то натуральные числа b и c тождественны;
- 5) если какое-либо предложение доказано для 1 и если из допущения, что оно верно для натурального числа n , вытекает, что оно верно для следующего за n натурального числа, то это предложение верно для всех натуральных чисел (принцип математической индукции).

Естественно было надеяться, что метод формальной аксиоматической теории позволит строить все содержание математики на такой точной и, казалось бы, надежной основе, как понятие выводимой формулы (теоремы формальной системы). Однако в 1931 году Курт Гедель доказал свою знаменитую теорему о неполноте.

Теорема 5.8 (теорема Геделя о неполноте). Всякая естественная непротиворечивая аксиоматическая теория T (формализация) арифметики или любой другой математической теории, содержащей арифметику (например, теория множеств), неполна и непополнима в том смысле, что а) в T имеются содержательно истинные неразрешимые формулы, т. е. такие формулы A , что ни A , ни отрицание A не выводимы (не доказуемы) в T ; б) каким бы конечным множеством дополнительных аксиом не расширить систему T , в новой, усиленной таким образом формальной системе неизбежно появятся свои неразрешимые формулы.



Курт Гёдель

При доказательстве своей теоремы Гёдель, образно говоря, построил математическую формулу, которая гласит следующее: "Я не доказуема". Если эта формула ложна, то тогда получаем, что она доказуема. Но поскольку любое доказательство в математике приводит только к истинным утверждениям, то мы приходим к противоречию. Обратное предположение, т. е. истинность этой формулы, сразу приводит к неполноте математики. Занимательное изложение теоремы Гёделя вместе с доказательством можно прочесть в книге Р. Смаллиана [27].

Со времен доказательства Гёделем своей теоремы математики искали пример такой истинной теоремы, которую было бы невозможно доказать используя аксиоматику Пеано. Только в 1977 году удалось обнаружить такую теорему в комбинаторике, так называемую бесконечную теорему Рамсея.

То, что истинное предложение бывает недоказуемым, можно проиллюстрировать таким примером. Рассмотрим предложение: для всякого натурального n имеет место формула $1+2+\dots+n = n(n+1)/2$. Обычно это доказывают, применяя аксиому математической индукции. Если бы в нашем распоряжении не было аксиомы индукции, то эту формулу нельзя было бы доказать, ибо среди всех аксиом арифметики и логики одна лишь аксиома индукции позволяет делать утверждения обо всей бесконечной совокупности натуральных чисел.

Можно было бы попытаться сбросить ярмо индукции и рассуждать так: если бы для некоторого n сумма $1+2+\dots+n$ не равнялась числу $n(n+1)/2$, то существовало бы наименьшее такое n ; оно не могло бы равняться 1, поскольку наше утверждение для $n=1$ верно; но оно не могло бы и быть больше 1, ибо тогда можно было бы показать, что $n-1$ тоже исключительное число, а это противоречит тому, что n - наименьшее из таких чисел. Увы, это рассуждение основано на принципе, утверждающем, что

каждое непустое множество натуральных чисел имеет наименьший элемент, а этот принцип равносителен аксиоме индукции.

Итак, без аксиомы индукции простые арифметические утверждения вроде $1+2+\dots+n = n(n+1)/2$ даже, несмотря на то, что они истинны, нельзя было бы вывести из остальных аксиом; можно сказать, что без аксиомы индукции арифметика была бы неполной.

Если бы я захотел читать, еще не зная букв, это было бы бессмыслицей. Точно так же, если бы я захотел судить о явлениях природы, не имея никакого представления о началах вещей, это было бы такой же бессмыслицей.

М.В. Ломоносов

Глава 6. ТЕОРИЯ АЛГОРИТМОВ

§1 Понятие алгоритма и неформальная вычислимость

В этом разделе будет уточнено понятие алгоритма. Кроме того, будут даны строгие математические понятия, которые формализуют представление о том, что некоторые функции поддаются вычислению с помо-

стью алгоритма, скажем на компьютере, как только для этого будет составлена надлежащая программа, тогда как другие функции, заданные неэффективным определением, могут требовать творческого подхода для вычисления своих значений.

Под *алгоритмом* понимается способ преобразования представления информации. Слово *algorithm* - произошло от имени аль-Хорезми - автора известного арабского учебника по математике (от его имени произошли также слова "алгебра" и "логарифм").

Интуитивно говоря, алгоритм - некоторое формальное предписание, действуя согласно которому можно получить решение задачи.

Алгоритмы типичным образом решают не только частные задачи, но и классы задач. Подлежащие решению частные задачи, выделяемые по мере надобности из рассматриваемого класса, определяются с помощью параметров. Параметры играют роль исходных данных для алгоритма.

Основные особенности алгоритма

Определенность. Алгоритм разбивается на отдельные шаги (этапы), каждый из которых должен быть простым и локальным.

Ввод. Алгоритм имеет некоторое (быть может, равное нулю) число входных данных, т. е. величин, заданных ему до начала работы.

Вывод. Алгоритм имеет одну или несколько выходных величин, т. е. величин, имеющих вполне определенное отношение к входным данным.

Детерминированность. После выполнения очередного шага алгоритма однозначно определено, что делать на следующем шаге.

Полезные алгоритмы должны быть практичными и хорошими с эстетической точки зрения.

Примеры алгоритмов широко известны: изучаемые в школе правила сложения и умножения десятичных чисел или, скажем, алгоритмы сортировки массивов. Для алгоритмически разрешимой постановки задачи всегда имеется много различных способов ее решения, т. е. различных алгоритмов.

Примеры "почти" алгоритмов: медицинский и кулинарный рецепты. Кстати, почему такие рецепты во многих случаях нельзя рассматривать как алгоритмы?

Данное здесь определение алгоритма не является, конечно, строгим, но оно интуитивно кажется вполне определенным. К сожалению, для решения некоторых задач не существует алгоритма. Установление таких фактов требует введение строгого понятия алгоритма.

Мы будем рассматривать алгоритмы, имеющие дело только с натуральными числами. Можно доказать, что это не является потерей общности, так как объекты другой природы можно закодировать натуральными числами. Для пользователей компьютеров такое утверждение должно быть очевидным.

Пусть \subseteq обозначает множество натуральных чисел $\{0, 1, 2, \dots\}$. Объекты, которые мы будем рассматривать будут функциями с областью определения $D_f \subseteq \subseteq^k$ (k - целое положительное число) и с областью значений $R_f \subseteq \subseteq$. Такие функции будем называть *k-местными частичными*. Слово "частичная" должно напомнить о том, что функция определена на подмножестве \subseteq^k (конечно, в частном случае может быть $D_f = \subseteq^k$, тогда функция называется *всюду определенной*).

Назовем k -местную частичную функцию $f: \subseteq^k \rightarrow \subseteq$ *эффективно вычислимой* (или просто *вычислимой*), если существует алгоритм, который вычисляет f . Этот алгоритм должен удовлетворять следующим критериям:

1. Если на вход алгоритма поступил вектор $x = \langle x_1, x_2, \dots, x_k \rangle$ из D_f , то вычисление должно закончиться после конечного числа шагов и выдать $f(x)$.
2. Если на вход алгоритма поступил вектор x , не принадлежащий области определения D_f , то алгоритм никогда не заканчивается.

Множество эффективно вычисляемых функций мы не отождествляем с множеством "практически вычисляемых" функций, так как не накладываем на первое множество никаких ограничений, связанных с современными вычислительными машинами.

Хотя каждое входное натуральное число, должно быть конечным, тем не менее не предполагается верхняя граница размера этого числа, так, например, количество цифр числа может быть больше числа электронов во Вселенной. Точно также, нет никакой верхней границы на число шагов, которые может сделать алгоритм для конкретных x из области определения.

Должно быть совершенно ясным, что предыдущее определение вычислимой функции не является формальным, поэтому мы собираемся дать строгое определение нового множества функций, которое в некотором смысле будет совпадать с множеством вычислимых функций. Мы дадим три различные формализации понятия алгоритма.

Все должно быть изложено так просто, как только возможно, но не проще.

Альберт Эйнштейн

§2 Частично-рекурсивные функции

Определения

Этот подход к формализации понятия алгоритма принадлежит Гёделю и Клини (1936).

Основная идея Гёделя состояла в том, чтобы получить все вычислимые функции из существенно ограниченного множества базисных функций с помощью простейших алгоритмических средств.

Множество исходных функций таково:



- постоянная функция $0(x) = 0$;
- одноместная функция следования $s(x) = x+1$;
- функция проекции pr_i , $1 \leq i \leq k$, $pr_i(x) = x_i$.

Нетривиальные вычислительные функции можно получать с помощью композиции (суперпозиции) уже имеющихся функций. Этот способ явно алгоритмический.

• Оператор суперпозиции. Говорят, что k -местная функция $f(x_1, x_2, \dots, x_k)$ получена с помощью суперпозиции из m -местной функции $\varphi(y_1, y_2, \dots, y_m)$ и k -местных функций $g_1(x_1, x_2, \dots, x_k)$, $g_2(x_1, x_2, \dots, x_k)$, ..., $g_m(x_1, x_2, \dots, x_k)$, если

$$f(x_1, x_2, \dots, x_k) = \varphi(g_1(x_1, x_2, \dots, x_k), g_2(x_1, x_2, \dots, x_k), \dots, g_m(x_1, x_2, \dots, x_k)).$$

Второй (несколько более сложный) способ действует так.

• Прimitивная рекурсия. При $n \geq 0$ из n -местной функции f и $(n+2)$ -местной функции g строится $(n+1)$ -местная функция h по следующей схеме:

$$h(x_1, \dots, x_n, 0) = f(x_1, \dots, x_n),$$

$$h(x_1, \dots, x_n, y+1) = g(x_1, \dots, x_n, y, h(x_1, \dots, x_n, y)).$$

При $n=0$ получаем (a - константа)

$$f(0) = a;$$

$$f(y+1) = g(y, f(y)).$$

Две упомянутых способа позволяют задать только всюду определенные функции. Частично-определенные функции порождаются с помощью третьего гёделева механизма.

• Оператор минимизации. Эта операция ставит в соответствие частичной функции $f: \subseteq^{k+1} \rightarrow \subseteq$ частичную функцию $h: \subseteq^k \rightarrow \subseteq$, которая определяется так:

1) область определения $D_h = \{ \langle x_1, \dots, x_k \rangle \mid \text{существует } x_{k+1} \geq 0, f(x_1, \dots, x_k, x_{k+1}) = 0 \text{ и } \langle x_1, \dots, x_k, y \rangle \in D_f \text{ для всех } y \leq x_{k+1} \}$;

2) $h(x_1, \dots, x_k) =$ наименьшее значение y , при котором $f(x_1, \dots, x_k, y) = 0$.

Оператор минимизации обозначается так $h(x_1, \dots, x_k) = \mu y [f(x_1, \dots, x_k, y) = 0]$. Очевидно, что даже если f всюду определено, но нигде не обращается в 0, то $\mu y [f(x_1, \dots, x_k, y) = 0]$ нигде не определено. Естественный путь вычисления $h(x_1, \dots, x_k)$ состоит в подсчете значения $f(x_1, \dots, x_k, y)$ последовательно для $y = 0, 1, 2, \dots$ до тех пор, пока не найдется y , обращающее $f(x_1, \dots, x_k, y)$ в 0. Этот алгоритм не остановится, если $f(x_1, \dots, x_k, y)$ нигде не обращается в 0.

Все функции, которые можно получить из базисных функций за конечное число шагов только с помощью трех указанных механизмов, называются *частично-рекурсивными*. Если функция получается всюду определенная, то тогда она называется *общерекурсивной*. Если функция получена без механизма минимизации, то в этом случае она называется *примитивно-рекурсивной*.

Можно легко показать [23, с.136], что введение фиктивных переменных, а также перестановка и отождествление переменных не выводят за

пределы класса примитивно-рекурсивных функций и класса частично-рекурсивных функций. Это проще всего объяснить на примерах.

Введение фиктивных переменных. Если $g(x_1, x_3)$ - примитивно-рекурсивная функция и $f(x_1, x_2, x_3) = g(x_1, x_3)$, то $f(x_1, x_2, x_3)$ - примитивно-рекурсивная функция.

Перестановка переменных. Если $g(x_1, x_2)$ - примитивно-рекурсивная функция и $f(x_2, x_1) = g(x_1, x_2)$, то f есть также примитивно-рекурсивная функция.

Отождествление переменных. Если $g(x_1, x_2, x_3)$ - примитивно-рекурсивная функция и $f(x_1, x_2) = g(x_1, x_2, x_1)$, то $f(x_1, x_2)$ есть также примитивно-рекурсивная функция.

Примеры рекурсивности

Рассмотрим примеры частично-рекурсивных функций. Все эти примеры и много других можно найти в [21, 23].

Сложение двух чисел

$$\text{sum}: \langle x, y \rangle \rightarrow x + y.$$

Эта функция является общерекурсивной в силу примитивной рекурсии

$$\text{sum}(x, 0) = \text{pr}_1(x) = x,$$

$$\text{sum}(x, y+1) = s(\text{sum}(x, y)) = \text{sum}(x, y) + 1.$$

Умножение двух чисел

$$\text{prod}: \langle x, y \rangle \rightarrow x \cdot y.$$

Используем примитивную рекурсию

$$\text{prod}(x, 0) = 0(x) = 0,$$

$$\text{prod}(x, y+1) = \text{sum}(\text{prod}(x, y), x).$$

Усеченное вычитание 1

$$\delta(x) = x - 1, \text{ если } x > 0,$$

$$\delta(0) = 0.$$

Эта функция примитивно-рекурсивна, действительно,

$$\delta(0) = 0 = 0(x),$$

$$\delta(y+1) = y = \text{pr}_2(\langle x, y \rangle).$$

Усеченная разность

$$x \dot{-} y = x - y, \text{ если } x \geq y,$$

$$x \dot{-} y = 0, \text{ если } x < y.$$

Эта функция примитивно-рекурсивна, действительно,

$$x \dot{-} 0 = x,$$

$$x \dot{-} (y+1) = \delta(x \dot{-} y).$$

Модуль разности

$|x-y| = x-y$, если $x \geq y$,

$|x-y| = y-x$, если $x < y$.

Эта функция примитивно-рекурсивна в силу суперпозиции

$|x-y| = (x \div y) + (y \div x)$.

Факториал

Действительно,

$0! = 1$,

$(y+1)! = \text{prod}(y!, y+1)$.

$\min(x, y)$ - наименьшее из чисел x и y

В силу суперпозиции: $\min(x, y) = x \div (x \div y)$.

Знак числа

$\text{sg}(x) = 0$, если $x = 0$,

$\text{sg}(x) = 1$, если $x > 0$.

В силу рекурсии

$\text{sg}(0) = 0$,

$\text{sg}(y+1) = 1$.

$\text{rm}(x, y)$ - остаток от деления y на x

В силу рекурсии и суперпозиции

$\text{rm}(x, 0) = 0$,

$\text{rm}(x, y+1) = \text{prod}(\text{sg}(\text{rm}(x, y)), \text{sg}(|x - \text{rm}(x, y)|))$.

Используя функции, для которых уже установлено, что они являются частично-рекурсивными, мы получаем все новые и новые частично-рекурсивные функции. Существуют критерии, которые позволяют установить частичную рекурсивность сразу для обширных классов функций (см., например, 23, с. 135-151).

Используя минимизацию (μ -оператор) можно получать частично-определенные функции из всюду определенных функций. Например, полагая $f(x, y)$ есть частично-рекурсивная функция $|x - y^2|$, мы обнаруживаем, что $g(x) = \mu y[f(x, y) = 0]$ - не всюду определенная функция

$g(x) = \sqrt{x}$, если x есть точный квадрат и неопределена в противном случае.

Таким образом, тривиально используя μ -оператор вместе с суперпозицией и рекурсией, можно построить больше функций, исходя из основных, чем только с помощью суперпозиции и рекурсии (так как эти операции порождают из всюду определенных функций - всюду определенные). Существуют, однако, и общерекурсивные (всюду определенные) функции, для построения которых нельзя обойтись без минимизации. Примером такой функции является функция Аккермана [13, с. 53]:

$$\begin{aligned}f(0,y) &= y+1, \\f(x+1,0) &= f(x,1), \\f(x+1,y+1) &= f(x, f(x+1,y))\end{aligned}$$

Чистая математика - это такой предмет, где мы не знаем, о чем мы говорим, и не знаем, истинно ли то, что мы говорим.

Бертран Рассел

Человек должен верить, что непонятное можно понять; иначе он не стал бы размышлять о нем.

В. Гёте

§3 Ламбда - исчисление

Значение ламбда-исчисления

Ламбда-исчисление было изобретено Алонсом Чёрчем около 1930 г. Чёрч первоначально строил λ -исчисление как часть общей системы функций, которая должна стать основанием математики. Но из-за найденных парадоксов эта система оказалась противоречивой. Книга Чёрча [2] содержит непротиворечивую подтеорию его первоначальной системы, имеющую дело только с функциональной частью. Эта теория и есть λ -исчисление.

Ламбда-исчисление - это безтиповая теория, рассматривающая функции как *правила*, а не как графики. В противоположность подходу (вводимому функции как множество пар, состоящих из аргумента и значения) более старое понятие определяет функцию как процесс перехода от аргумента к значению. С первой точки зрения x^2-4 и $(x+2)(x-2)$ - разные обозначения одной и той же функции; со второй точки зрения это разные функции.

Функции как правила рассматриваются в полной общности. Например, мы можем считать, что функции заданы определениями на обычном русском языке и применяются к аргументам также описанным по-русски. Также мы можем рассматривать функции заданными программами и применяемые к другим программам. В обоих случаях перед нами *безтиповая структура*, где объекты изучения являются одновременно и функциями и



Алонсо Черч

аргументами. Это отправная точка безтипового λ -исчисления. В частности, функция может применяться к самой себе.

Лямбда-исчисление представляет класс (частичных) функций (λ -определимые функции), который в точности характеризует неформальное понятие эффективной вычислимости. Другими словами, λ -исчисления, наряду с другими подходами формализует понятие алгоритма [7, с. 143-146].

Лямбда-исчисление стало объектом особенно пристального внимания в информатике после того, как выяснилось, что оно представляет собой удобную теоретическую модель современного функционального программирования [32].

Большинство функциональных языков программирования, например, Лисп, используют λ -исчисление в качестве промежуточного кода, в который можно транслировать исходную программу. Функциональные языки "улучшают" нотацию λ -исчисления в прагматическом смысле, но при этом, в какой-то мере, теряется элегантность и простота последнего.

Изучение и понимание многих сложных ситуаций в программировании, например, таких, как автоаппликативность (самоприменимость) или авторепликативность (самовоспроизводство), сильно облегчается, если уже имеется опыт работы в λ -исчислении, где выделены в чистом виде основные идеи и трудности.

Лямбда-выражения и их вычисление

Что значит "функция $5x^3+2$ "? Если кто-то хочет быть точным, он вводит по этому поводу функциональный символ, например f , и говорит: "функция $f : \nabla \rightarrow \nabla$, определенная соотношением $f(x) = 5x^3+2$ ". При этом очевидно, что переменную x можно здесь, не меняя смысла, заменить на другую переменную y . Лямбда-запись устраняет произвольность в выборе f в качестве функционального символа. Она предлагает вместо f выражение " $\lambda x. 5x^3+2$ ".

Кроме того, обычная запись $f(x)$ может обозначать как имя функции f , так и вызов функции с аргументом x . Для более строгого подхода это необходимо различать. В лямбда-обозначениях вызов функции с аргументом x выглядит как $(\lambda x. 5x^3+2)x$.

Русский математик Шейфинкель заметил, что не обязательно вводить функции более чем одной переменной [3]. Действительно, для функции, скажем от двух переменных, $f(x,y)$ мы можем рассмотреть функцию g_x с соотношением $g_x(y) = f(x,y)$, а затем f' с соотношением $f'(x) = g_x$. Отсюда $(f'(x))(y) = f(x,y)$. Позднее Карри [1] переоткрыл это свойство и по этому сейчас сведение функций с несколькими переменными только к функциям одного переменного носит название карринг.

Лямбда-исчисление изучает функции и их аппликативное поведение (т. е. поведение относительно применения к аргументу). Поэтому приме-

нение (аппликация) функции к аргументу является исходной операцией λ -исчисления. Функция f , примененная к аргументу a , обозначается через $f\ a$. Поэтому записи функции в виде $\lambda x. x^2+3$ соответствует правило:

для любого a $(\lambda x. x^2+3)\ a = a^2+3$.

Выражение $x + 2y$ можно считать как функцию от x (записывается $\lambda x. x+2y$) и как функцию от y (записывается $\lambda y. x+2y$). Вызов $(\lambda x. x+2y)\ a$ приводит к значению $a+2y$, а вызов $(\lambda y. x+2y)\ a$ - к значению $x+2a$.

Запись $\lambda y. \lambda x. E$ понимается как $\lambda y. (\lambda x. E)$. Поэтому $((\lambda y. \lambda x. x+2y)\ a)\ b \rightarrow (\lambda x. x+2a)\ b \rightarrow b+2a$ и $((\lambda x. \lambda y. x+2y)\ a)\ b \rightarrow (\lambda y. a+2y)\ b \rightarrow a+2b$.

Мы читаем символ λ как “функция от” и точку $(.)$ как “которая возвращает”.

Определение λ -термов (λ -выражений)

- Каждая переменная есть λ -терм.
- Каждая константа есть λ -терм.
- По любым λ -термам M и N можно построить новый λ -терм (MN) (обозначающий применение, или *аппликацию*, оператора M к аргументу N).

По любой переменной x и любому λ -терму M можно построить новый λ -терм $(\lambda x. M)$ (обозначающий функцию от x , определяемую λ -термом M). Такая конструкция называется *λ -абстракция*.

Набор констант произволен: целые числа, булевы константы, арифметические операции, булевы функции и т. п., причем для записи применения константы-оператора к операндам используется префиксная запись (так $+\ 3\ 4$ обозначает $3+4$).

Символ x после λ называется связанной переменной абстракции и соответствует понятию формального параметра в традиционной процедуре или функции. Выражение справа от точки называется телом абстракции, и, подобно коду традиционной процедуры или функции, оно описывает, что нужно сделать с параметром, поступившим на вход функции.

Переменная, расположенная не на месте связанной переменной, может ли быть *связанной* или *свободной*, что определяется с помощью следующих правил:

1. Переменная x оказывается свободной в выражении x .
2. Все x , имеющиеся в $\lambda x. M$, являются связанными. Если кроме x в $\lambda x. M$ есть переменная y , то последняя будет свободной или связанной в зависимости от того, свободно она или связана в M .
3. Переменная встречающаяся в термах M или N выражения (MN) будет связанной или свободной в общем терме в зависимости от того, свободна она или связана в M или N . Свободные (связанные) переменные - это переменные, которые, по крайней мере, один раз появляются в терме в свободном (связанном) виде.

Мы отождествляем термы, отличающиеся только названием своих связанных переменных, например $\lambda x. x \equiv \lambda y. y$.

Тело абстракции может быть любым допустимым λ -выражением, и поэтому оно также может содержать другую абстракцию, например:

$$\lambda x . \lambda y . (x+y)*2$$

Это выражение читается как “функция от x , которая возвращает функцию от y , которая возвращает $(x+y)*2$ ”.

Используется соглашение: запись $(\dots(((\lambda x_1.\lambda x_2.\dots\lambda x_n.E)a_1)a_2)\dots a_n)$ кратко пишется как $(\lambda x_1.\lambda x_2.\dots\lambda x_n.E)a_1a_2\dots a_n$.

Нам понадобится следующее определение *подстановки*. Для любых λ -термов M , N и переменной x через $[N/x]M$ обозначим результат подстановки N вместо каждого свободного вхождения x в M и замены всех λy в M таким образом, чтобы свободные переменные из N не стали связанными в $[N/x]M$. Более формально (мы употребляем запись $M \equiv N$ для обозначения того, что термы M и N совпадают):

- a) $[N/x]x \equiv N$;
- b) $[N/x]y \equiv y$, если переменная y не совпадает с x ;
- c) $[N/x](PQ) \equiv ([N/x]P [N/x]Q)$;
- d) $[N/x](\lambda x.P) \equiv \lambda x.P$;
- e) $[N/x](\lambda y.P) \equiv \lambda y.[N/x]P$, если y не имеет свободных вхождений в N и x имеет свободное вхождение в P ;
- f) $[N/x](\lambda y.P) \equiv \lambda z.[N/x]([z/y]P)$, если y имеет свободное вхождение в N и x имеет свободное вхождение в P и z - любая переменная, не имеющая свободных вхождений в N ;
- g) $[N/x]t \equiv t$, если t является константой.

Следующие примеры пояснят суть определения. Пусть $M \equiv \lambda y.ux$.

Если $N \equiv vx$, то $[(vx)/x](\lambda y.ux) \equiv \lambda y. [(vx)/x](yx)$ согласно (e)

$$\equiv \lambda y. y(vx) \text{ согласно (a).}$$

Если $N \equiv ux$, то $[(ux)/x](\lambda y.ux) \equiv \lambda z. [(ux)/x](zx)$ согласно (f)

$$\equiv \lambda z.z(ux) \text{ согласно (a).}$$

Если бы пункт (f) в определении был опущен, то мы столкнулись бы со следующим нежелательным явлением. Хотя $\lambda v.x$ и $\lambda u.x$ обозначают одну и ту же функцию (константу, чье значение всегда есть x), после подстановки v вместо x они стали бы обозначать разные функции: $[v/x](\lambda u.x) \equiv \lambda u.v$, $[v/x](\lambda v.x) \equiv \lambda v.v$.

Мы рассмотрели, как λ -нотация может быть использована для представления функциональных выражений и сейчас готовы к тому, чтобы определить *правила вывода* λ -исчисления, которые описывают, как вычислять выражение, т. е. как получать конечное значение выражения из его первоначального вида.

Константы являются самоопределенными, т. е. их невозможно преобразовать в более простые выражения. Если константа, обозначающая функцию (оператор), применяется к соответствующему числу операндов,

то такой подтерм называется δ -редексом, процесс применения функции называется δ -сворачиванием, и в результате появляется новое лямбда-выражение.

Применение константной функции записывается в виде встроенных соответствующих δ -правил, например

$$+ 1 3 \rightarrow_{\delta} 4$$

Терм вида $(\lambda x.M)N$ называется β -редексом. (Он также обозначает применение оператора к входному значению.) Если β -редекс содержится в терме P и одно из его вхождений заменяется термом $[N/x]M$, то мы будем говорить, что происходит *свертывание* этого вхождения. Обозначение $P \rightarrow_{\beta} Q$ означает, что P β -сворачивается к Q . Конечная последовательность δ - или β -свертываний называется *редукцией*. Если из контекста ясно, было ли δ -свертывание или же β -свертывание, то один шаг редукции обозначается просто знаком \rightarrow без индекса.

Пример 6.1. Редукция выражений (используемые редексы подчеркнуты):

$$1) (\lambda f. \lambda x. f 3x) (\lambda y. \lambda x. * x y) 0$$

$$\rightarrow (\lambda x. (\lambda y. \lambda x. * xy) 3x) 0 \text{ - выбрали один из двух возможных редексов}$$

$$\rightarrow (\lambda y. \lambda x. * xy) 3 0$$

$$\rightarrow (\lambda x. * x 3) 0$$

$$\rightarrow * 0 3$$

$$\rightarrow 0$$

$$2) (\lambda f. \lambda x. f 3x) (\lambda y. \lambda x. * xy) 0$$

$$\rightarrow (\lambda x. (\lambda y. \lambda x. * xy) 3 x) 0 \text{ - выбрали один из двух возможных редексов}$$

$$\rightarrow (\lambda x. (\lambda x. * x 3) x) 0 \text{ - снова делаем произвольный выбор}$$

$$\rightarrow (\lambda x. * x 3) 0$$

$$\rightarrow * 0 3$$

$$\rightarrow 0$$

Использование констант и δ -правил является излишним. Все необходимые константы (числа, встроенные функции и т. п.) можно реализовать, используя в качестве атомарных термов только переменные (так называемое чистое λ -исчисление).

Нормальные формы

Говорят, что λ -выражение находится в *нормальной форме*, если к нему нельзя применить никакое правило редукции. Другими словами, λ -выражение - в нормальной форме, если оно не содержит редексов. Нормальная форма, таким образом, соответствует понятию конца вычислений в традиционном программировании. Отсюда немедленно вытекает наивная схема вычислений:

```
while существует хотя бы один редекс
do преобразовывать один из редексов
```

end

(выражение теперь в нормальной форме).

Различные варианты сворачивания в процессе редукции могут приводить к принципиально различным последствиям.

Пример 6.2.

$(\lambda x. \lambda y. y) ((\lambda z. z z) (\lambda z. z z))$
 $\rightarrow (\lambda x. \lambda y. y) ((\lambda z. z z) (\lambda z. z z))$
 $\rightarrow (\lambda x. \lambda y. y) ((\lambda z. z z) (\lambda z. z z))$
 $\rightarrow \dots$
 (бесконечный процесс редукции)

$(\lambda x. \lambda y. y) ((\lambda z. z z) (\lambda z. z z))$
 $\rightarrow \lambda y. y$
 (редукция закончилась)

Порядок редукций (стратегия выбора редексов)

Самым левым редексом называется редекс, символ λ которого (или идентификатор примитивной функции в случае δ -редекса) текстуально расположен в λ -выражении левее всех остальных редексов. (Аналогично определяется *самый правый редекс*.)

Самым внешним редексом называется редекс, который не содержится внутри никакого другого редекса.

Самым внутренним редексом называется редекс, не содержащий других редексов.

В контексте функциональных языков и λ -исчисления существуют два важных порядка редукций [32].

Аппликативный порядок редукций (АПР), который предписывает вначале преобразовывать самый левый из самых внутренних редексов.

Нормальный порядок редукций (НПР), который предписывает вначале преобразовывать самый левый из самых внешних редексов.

$(\lambda x. \lambda y. y) ((\lambda z. z z) (\lambda z. z z))$ - самый левый из самых внутренних редексов - вычисление никогда не закончится.

$(\lambda x. \lambda y. y) ((\lambda z. z z) (\lambda z. z z))$ - самый левый из самых внешних редексов - вычисление закончится за один шаг.

Функция $\lambda x. \lambda y. y$ - это классический пример функции, которая отбрасывает свой аргумент. НПР в таких случаях эффективно откладывает вычисление любых редексов внутри выражения аргумента до тех пор, пока это возможно, в расчете на то, что такое вычисление может оказаться ненужным.

В функциональных языках стратегии НПР соответствуют так называемые *ленивые вычисления*. “Не делай ничего, пока это не потребуется” -

механизм вызова по необходимости (все аргументы передаются функции в не вычисленном виде и вычисляются только тогда, когда в них возникает необходимость внутри тела функции). Clean - один из языков с ленивыми вычислениями.

Стратегия АПР приводит к *энергичным вычислениям*. “Делай все, что можешь”; другими словами, не надо заботиться о том, пригодится ли, в конечном счете, полученный результат. Таким образом, реализуется механизм вызова по значению (значение аргумента передается в тело функции). В языке Лисп реализуются, как правило, энергичные вычисления.

Следствие из теоремы Чёрча-Россера [7, с.74]. Если выражение E может быть приведено двумя различными способами к двум нормальным формам, то эти нормальные формы совпадают или могут быть получены одна из другой с помощью замены связанных переменных.

Теорема стандартизации [7, с. 298-303]. Если выражение E имеет нормальную форму, то НПР гарантирует достижение этой нормальной формы (с точностью до замены связанных переменных).

Рекурсивные функции

Поскольку в λ -исчислении все функции анонимны, то для представления рекурсивных функций с помощью лямбда-термов необходимо придумать метод, позволяющий функциям вызывать себя не по имени, а каким-то другим образом.

Представим рекурсивную функцию как функцию, имеющую себя в качестве аргумента. В этом случае функция может оказаться связанной с одной из её собственных переменных и будет, таким образом, содержать в своем теле ссылки на самое себя.

Рассмотрим, например, рекурсивную функцию $\text{sum}(x)$, определенную для сложения все целых чисел от 1 до n

$$\text{sum}(n) = (\text{IF } (= n 0) 0 (+ n (\text{sum } (1- n))))).$$

В этой записи мы используем пять констант: число 0, булевские функцию для равенства (=), функции для условного выражения (IF), для суммы (+) и для вычитания 1 (1-).

Это выражение может быть представлено в виде λ -абстракции, имеющей дополнительный параметр, который при применении этой абстракции связывается с самой функцией. Мы запишем эту промежуточную версию функции sum :

$$\text{sum} = \lambda s . \lambda n . \text{IF } (= 0 n) 0 (+ n (s (1- n)))$$

Все, что нам осталось сделать теперь, - это связать переменную s со значением функции sum , которую пытаемся определить. Это можно сделать, используя специальную функцию, называемую *Y-комбинатором*, которая удовлетворяет следующему уравнению:

$$Y f = f (Y f)$$

Y также известен как *комбинатор неподвижной точки*. “Неподвижная точка” функции f - это выражение, которое не изменяется при примене-

нии к нему функции f . (Заметим, что функция может иметь несколько неподвижных точек. Например, функция тождества $\lambda x . x$, имеет бесконечное их число.) Выражение $Y f$ дает наименьшую неподвижную точку функции f .

$$\begin{aligned} & Y \text{ sum} \\ &= Y(\lambda s . \lambda n . \text{IF } (= 0 n) 0 (+ n (s (1- n)))) \\ &\rightarrow (\lambda s . \lambda n . \text{IF } (= 0 n) 0 (+ n (s (1- n)))) (Y \text{ sum}) \\ &\rightarrow \lambda n . \text{IF } (= 0 n) 0 (+ n ((Y \text{ sum}) (1- n))) \\ &\rightarrow \lambda n . \text{IF } (= 0 n) 0 \\ &\quad (+ n ((\lambda m . \text{IF } (= 0 m) 0 (+ m ((Y \text{ sum}) (1- m)))) \\ &\quad (1- n)))) \end{aligned}$$

Данное выражение ведет себя точно так же, как исходное рекурсивное определение sum . Внутреннее вхождение $Y \text{ sum}$ конструирует копию исходной функции sum , помещая само себя (т. е. $Y \text{ sum}$) вместо s в тело копии.

Таким образом, функция sum выражается в λ -исчислении в виде $Y(\lambda s . \lambda n . \text{IF } (= 0 n) 0 (+ n (s (1- n))))$

Проверим:

$$\begin{aligned} & (Y(\lambda s . \lambda n . \text{IF } (= 0 n) 0 (+ n (s (1- n)))) 1 \\ &\rightarrow \lambda n . \text{IF } (= 0 n) 0 (+ n ((Y(\lambda s . \lambda n . \text{IF } (= 0 n) 0 (+ n (s (1- n)))) \\ &\quad (1- n))) 1 \\ &\rightarrow \text{IF } (= 0 1) 0 (+ 1 ((Y(\lambda s . \lambda n . \text{IF } (= 0 n) 0 (+ n (s (1- n)))) \\ &\quad (1- 1))) \\ &\rightarrow (+ 1 ((Y(\lambda s . \lambda n . \text{IF } (= 0 n) 0 (+ n (s (1- n)))) 0)) \\ &\rightarrow (+ 1 (\lambda n . \text{IF } (= 0 n) 0 \\ &\quad (+ n ((Y(\lambda s . \lambda n . \dots)))) 0) \\ &\rightarrow (+ 1 (\text{IF } (= 0 0) 0 (+ 0 ((Y(\lambda s . \lambda n . \dots)))))) \\ &\rightarrow (+ 1 0) \\ &\rightarrow 1 \end{aligned}$$

В общем случае рекурсивная функция f с телом, задаваемым выражением E , записывается в λ -исчислении в виде $Y (\lambda f . E)$.

Определим комбинатор неподвижной точки Y следующим образом:

$$Y = \lambda h . (\lambda x . h (x x)) (\lambda x . h (x x))$$

Проверим:

$$\begin{aligned} & Yf = \\ & (\lambda h . (\lambda x . h (x x)) (\lambda x . h (x x))) f \\ &\rightarrow (\lambda x . f (x x)) (\lambda x . f (x x)) \\ &\rightarrow f ((\lambda x . f (x x)) (\lambda x . f (x x))) \\ &\rightarrow f (Y f) \end{aligned}$$

Чистое λ -исчисление

Удалив константы и δ -правила, мы получаем чистое λ -исчисление. В нем можно выразить любые константы и функции. Например, булевы константы и условное выражение можно представить с помощью термов:

$$\text{TRUE} = \lambda x . \lambda y . x$$

$$\text{FALSE} = \lambda x . \lambda y . y$$

$$\text{IF} = \lambda p . \lambda q . \lambda r . p \ q \ r$$

Легко проверить, что выполняются следующие редукции:

$$\text{TRUE } A \ B \rightarrow A$$

$$\text{FALSE } A \ B \rightarrow B$$

$$\text{IF } \text{TRUE } A \ B =$$

$$(\lambda p . \lambda q . \lambda r . p \ q \ r) (\lambda x . \lambda y . x) \ A \ B$$

$$\rightarrow (\lambda q . \lambda r . (\lambda x . \lambda y . x) \ q \ r) \ A \ B$$

$$\rightarrow (\lambda q . \lambda r . (\lambda y . q) \ r) \ A \ B$$

$$\rightarrow (\lambda q . \lambda r . q) \ A \ B$$

$$\rightarrow (\lambda r . A) \ B$$

$$\rightarrow A$$

Нумерация Чёрча. Чёрч предложил натуральное число n представлять термом n -кратной композиции (обозначение $x^n(y)$ служит сокращением для $x(x(\dots(xy)\dots))$, x повторяется n раз).

Для каждого натурального числа n положим

$$\langle 0 \rangle = \lambda x . \lambda y . y, \quad \langle n \rangle = \lambda x . \lambda y . x^n(y)$$

Тогда сложение чисел определяется лямбда-выражением

$$+ \ x \ y = \lambda p . \lambda q . x \ p \ (y \ p \ q)$$

Проверим, что это определение удачно.

$$+ \ 1 \ 1 = \lambda p . \lambda q . \langle 1 \rangle \ p \ (\langle 1 \rangle \ p \ q)$$

$$= \lambda p . (\lambda q . ((\lambda x . \lambda y . x \ y) \ p \ ((\lambda x . \lambda y . x \ y) \ p \ q)))$$

$$\rightarrow \lambda p . (\lambda q . ((\lambda x . \lambda y . x \ y) \ p \ p \ q))$$

$$\rightarrow \lambda p . (\lambda q . (p \ p \ q))$$

$$= \langle 2 \rangle$$

Говорят, что частичная функция φ с k аргументами λ -определима термом M , когда

$M\langle n_1 \rangle \langle n_2 \rangle \dots \langle n_k \rangle$ β -редуцируется к терму $\langle \varphi(n_1, n_2, \dots, n_k) \rangle$, если значение $\varphi(n_1, n_2, \dots, n_k)$ определено, и

$M\langle n_1 \rangle \langle n_2 \rangle \dots \langle n_k \rangle$ не имеет нормальной формы, если $\varphi(n_1, n_2, \dots, n_k)$ не определено.

Теорема Клини [7, с. 189]. Частичная функция частично-рекурсивна тогда и только тогда, когда она λ -определима.

§4 Машины Тьюринга

Рассмотрим еще один способ определения вычислимых функций, следуя в изложении [29, стр. 12-14]. Формулировка, выраженная в терминах воображаемой вычислительной машины, была дана английским математиком Аланом Тьюрингом в 1936 г. [4]. Главная трудность при нахождении этого определения была в том, что Тьюринг искал его до создания реальных цифровых вычислительных машин. Познание шло от абстрактного к конкретному: фон Нейман был знаком с работой Тьюринга, и сам Тьюринг позднее сыграл вдохновляющую роль в развитии вычислительных машин.



Алан Тьюринг

На неформальном уровне мы можем описывать машину Тьюринга как некий черный ящик с лентой. Лента разбита на ячейки и каждая ячейка может содержать пустой символ 0, либо непустой символ 1. Лента потенциально бесконечна в обе стороны в том смысле, что мы никогда не приходим к ее концу, но в любое время лишь конечное число ячеек может быть непустым. В начале лента содержит числа входа, в конце - число-выход. В промежуточное время лента используется как пространство памяти для вычисления.

Если мы откроем черный ящик, то обнаружим, что он устроен очень просто. В любой момент времени он может обозревать лишь одну ячейку памяти. Устройство содержит конечный список инструкций (или *состояний*) q_0, q_1, \dots, q_n . Каждая инструкция может указать два возможных направлений действий; одного нужно придерживаться, если на обозреваемой ячейке ленты находится 0, а другого, - если там находится 1. В любом случае следующее действие может состоять из таких трех типов элементарных шагов:

- 1) символ (возможно, такой же, как старый) пишется на обозреваемой ячейке ленты, при этом предыдущий символ стирается;
- 2) лента сдвигается на одну ячейку влево или вправо;
- 3) указывается следующая инструкция.

Таким образом, список инструкций определяет некоторую функцию перехода, которая по данной инструкции и обозреваемому символу указывает три компоненты того, что нужно делать. Мы можем формализовать эти идеи, взяв в качестве машины Тьюринга эту функцию перехода.

Определение. *Машина Тьюринга* - это функция M такая, что для некоторого натурального числа n , область определения этой функции есть подмножество множества $\{0, 1, \dots, n\} \times \{0, 1\}$, а область значений есть подмножество множества $\{0, 1\} \times \{L, P\} \times \{0, 1, \dots, n\}$.

Например, пусть $M(3,1) = \langle 0, Л, 2 \rangle$. Подразумеваемый смысл этого состоит в том, что как только машина дойдет до инструкции q_3 , а на обозреваемой ячейке написан символ 1, она должна стереть 1 (оставляя на ячейке 0), передвинуть ленту так, чтобы обозреваемая ячейка стала левая соседняя ячейка от той, которая обозревалась, и перейти к следующей инструкции q_2 . Если $M(3,1)$ не определено, то как только машина дойдет до инструкции q_3 , а на обозреваемой ячейке написан символ 1, то машина останавливается. (Это единственный путь остановки вычисления.)

Такая подразумеваемая интерпретация не включена в формальное определение машины Тьюринга, но она мотивирует и подсказывает формулировки всех следующих определений. В частности, можно определить, что означает для машины M передвижение (за один шаг) от одной конфигурации до другой. Нам не нужно здесь давать формальных определений, так как они являются простыми переводами наших неформальных идей.

Входные и выходные данные - это строки из 1, разделенные 0. Пусть $\langle n \rangle$ будет строкой из 1 длины $n+1$. Тогда

$$\langle n_1 \rangle 0 \langle n_2 \rangle 0 \dots 0 \langle n_k \rangle$$

получено комбинацией k строчек из 1, каждая отделена от другой 0.

Наконец, мы можем определить вычислимость.

Определение. Пусть $D_f \subseteq N^k$ - область определения k -местной функции $f: D_f \rightarrow N$ (N - множество натуральных чисел). Функция f называется *вычислимой*, если существует машина Тьюринга M такая, что как только M начинает с инструкции q_0 , обозревая самый левый символ строки

$$\langle n_1 \rangle 0 \langle n_2 \rangle 0 \dots 0 \langle n_k \rangle,$$

(вся остальная часть ленты пуста), тогда:

- если $f(n_1, n_2, \dots, n_k)$ определено, то M в конце концов остановится, обозревая самый левый символ строки $\langle f(n_1, n_2, \dots, n_k) \rangle$, при этом часть, находящаяся справа от этой строчки, пустая;
- если $f(n_1, n_2, \dots, n_k)$ не определено, то M никогда не останавливается.

Заметим, что имеется бесконечное множество машин Тьюринга, для каждой вычислимой функции своя. Более того, для любой вычислимой функции имеется бесконечное множество машин Тьюринга, вычисляющих эту функцию.

Пример 6.3. Построим машину Тьюринга, вычисляющую сумму $n_1 + n_2$.

Зададим функцию M следующим образом:

$$\begin{aligned} M(0, 1) &= \langle 1, П, 0 \rangle; \\ M(0, 0) &= \langle 1, П, 1 \rangle; \\ M(1, 1) &= \langle 1, П, 1 \rangle; \\ M(1, 0) &= \langle 0, Л, 2 \rangle; \\ M(2, 1) &= \langle 0, Л, 3 \rangle; \\ M(3, 1) &= \langle 0, Л, 4 \rangle; \end{aligned}$$

$$M(4, 1) = \langle 1, Л, 4 \rangle;$$

$$M(4, 0) = \langle 0, П, 5 \rangle.$$

Посмотрим как происходит сложение $1+1$. В текущей строке символов обозреваемый символ выделен.

Номер инструкции	Текущая строка символов	Комментарий
0	0 I 10110	прохождение через первое слагаемое
0	01 I 0110	
0	011 0 110	заполнение промежутка
1	0111 I 10	прохождение через второе слагаемое
1	01111 I 0	
1	011111 0	конец второго слагаемого
2	01111 I 0	стирание 1
3	01111 I 00	стирание второй 1
4	011 I 000	движение назад
4	01 I 1000	
4	0 I 11000	
4	0 111000	остановка
5	0 I 11000	

Мы должны заметить, что многие детали нашего определения машины Тьюринга до некоторой степени произвольны. Если бы было более одной ленты, то класс вычислимых функций остался бы неизменным, хотя некоторые функции могли бы быть вычислены более быстро. Аналогично, мы могли бы допускать больше символов, чем 0 и 1, или же у нас могла бы быть лента бесконечная только в одну сторону от начальной точки вместо имеющихся бесконечной в обоих направлениях. Ни одно из этих изменений не затрагивает класса вычислимых функций. Что действительно существенно в этом определении - это разрешение произвольно большого количества материала для запоминающего устройства и произвольно длинных вычислений.

§5 Тезис Чёрча

За последние 60 лет было предложено много различных математических уточнений интуитивного понятия алгоритма. Три из этих подхода мы разобрали. Перечислим некоторые другие альтернативные способы, которые предполагались следующими авторами:

- Гёдель-Эбран-Клини. Общерекурсивные функции, определенные с помощью исчисления рекурсивных уравнений [23, с. 261-278].

- Пост. Функции, определяемые каноническими дедуктивными системами [13, с. 66-72].

- Марков. Функции, задаваемые некоторыми алгоритмами (известные под названием нормальные алгоритмы) над конечным алфавитом [23, с.228-250].

- Шепердсон-Стерджис. МНР-вычислимые функции [13].

Между этими подходами (в том числе, и три выше рассмотренных) имеются большие различия; каждый из них имеет свои преимущества для соответствующего описания вычислимости. Следующий замечательный результат получен усилиями многих исследователей.

Основной результат [13, с. 57]. Каждое из вышеупомянутых уточнений эффективной вычислимости приводит к одному и тому же классу вычислимых функций.

Вопрос: насколько хорошо неформальное и интуитивное понятие эффективно вычислимой функции отражено в различных формальных описаниях?

Чёрч, Тьюринг и Марков каждый в соответствии со своим подходом выдвинули утверждение (тезис) о том, что класс определенных ими функций совпадает с неформально определенным классом вычислимых функций. В силу основного результата все эти утверждения логически эквивалентны. Название *тезис Чёрча* теперь применяется к этим и аналогичным им утверждениям.

Тезис Чёрча. Интуитивно и неформально определенный класс эффективно вычислимых функций совпадает с классом λ -определимых функций.



Андрей Андреевич
Марков

Сразу же заметим, что этот тезис не является теоремой, а скорее утверждение, которое принимается на веру, причем вера подкрепляется следующими аргументами [13, с. 75-76].

- Фундаментальный результат: многие независимые варианты уточнения интуитивного понятия вычислимости привели к одному и тому же классу функций.

- Обширное семейство эффективно вычислимых функций принадлежит этому классу. Конкретные функции, рассмотренные в 6.2, образуют исходную часть этого семейства, которую можно расширять до бесконечности методами из 6.2 или более мощными и сложными методами.

- Никто еще не нашел функцию, которую можно было бы признать вычислимой в неформальном смысле, но которую нельзя было бы построить, используя один из формальных методов.

... Найти задачу - не меньшая радость, чем отыскать решение.

Томас де Куинси

– Это же проблема Бен Бецалеля. Калиостро же доказал, что она не имеет решения.... Как же искать решения, когда его нет? Бессмыслица какая-то...

– Бессмыслица - искать решение, если оно и так есть. Речь идет о том, как поступать с задачей, которая решения не имеет.

А. и Б. Стругацкие

"Понедельник начинается в субботу"

§6 Некоторые алгоритмически неразрешимые проблемы

Определение. Предикат $M(x)$ называется *разрешимым*, если его характеристическая функция, задаваемая формулой

$c_M(x) = 1$, если $M(x)$ истинно;

$c_M(x) = 0$, если $M(x)$ ложно

вычислима.

Определение. Предикат $M(x)$ называется *неразрешимым*, если он не является разрешимым. В контексте разрешимости предикаты часто называются *проблемами*.

Имея точное определение вычислимости, удалось доказать, что некоторые проблемы неразрешимы.

- Теорема Черча о неразрешимости логики предикатов. Не существует алгоритма, который для любой формулы логики предикатов устанавливает, общезначима она или нет.

- Проблема остановки неразрешима [13, с. 108]. Не существует никакого общего алгоритма, позволяющего установить, остановится ли некоторая конкретная программа (на любом языке программирования), запущенная после введения в неё некоторого конкретного набора данных. Смысл этого утверждения для теоретического программирования очевиден: *не существует совершенно общего метода проверки программ на наличие в них бесконечных циклов*. В терминах лямбда-исчисления утверждение о неразрешимости проблемы остановки можно сформулировать в таком виде: *"Не существует алгоритма, с помощью которого можно было бы узнать имеет ли данное лямбда-выражение нормальную форму или нет"*.

- Не существует никакого общего алгоритма, позволяющего установить, вычисляет ли некоторая конкретная программа (на любом языке программирования) постоянную нулевую функцию [13, с. 110]. То же самое справедливо и для любой другой конкретной вычислимой функции. И как следствие, можно утверждать, что вопрос о том, вычисляют ли две данные программы одну и ту же одноместную функцию, также неразрешим. Тем самым, мы получаем, что в области тестирования компьютерных программ, мы имеем принципиальные ограничения.

Диофантовы уравнения [13, с. 114]. Современная математика вообще изобилует разрешимыми и неразрешимыми проблемами. Одна из проблем связана с диофантовыми уравнениями.

Пусть $p(x_1, x_2, \dots, x_n)$ - многочлен от переменных x_1, x_2, \dots, x_n с целыми коэффициентами. Тогда уравнение

$$p(x_1, x_2, \dots, x_n) = 0,$$

для которого мы ищем только целые решения называется диофантовым уравнением. Диофантовы уравнения не обязательно имеют решения. Например, не имеет решения уравнение $x^2 - 2 = 0$.

Десятая проблема Гильберта, сформулированная в 1900 году, состоит в том, чтобы установить, существует ли алгоритм, с помощью которого можно было бы проверить, имеет ли данное диофантово уравнение решение. В 1970 году советский математик Ю. Матиясевич доказал, что такого алгоритма не существует. Доступное доказательство этого можно найти в [21].

Отметим также, что знаменитую теорему Гёделя о неполноте можно легко доказать, используя теорию алгоритмов. Элементарное доказательство этого приведено в [31]. Занимательному изложению вопросов вычислимости, вплоть до получения доказательства теоремы Гёделя, посвящена книга Р. Смаллиана [28].

§7 Сложность алгоритмов

Применение математики во многих приложениях, требует как правило, использования различных алгоритмов. Для решения многих задач не трудно придумать комбинаторные алгоритмы, сводящиеся к полному перебору вариантов. Но здесь вступает в силу различие между математикой и информатикой: в информатике недостаточно высказать утверждение о существовании некоторого объекта в теории и даже недостаточно найти конструктивное доказательство этого факта, т.е. алгоритм. Мы должны учитывать ограничения, навязываемые нам миром, в котором мы живем: не-

обходимо, чтобы решение можно было вычислить, используя объем памяти и время, приемлемые для человека и компьютера.

Основные понятия

Класс однородных вычислительных задач мы будем называть *проблемой* (также используется понятие *массовая задача*). Индивидуальные случаи проблемы T мы будем называть *частными случаями* проблемы T . Таким образом, T есть множество всех своих частных случаев. Такое описание проблемы есть только предмет соглашения и удобство обозначений. Мы можем, например, говорить о проблеме умножения матриц. Частные случаи этой проблемы суть (для любого целого n -размера квадратных матриц) пары матриц, которые нужно перемножить.

В качестве другого примера рассмотрим классическую задачу о коммивояжере. Параметры этой массовой задачи состоят из конечного набора "городов" $C = \{c_1, c_2, \dots, c_m\}$ и "расстояний" $d(c_i, c_j)$ между каждой парой городов c_i, c_j из C . Решение - это такой упорядоченный набор $\langle c_{k(1)}, c_{k(2)}, \dots, c_{k(m)} \rangle$ заданных городов, который минимизирует величину

$$\sum_{i=1}^{m-1} d(c_{k(i)}, c_{k(i+1)}) + d(c_{k(m)}, c_{k(1)}).$$

Это выражение дает длину маршрута, начинающегося в городе $c_{k(1)}$, проходящего последовательно через все города и возвращающегося в $c_{k(1)}$ непосредственно из последнего города $c_{k(m)}$. Индивидуальная задача о коммивояжере задается любыми конкретными множествами $\{c_1, c_2, \dots, c_m\}$ и $\{d(c_i, c_j)\}$.

С каждым частным случаем проблемы $I \in T$ мы связываем размер $|I|$ (обычно целое число). Эта функция $|I|$ не единственна, и ее выбор диктуется теоретическими и практическими соображениями, связанными с тем, с какой точки зрения интересна эта проблема.

Возвратимся к примеру умножения матриц, отметим, что разумной мерой для пары $I=(A, B)$ ($n \times n$)-матриц, которые надо перемножить, является $|I|=n$. Если мы изучаем объем памяти, требующейся для алгоритма умножения матриц, то подходящей может быть мера $|I|=n^2$. В задаче о коммивояжере $|I|$ можно определить как количество данных городов m .

Пусть T - проблема и A - алгоритм, решающий ее. При решении частного случая $I \in T$ алгоритм A выполняет некоторую последовательность вычислений S_I . С S_I мы связываем некоторые числовые характеристики.

Существенными являются, например, следующие характеристики:

- длина S_I , которая характеризует время вычисления;
- глубина S_I , т. е. число уровней параллельных шагов, на которые S_I может быть разложено; она соответствует времени, которое S_I потребовалось бы при параллельных вычислениях;

- объемом памяти, требуемый для вычисления S_I ;
- вместо общего числа шагов в S_I мы можем подсчитывать число шагов некоторого вида, таких как арифметические операции при алгебраических вычислениях, число сравнений при сортировке или число обращений к памяти.

Для аппаратной реализации алгоритмов мы обычно определяем размер $|I|$, так, чтобы все частные случаи I одинакового размера n решались при помощи одной и той же схемы C_n . Сложность схемы C определяется разными способами, например, как число элементов, глубина, снова связанная со временем вычислений, или выбираются другие меры сложности, такие как число модулей, связанное с технологией, используемой при построении схемы.

После того как выбрана мера μ вычисления S функция F_A сложности вычисления может быть определена несколькими способами, два главных из них - сложность в *наихудшем случае* и сложность *поведения в среднем*.

Первое понятие определяется следующим образом:

$$F_A(n) = \max \{ \mu(S_I) \mid I \in T, |I| = n \}.$$

Для того чтобы определить поведение в среднем, мы задаем распределение вероятностей T на каждом множестве $T_n = \{I \mid I \in T, |I| = n\}$. Так, для $I \in T$, $|I| = n$, величины $p(I)$ - вероятность появления I среди всех других частных случаев размера n . *Поведение в среднем* алгоритма A тогда определяется так:

$$M_A(n) = \sum_{I \in P_n} p(I) \mu(S_I).$$

Анализ алгоритмов связан со следующим вопросом. Для заданной функции размера $|I|$ и меры вычисления $\mu(S_I)$ точно определить для данного алгоритма A , решающего проблему T , либо сложность F_A для наихудшего случая, либо, при подходящих предположениях, поведение в среднем M_A . Вопросы анализа алгоритмов подробно рассматриваются в трехтомнике Д. Кнута "Искусство программирования для ЭВМ" [15].

Сложность задачи - это сложность наилучшего алгоритма, известного для ее решения.

Для оценок сложности потребуется следующее определение. Будем говорить, что функция $f(n)$ есть $O(g(n))$, если существует константа C такая, что $|f(n)| \leq C(g(n))$ для всех натуральных n .

Основной вопрос теории сложности: насколько успешно или с какой стоимостью может быть решена заданная проблема T ? Мы не имеем в виду никакого конкретного алгоритма решения T . Наша цель - рассмотреть все возможные алгоритмы решения T и попытаться сформулировать утверждение о вычислительной сложности, внутренне присущей T . В то время как всякий алгоритм A для T дает верхнюю оценку величины F_A сложности T , нас интересует нижняя оценка. Знание нижней оценки пред-

ставляет интерес математический и кроме того, руководит нами в поиске хороших алгоритмов, указывая, какие попытки заведомо будут безуспешны.

Классификация задач по степени сложности

Быстрыми являются линейные алгоритмы, которые обладают сложностью порядка n и называются также алгоритмами порядка $O(n)$, где n - размерность входных данных. К линейным алгоритмам относится школьный алгоритм нахождения суммы десятичных чисел, состоящих из n_1 и n_2 цифр. Сложность этого алгоритма - $O(n_1 + n_2)$. Есть алгоритмы, которые быстрее линейных, например, алгоритм двоичного поиска в линейном упорядоченном массиве имеет сложность $O(\log_2 n)$, n - длина массива.

Другие хорошо известные алгоритмы - деление, извлечение квадратного корня, решение систем линейных уравнений и др. - попадают в более общий класс полиномиальных алгоритмов.

Полиномиальным алгоритмом (или алгоритмом полиномиальной временной сложности, или алгоритмом принадлежащим классу **P**) называется алгоритм, у которого временная сложность равна $O(n^k)$, где k - целое число > 0 . Алгоритмы, для временной сложности которых не существует такой оценки, называются *экспоненциальными*.

Задача считается *труднорешаемой*, если для него не существует разрешающего полиномиального алгоритма.

Заметим, что при небольших значениях n экспоненциальный алгоритм может быть более быстрым, чем полиномиальный (почему?). Однако различие между этими двумя типами задач велико и всегда проявляется при больших значениях n .

Класс P

Мы называем задачу "хорошей" если для нее существует полиномиальный алгоритм. Приведем список некоторых хорошо решаемых задач.

- Рассортировать множество из n чисел. Сложность поведения в среднем порядка $O(n \log n)$ для быстрого алгоритма Хоара [26, стр.316-321].
- Найти эйлеровый цикл на графе из m ребер. В силу теоремы Эйлера мы имеем необходимое и достаточное условие для существования эйлерова цикла и проверка этого условия есть алгоритм порядка $O(m)$ [24, стр. 200-201].
- Задача Прима-Краскала. *Дана плоская страна и в ней n городов. Нужно соединить все города телефонной связью так, чтобы общая длина телефонных линий была минимальной.* В терминах теории графов задача Прима-Краскала выглядит следующим образом: *Дан граф с n вершинами; длины ребер заданы матрицей $(d[i,j])$, $i,j = 1,...,n$. Найти остовное дерево*

минимальной длины. Эта задача решается с помощью жадного алгоритма сложности $O(n \log n)$ [26, стр.357-358].

- Кратчайший путь на графе, состоящем из n вершин и m ребер. Сложность алгоритма $O(m \log n)$ [26, стр.377-382].

- Связные компоненты графа. Определяются подмножества вершин в графе (связные компоненты), такие, что две вершины, принадлежащие одной и той же компоненте, всегда связаны цепочкой дуг. Если n - количество вершин, а m - количество ребер, то сложность алгоритма $O(n+m)$ [26, стр.364-365].

- Быстрое преобразование Фурье [6, стр. 284-302], требующее $O(n \log n)$ арифметических операций, - один из наиболее часто используемых алгоритмов в научных вычислениях.

- Умножение целых чисел. Алгоритм Шёнхаге-Штрассена [6, стр. 304-308]. Сложность алгоритма порядка $O(n \log n \log \log n)$. Отметим, что школьный метод для умножения двух n -разрядных чисел имеет сложность порядка $O(n^2)$.

- Умножение матриц. Алгоритм Штрассена [6, стр. 259-261] имеет сложность порядка $O(n^{\log 7})$, для умножения двух матриц размера $n \times n$. Очевидный алгоритм имеет порядок сложности $O(n^3)$.

Класс E: задачи, экспоненциальные по природе

К экспоненциальным задачам относятся задачи, в которых требуется построить множество всех подмножеств данного множества, все полные подграфы некоторого графа или же все поддеревья некоторого графа.

Задачи не попадающие ни в класс P, ни в класс E

На практике существуют задачи, которые заранее не могут быть отнесены ни к одному из рассмотренных выше классов. Хотя в их условиях не содержатся экспоненциальных вычислений, однако для многих из них до сих пор не разработан эффективный (т.е. полиномиальный) алгоритм.

К этому классу относятся следующие задачи [20, с. 207]:

- задача о выполнимости: существует ли для данной булевой формулы, находящейся в КНФ, такое распределение истинностных значений, что она имеет значение И?

- задача коммивояжера;
- решение систем уравнений с целыми переменными;
- составление расписаний, учитывающих определенные условия;
- размещение обслуживающих центров (телефон, телевидение, срочные службы) для максимального числа клиентов при минимальном числе центров;

- оптимальная загрузка емкости (рюкзак, поезд, корабль, самолёт) при наименьшей стоимости;
- оптимальный раскрой (бумага, картон, стальной прокат, отливки), оптимизация маршрутов в воздушном пространстве, инвестиций, станочного парка;
- задача распознавания простого числа; самый лучший в настоящее время тест на простоту имеет сложность порядка $O(L(n)^{L(L(n))})$, где $L(n)$ - количество цифр в числе n (выражение $L(L(L(n)))$ стремиться к бесконечности очень медленно; первое число, для которого $L(L(L(n))) = 2$, равно $10^{999999999}$) [5, стр. 102].

Недетерминированные алгоритмы

Мы собираемся более подробно классифицировать задачи, не попадающие ни в класс **P**, ни в класс **E**. Для этого вводится понятие недетерминированного алгоритма [26, стр. 443].

Неформально, мы определяем *состояние* алгоритма как комбинацию адреса выполняемой в текущий момент команды и значений всех переменных. Все алгоритмы, рассматривавшиеся до сих пор были *детерминированными*; иначе говоря, во всех них для любого данного состояния существует не больше одного вполне определенного "следующего" состояния. Другими словами, детерминированный алгоритм в каждый момент времени может делать только что-либо одно. В недетерминированном алгоритме для любого данного состояния может быть больше одного допустимого следующего состояния; другими словами, недетерминированный алгоритм в каждый момент времени может делать больше одной вещи. Недетерминированные алгоритмы не являются в каком-то смысле вероятностными или случайными алгоритмами; они являются алгоритмами, которые могут находиться одновременно во многих состояниях.

Недетерминированность лучше всего понять, рассматривая алгоритм, который производит вычисления до тех пор, пока не доходит до места, в котором должен быть сделан выбор из нескольких альтернатив. Детерминированный алгоритм исследовал бы одну альтернативу, а потом бы возвращался бы для исследования другой альтернативы. Недетерминированный алгоритм может исследовать все возможности одновременно, "копируя", в сущности, самого себя для каждой альтернативы. Все копии работают независимо, не сообщаясь друг с другом каким-либо образом. Эти копии, конечно, могут создавать дальнейшие копии и т. д. Если копия обнаруживает, что она сделала неправильный (или безрезультатный) выбор, она прекращает выполняться. Если копия находит решение, она объявляет о своем успехе, и все копии прекращают работать.

Недетерминированный алгоритм можно моделировать с помощью *недетерминированной машины Тьюринга*. Машина Тьюринга, которая бы-

ла введена в §4, является очевидно детерминированной. Обобщим данное там определение, допустив, что каждое значение функции M является множеством троек $\{ \langle \text{записываемый символ} \rangle, \langle \text{переход} \rangle, \langle \text{номер инструкции} \rangle \}$. Теперь для каждого состояния машины может быть несколько следующих состояний, в соответствии с функцией перехода. И в каждом следующем состоянии запускается новая копия данной машины Тьюринга. Формальное определение недетерминированной машины Тьюринга см. в [6].

Очевидно, никакое физическое устройство не способно на неограниченное недетерминированное поведение; недетерминированные алгоритмы - это абстракция, которая позволяет нам игнорировать некоторые проблемы программирования поиска с возвратом.

Определим \mathbf{NP} как класс всех задач, которые можно решить недетерминированными алгоритмами, работающими в течение полиномиального времени, т. е. недетерминированными алгоритмами, в которых всегда есть путь успешного вычисления за время, полиномиальное относительно входа; очевидно, $\mathbf{P} \subseteq \mathbf{NP}$. Поскольку путей вычисления может быть экспоненциально много, вероятно, что алгоритмы, допустимые в этом случае, намного сильнее, чем детерминированные алгоритмы, допустимые для задач из \mathbf{P} .

Укажем причины, по которым задача коммивояжера попадает в класс \mathbf{NP} . С оптимизационными проблемами (такими, например, как задача коммивояжера) связаны соответствующие *проблемы распознавания свойств*. Такие задачи имеют только два возможных решения - "да" или "нет". Выражаясь абстрактно, проблема распознавания T состоит просто из двух множеств: множества D_T всех возможных частных случаев (индивидуальных задач) и множества Y_T ($Y_T \subset D_T$) частных случаев с ответом "да".

Задача распознавания, соответствующая задаче о коммивояжере, может быть сформулирована следующим образом.

Условие. Заданы конечное множество $C = \{c_1, c_2, \dots, c_m\}$ "городов", "расстояние" $d(c_i, c_j)$ между каждой парой городов c_i, c_j из C и граница B - положительное число.

Вопрос. Существует ли "маршрут", проходящий через все города из C , длина которого не превосходит B ? Другими словами, существует ли последовательность $\langle c_{k(1)}, c_{k(2)}, \dots, c_{k(m)} \rangle$ элементов C такая, что

$$\sum_{i=1}^{m-1} d(c_{k(i)}, c_{k(i+1)}) + d(c_{k(m)}, c_{k(1)}) \leq B?$$

Относительно соответствия между задачами распознавания и задачами оптимизации важно отметить, что задача распознавания не может быть сложнее соответствующей задачи оптимизации. Если для задачи о коммивояжере можно за полиномиальное время найти маршрут минимальной длины, то совершенно ясно как за полиномиальное время решить

соответствующую задачу распознавания. Для этого только нужно найти маршрут минимальной длины, вычислить его длину и сравнить с заданной границей B .

Полиномиальный алгоритм задачи коммивояжера неизвестен. Предположим, однако, что имеется некоторый маршрут между городами, претендующий на решение задачи распознавания. Нетрудно проверить, является ли этот маршрут полным обходом всех городов, а если это так, то вычислить его длину, сравнить с границей B и тем самым выяснить является ли этот маршрут положительным решением задачи распознавания. Более того, эту "процедуру проверки" можно представить в виде алгоритма, временная сложность которого ограничена в виде полинома от $|I|$.

Недетерминированный алгоритм, во многих случаях, можно применить для решения задачи распознавания. Такой алгоритм состоит из двух различных стадий - *стадии угадывания* и *стадии проверки*. По заданному частному случаю I проблемы T на первой стадии происходит просто угадывание (генерация) некоторой структуры S . Мы можем считать, что для решения задачи запускается одновременно столько копий алгоритма, сколько существует различных структур S . Затем в каждой копии I и S вместе подаются в качестве входа на стадию проверки, которая выполняется обычным детерминированным образом и либо заканчивается ответом "да", либо заканчиваются ответом "нет", либо продолжается бесконечно без остановки (два последних случая можно не различать). Недетерминированный алгоритм "решает" проблему распознавания T , если для каждого частного случая $I \in D_T$ выполнены следующие два свойства:

1. Если $I \in Y_T$, то существует такая структура S , угадывание которой для входа I приведет к тому, что стадия проверки, начиная работу на входе (I, S) , закончится ответом "да".
2. Если $I \notin Y_T$, то не существует такой структуры S , угадывание которой для входа I обеспечило бы окончание стадии проверки на входе (I, S) ответом "да".

Например, недетерминированный алгоритм решения задачи о коммивояжере можно было бы построить, используя в качестве стадии угадывания просто выбор произвольной последовательности городов, а в качестве стадии проверки упомянутую выше "полиномиальную" процедуру проверки маршрута. Очевидно, что для любого частного случая I найдется такая догадка S , что результатом работы стадии проверки на входе (I, S) будет "да" в том и только том случае, если для частного случая I существует маршрут искомой длины.

NP-трудные и NP-полные задачи

Различные задачи, относящиеся к классу NP являются эквивалентными относительно некоторого отношения, которое мы сейчас определим.

Определение. Задача Q полиномиально сводится к задаче R тогда и только тогда, когда выполнены следующие условия:

- существуют функции $g(x)$ и $f(x)$, вычисляемые за полиномиальное время;
- для любого входа x любого частного случая задачи Q значение $g(x)$ является входом частного случая задачи R ;
- для любого решения (выхода) y задачи R значение $f(y)$ является решением задачи Q .

Таким образом, для решения одной задачи (в данном случае - Q) используется алгоритм другой задачи (R) (рис. 12).

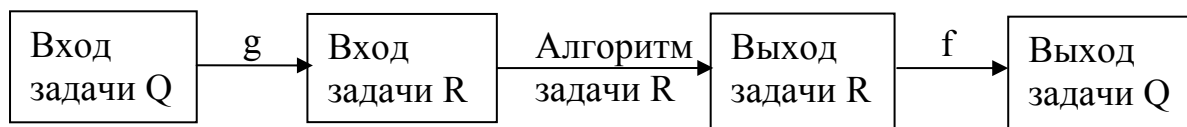


Рис. 12. Полиномиальная сводимость программы Q к R

Определение. Если одновременно задача Q полиномиально сводится к задаче R и R полиномиально сводится к Q , то задачи Q и R полиномиально эквивалентны.

Определение.

Задача является **NP**-трудной (или **NP**-сложной), если каждая задача из **NP** полиномиально сводится к ней.

Задача является **NP**-полной, если она входит в **NP** и является **NP**-трудной. Другими словами, задача T является **NP**-трудной, если она по крайней мере так же сложна, как любая задача в **NP**.

Мы можем отождествить **NP**-полные задачи с "самыми трудными задачами из **NP**". Если хотя бы одна **NP**-полная задача может быть решена за полиномиальное время, то и все **NP**-полные задачи труднорешаемы. Следовательно, любая **NP**-полная задача T обладает свойством: если $P \neq NP$, то $T \in NP \setminus P$. Точнее, $T \in P$ тогда и только тогда, когда $P = NP$.

Первой задачей, для которой было доказано, что она является **NP**-полной, проблема о выполнимости:

Условие. Дана формула исчисления высказываний F , находящаяся в конъюнктивной нормальной форме.

Вопрос. Существует ли такое распределение истинностных значений высказывательных переменных, при которых формула F выполнима?

Теорема (теорема Кука). Задача о выполнимости является **NP**-полной [12, стр. 56-63].

Теперь, для того, чтобы доказать, что Q является **NP**-полной, мы каждый раз должны доказывать, что

- Q попадает в класс **NP**;

- задача о выполнимости полиномиально сводится к Q .

К настоящему времени установлена NP -полнота большого числа задач [12]. Выше мы перечислили некоторые задачи, которые не попадают ни в класс P , ни в класс E . Все они являются NP -полными.

Проблема состоит в следующем: можем ли мы надеяться, что какая-либо из этих задач имеет полиномиальную сложность?

По-видимому, ответ будет неудовлетворительным. Очень важным аргументом для такого вывода служит тот факт, что все эти задачи эквивалентны по сложности - стоит нам найти какой-то полиномиальный алгоритм для одной из этих задач, то все эти задачи становятся полиномиально сложными.

Вероятно, что большинство предполагаемых розенкрейцеров, широко известных в качестве таковых, на самом деле являлись всего лишь розенкрейцерами... и, более того, совершенно очевидно, что они не являлись розенкрейцерами, по простой той причине, что входили в общество розенкрейцеров, что может показаться парадоксальным и на первый взгляд противоречивым, но тем не менее вполне уместным...

Умберто Эко "Маятник Фуко"

Глава 7. ЛОГИЧЕСКИЕ ПАРАДОКСЫ

Приведем несколько широко известных логических парадоксов. Утверждение A называется парадоксом, если из истинности A следует истинность $\neg A$ и из истинности $\neg A$ следует истинность A . Тексты этих парадоксов, там где это специально не оговорено, взяты из книги М. Гарднера [9]. Все эти парадоксы являются подлинными в том смысле, что они не содержат явных логических изъянов. Анализ парадоксов привел к различным планам их устранения. Все эти планы предлагают тем или иным путем ограничить "наивные" понятия ("множество", "характеризовать", "истинный", "прилагательное" и т. п.), участвующих в выводе этих парадоксов. [23, стр. 7-10].

Парадокс лжеца

По преданию, Эпименид утверждал, что все критяне лжецы. Верно ли это утверждение, если учесть, что сам Эпименид родом с острова Крит?

Другая простейшая форма этого парадокса: истинно ли следующее утверждение в рамочке?

Это утверждение ложно

Еще одна форма этого парадокса.

Некто говорит: "Я лгу". Если он при этом лжет, то сказанное им есть ложь, и, следовательно, он не лжет. Если он при этом не лжет, то сказанное им есть истина и, следовательно, он лжет. В любом случае оказывается, что он лжет и не лжет одновременно.

Прямое и противоположное утверждения

Это предложение
содержит шесть слов

Это предложение не
содержит шесть слов

Два утверждения в рамочках являются взаимно исключающими утверждения. Значит одно из них истинно, а другое ложно. Какое именно?

Парадокс Платона и Сократа

Платон: Следующее высказывание Сократа будет ложным.

Сократ: То, что сказал Платон, истинно.

В связи с парадоксом лжеца может возникнуть мысль, что парадокс возникает, когда утверждение ссылается на свою собственную истинность. Но парадокс Платона и Сократа показывает, что причина лежит глубже.

Парадокс Ришара (1905)[23, стр. 9]

С помощью некоторых фраз русского языка могут быть охарактеризованы те или иные вещественные числа. Например, фраза "отношение длины окружности к длине диаметра в круге" характеризует число π . Все фразы русского языка могут быть перенумерованы некоторым стандартным способом, а именно: упорядочим сперва лексикографически (т. е. как в словаре) все фразы, содержащие в точности k букв, а затем поместим все фразы из k букв впереди всех фраз с большим числом букв. Теперь можно перенумеровать все фразы русского языка, которые характеризуют то или иное вещественное число. Для этого достаточно в стандартной нумерации всех фраз опустить все остальные фразы. Число, получающее при такой нумерации номер n , назовем n -ым числом Ришара. Рассмотрим такую фразу: "вещественное число, у которого n -ый десятичный знак равен 1, если у n -го числа Ришара n -ый десятичный знак не равен 1, и n -ый десятичный знак равен 2, если у n -го числа Ришара n -ый десятичный знак равен 1". Эта фраза определяет некоторое число Ришара, допустим k -ое; однако, согласно определению, оно отличается от k -го числа Ришара в k -ом десятичном знаке.

Варианты парадокса Б. Рассела

Курт Греллинг (немецкий математик, 1908):

Разделим все прилагательные на два множества: *самодескриптивные*, обладающие тем свойством, которые они выражают, и *несамодескриптивные*. Такие прилагательные, как "многосложное", "русское" и "видимое", принадлежат к числу самодескриптивных, а такие прилагательные, как "односложное", "немецкое" и "невидимое", - к числу несамодескриптивных. К какому множеству принадлежит прилагательное "несамодескриптивное"?

Дж. Дж. Берри (библиотекарь оксфордского университета, 1906):

В парадоксе речь идет о "наименьшем целом числе, которое не может быть задано менее чем тринадцатью словами". Выражение, взятое в кавычки, содержит 12 слов. Какому множеству принадлежит определяемое число: множеству целых чисел, которые на русском языке задаются менее чем тринадцатью словами, или множеству целых чисел, задаваемых на русском языке 13 и более словами?

Любой из двух ответов приводит к противоречию.

Макс Блэк (философ):

В книге упоминаются различные целые числа. Сосредоточим наше внимание на наименьшем целом числе, которое ни прямо, ни косвенно не упоминается в этой книге. Существует ли такое число?

"Казнь врасплох" [10, с. 96-97]

"Преступника приговорили к смертной казни через повешение и поместили его в тюрьму в субботу.

– Тебя повесят в полдень, - сказал ему судья, - в один из семи дней на следующей неделе. Но в какой именно день это должно произойти, ты узнаешь лишь утром в день казни.

Судья славился тем, что всегда держал свое слово. Осужденный вернулся в камеру в сопровождении адвоката. Как только их оставили вдвоем, защитник удовлетворенно ухмыльнулся.

– Неужели не понятно? - воскликнул он. - Ведь приговор судьи нельзя привести в исполнение!

– Как? Ничего не понимаю, - пробормотал узник.

– Сейчас объясню. Очевидно, что в следующую субботу тебя не могут повесить: суббота - последний день недели, и в пятницу днем ты бы уже знал наверняка, что тебя повесят в субботу. Таким образом, о дне казни тебе бы стало известно до официального уведомления в субботу утром, следовательно, приказ судьи был бы нарушен.

– Верно, - согласился заключенный.

– Итак, суббота, безусловно, отпадает, - продолжал адвокат, поэтому пятница становится последним днем, когда тебя смогут повесить. Однако и в пятницу повесить тебя нельзя, ибо после четверга осталось бы всего два дня - пятница и суббота. Поскольку суббота не может быть днем казни, повесить тебя должны лишь в пятницу. Но раз тебе об этом станет известно еще в четверг, то приказ судьи опять будет нарушен. Следовательно, пятница тоже отпадает. Итак, последний день, когда тебя еще могли казнить, это четверг. Однако четверг тоже не годится, потому что, оставшись в среду живым, ты сразу поймешь, что казнь должна состояться в четверг.

– Все понятно! - воскликнул заключенный, воспрянув духом. - Точно так же я могу исключить среду, вторник и понедельник. Остается только завтрашний день. Но завтра меня наверняка не повесят, потому что я знаю об этом уже сегодня.

Итак, безупречными логическими рассуждениями приговоренный убедился в том, что, не нарушив приговора, казнь совершить невозможно. И вдруг, к немалому удивлению осужденного, в четверг утром в камеру является палач. Осужденный, конечно, этого не ждал, но самое удивитель-

ное, что приговор оказался совершенно точным - его можно привести в исполнение в полном соответствии с формулировкой".

Парадокс о вычислимых функциях [14, с. 184]

Легко доказать, что множество всюду определенных вычислимых функций $f: \subseteq \rightarrow \subseteq$ является перечислимым, т. е. их можно перенумеровать в виде последовательности f_1, f_2, f_3, \dots .

Определим теперь новую функцию g формулой

$$g(n) = f_n(n) + 1.$$

Она не входит в нашу последовательность, поскольку при $n=1$ она отличается от f_1 , при $n=2$ - от f_2 и т. д. Следовательно, она не вычислима.

С другой стороны, ясно, что она вычислима, так как $f_n(n)$ вычислима, а прибавив 1 к $f_n(n)$, мы получим $g(n)$.

Этот парадокс можно объяснить. На самом деле мы здесь используем две формальные системы. В рамках одной системы (скажем, арифметики), мы описываем вычислимые функции f , а то, что g является вычислимой функцией мы получаем в рамках другой формальной системы, в которой уже используется возможность упорядочить f . Вторая формальная система является надсистемой или метасистемой относительно первой.

Глава 8. МНОГОЗНАЧНЫЕ ЛОГИКИ

Двузначная логика предполагает истинность и ложность высказываний («0» или «1»). В многозначных логиках число значений истинности аргументов и функций может быть даже (в общем случае) бесконечным. Обозначим через Nx или $\neg x$ — отрицание, Sxy или $x \supset y$ — импликацию, Kxy или $x \wedge y$ — конъюнкцию, Axy или $x \vee y$ — дизъюнкцию. Значение функции от аргумента a обозначим как $[a]$.

Напомним, что тавтологией называется формула, принимающая значение «истина» (или 1) при любых комбинациях значений входящих в нее переменных. Развитие многозначных логик приводит к тому, что ряд утверждений, являющихся тождественно-истинными в одной логической системе, становятся нетождественно-истинными в другой системе.

Рассмотрим несколько примеров многозначных логик, которые подтверждают это утверждение.

§1. Трехзначная система Я. Лукасевича

Эта пропозиционная логика была построена Я. Лукасевичем в 1920 году [34]. Лукасевич обозначил «истину» за «1», «ложь» за «0» и ввел третье значение — «нейтрально» — $\frac{1}{2}$. Основными функциями им были взяты отрицание и импликация, а производными — конъюнкция и дизъюнкция. Тавтология в логике Я. Лукасевича принимает значение «1». Отрицание и импликация определяются таблицами 11, 12 и равенствами:

Таблица 11

X	Nx
1	0
$\frac{1}{2}$	$\frac{1}{2}$
0	1

Таблица 12

$x \supset y$	1	$\frac{1}{2}$	0
1	1	$\frac{1}{2}$	0
$\frac{1}{2}$	1	1	$\frac{1}{2}$
0	1	1	1

$$[Nx] = 1 - [x];$$

$$[Sxy] = 1, \text{ если } [x] \leq [y];$$

$$[Sxy] = 1 - [x] + [y], \text{ если } [x] > [y] \text{ или в общем виде;}$$

$$[Sxy] = \min(1, 1 - [x] + [y]).$$

Конъюнкция в системе Лукасевича определяется как минимум значений аргументов: $[Kxy] = \min([x], [y])$, а дизъюнкция – как максимум значений аргументов x и y : $[Axy] = \max([x], [y])$.

Очевидно, что в этих определениях не являются законами логики (тавтологиями) законы двузначной логики: исключенного третьего, непротиворечия, отрицания законов непротиворечия и исключенного третьего. Поэтому система Лукасевича не является отрицанием двузначной логики. В его логике правило снятия двойного отрицания, четыре правила де Моргана и правило контрапозиции: $\neg a \supset b \sim b \supset a$ являются тавтологиями. Не являются тавтологиями правила приведения к абсурду двузначной логики: $(\neg x \supset \neg x) \supset x$ и $(x \supset (y \wedge \neg y)) \supset \neg x$ (т.е. если из x вытекает противоречие, то из этого следует отрицание x). Доказывается это, если задать $[x] = \frac{1}{2}$ и $[y] = \frac{1}{2}$.

В данной логике не являются тавтологиями и ряд формул, выражающие правильные дедуктивные умозаключения традиционной логики, формализованные средствами алгебры логики.

Очевидно, что все тавтологии логики Лукасевича являются тавтологиями в двузначной логике. Но т.к. у Лукасевича имеется еще одно значение истинности - $\frac{1}{2}$, то обратное утверждение неверно.

§2. Логика Гейтинга

Из закона исключенного третьего в двузначной логике выводятся:

$$1. \neg\neg x \supset x \qquad 2. x \supset \neg\neg x$$

Гейтинг создал трехзначную пропозициональную логику, основываясь на утверждении, что истинным является лишь $x \supset \neg\neg x$. Импликация и отрицание (таблицы 13, 14 отличаются от определений этих операций. В предыдущей логике лишь в одном случае “истина” обозначена Гейтингом за “1”, “ложь” – “0” и введено понятие “неопределенность” - $\frac{1}{2}$. Тавтология принимает значение 1.

Таблица 13

X	Nx
1	0
$\frac{1}{2}$	0
0	1

Таблица 14

x \ y	1	$\frac{1}{2}$	0
1	1	$\frac{1}{2}$	0
$\frac{1}{2}$	1	1	0
0	1	1	1

$[Cxy] = 1$, если $[x] \leq [y]$;

$[Cxy] = [y]$, если $[x] > [y]$

Kxy и Axy определены как минимум и максимум значений аргумента.

Очевидно, что учет лишь значений функций 1 и 0 приводит к вычислению матрицы двузначной логики. В логике Гейтинга законы непротиворечия, формула $(x \supset y) \supset (y \supset x)$, де Моргана и исключенного четвертого: $(\neg x \vee x \vee \neg \neg x)$ являются тавтологиями. Но ни закон исключенного третьего, ни его отрицание не являются тавтологиями.

Казалось бы незначительные изменения системы Лукасевича (матрицы отрицания и импликации), сделанные Гейтингом, не должны значительно изменить полученные результаты. Однако это не так, поскольку в логике Гейтинга многие формулы двузначного исчисления высказываний являются тавтологиями [35].

§3. Трехзначная система Бочвара Д.А.

Система создавалась Бочваром Д.А. [36] для разрешения парадоксов классической математической логики методом формального доказательства бессмысленности определенных высказываний. Например, ему удалось разрешить парадокс Рассела о множестве всех нормальных множеств и доказать несуществование такого предмета, как множество всех нормальных множеств. Это означает, что множество всех нормальных множеств нельзя рассматривать как фиксированный объект, не изменяющийся от времени.

Создавая свою систему Д.А. Бочвар, разделил высказывания на имеющие смысл («истина» или «ложь») и бессмысленными. Он выделил внешние и внутренние формы (функции).

Внутренние функции называются классическими содержательными функциями переменных высказываний, а внешние – неклассическими. Обозначив «истина» за R или 1, «ложь» – F или 3, «бессмысленность» – S или 2, автор ввел отрицание внутреннее – « $\sim a$ », внешнее отрицание – « $\neg a$ », « \bar{a} » – внутреннее отрицание внешнего утверждения, « \equiv » – внешняя равнозначность, « \leftrightarrow » – внешняя равносильность. В логике Бочвара Д.А. законы тождества, отрицания двузначной логики не являются тавтологиями.

Отрицание закона тождества как раз и позволило разрешить парадокс Рассела.

В логике Бочвара Д.А. формулы, приведенные ниже являются противоречиями:

$a \wedge \neg a$;

$a \leftrightarrow \bar{a}$;

$a \equiv \neg a$.

§4. K - значная логика Поста Е.Л.

Логика Поста [37] является обобщением частного случая – двузначной логики, когда $K=2$. Действительно, по Посту значения истинности принимают значения $1, 2, \dots, K$ (при $K \geq 2$ и K – конечно). В этих терминах формула является тавтологией, когда принимает такое значение i , что $1 \leq i \leq S$, где $1 \leq S \leq K-1$. Значения $1, \dots, S$ называются выделенными (отмеченными). При этом S может быть и больше 2. Пост ввел N^1_x – циклическое отрицание, N^2_x – симметричное отрицание. Они определяются таблицей 15 и равенствами.

Таблица 15

X	N^1_x	N^2_x
1	2	K
2	3	$K-1$
3	4	$K-2$
⋮	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮
$K-1$	K	2
K	1	1

Циклическое отрицание определяется равенствами:

$$[N^1_x] = [x] + 1 \text{ при } [x] \leq K-1$$

$$[N^1_x] = 1.$$

Симметричное отрицание по Посту определяется:

$$[N^2_x] = K - [x] + 1$$

Очевидно, что при $K=2$ циклическое и симметричное отрицания совпадают с отрицанием двузначной логики и между собой.

Операции конъюнкции и дизъюнкции определяются как минимум и максимум значений аргументов.

Читайте книги - некоторые из них
специально для этого написаны.

Михаил Генин

Автор пишет только половину книги:
другую половину пишет читатель.

Джозеф Конрад

ЛИТЕРАТУРА

1. Carry H. B., Feys R. Combinatory Logic, vol. I, Amsterdam: North-Holland Co., 1958.
2. Church A. The Calculi of Lambda Conversion. Princeton University Press, Princeton, 1941.
3. Schönfinkel M. Über die Bausteine der mathematischen Logik. Math. Annalen, 92, 1924, s. 305-316.
4. Turing A. M. On computable numbers with an application to the Entscheidungs-problem. - Proc. London Math. Soc., Ser. 2, 1936, 42, p. 230-265.
5. Акритас А. Основы компьютерной алгебры с приложениями: пер. с англ. - М.: Мир, 1994. - 544 с.
6. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. М.: Мир, 1979. - 536 с.
7. Барендрегт Х. Ламбда-исчисление. Его синтаксис и семантика. - М.: Мир, 1985.-606с.
8. Де Боно Э. Латеральное мышление - СПб.: Питер Пвблишинг, 1997. - 320с.
9. Гарднер М. А ну-ка, догадайся! М.: Мир, 1984.- 213 с.
10. Гарднер М. Математические досуги: Пер. с англ. - М.: Оникс, 1995.- 496с.
11. Грэй П. Логика, алгебра и базы данных: Пер. с англ.- М.: Машиностроение, 1989. - 359 с.
12. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. - М.: Мир, 1982.
13. Катленд Н. Вычислимость. Введение в теорию рекурсивных функций: Пер. с англ.- М.: Мир, 1983.-256 с.
14. Кац М., Улам С. Математика и логика. Ретроспектива и перспективы: Пер. с англ. - М.: Мир, 1971. - 254 с.
15. Кнут Д. Искусство программирования для ЭВМ.
16. Том 1. Основные алгоритмы. М.: Мир, 1976. - 736 с.
17. Том 2. Получисленные алгоритмы. М.: Мир, 1977. - 724 с.
18. Том 3. Сортировка и поиск. М.: Мир, 1978. - 844 с.
19. Литлвуд Дж. Математическая смесь: Пер. с англ.- М.: Наука, 1990. - 140 с.

20. Лорьер Ж.-Л. Системы искусственного интеллекта - М.: Мир, 1991. - 568с.
21. Манин Ю. И. Вычислимое и невычислимое. - М.: Сов. радио. 1980. - 128 с.
22. Манин Ю. И. Доказуемое и недоказуемое. М.: "Советское радио", 1979.
23. Мендельсон Э. Введение в математическую логику - М.: Наука, 1976.- 320с.
24. Нефедов В. Н., Осипова В. А. Курс дискретной математики. М., 1992.
25. Никифоров А. Книга о логике. М.: "Гнозис", 1996
26. Рейнгольд Э., Нивергельт Ю., Део Н. Комбинаторные алгоритмы. Теория и практика. Пер. с англ. - М.: Мир, 1980. - 478 с.
27. Смаллиан Р. Как же называется эта книга? М.: Мир, 1981.
28. Смаллиан Р. Принцесса или тигр? М.: Мир, 1985. - 221 с.
29. Справочная книга по математической логике: В 4-х частях/Под ред. Дж. Барвайса. - Ч. III. Теория рекурсии: Пер. с англ.-М.: Наука, 1982.- 360 с.
30. Таранов П. С. Секреты поведения людей: Опыт всемирной энциклопедии жизни людей в законах и примерах. - Симферополь: Таврия, 1997. - 544 с.
31. Успенский В. А. Теорема Гёделя о неполноте - М.: Наука, 1982. - 112 с.
32. Филд А., Харрисон П. Функциональное программирование - М.: Мир, 1993.-637с.
33. Штейнгауз Г. Математический калейдоскоп: Пер. с польского. - М.: Наука, 1981. - 160 с.
34. Lukasiewicz. Opojeciu mozliewosci.-Ruch Filozoficzny. Lwow. 1920. R.5 №9.
35. Гетманова А.Д. Учебник по логике. 2-е изд. - М.: Владос, 1995.-303с.
36. Бочвар Д.А. Об одном трехзначном исчислении и его применении к анализу парадоксов классического расширенного функционального исчисления. //Математический сборник.-1938. Т. 4(46). № 2.
37. Post E.L. Introduction to a General Theory of Elementary Propositions //American Journal of Mathematics. 1921. Vol. 43. № 3.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО КУРСУ «МАТЕМАТИЧЕСКАЯ ЛОГИКА И ТЕОРИЯ АЛГОРИТМОВ»

Цели и задачи дисциплины

Цели преподавания дисциплины является ознакомление студентов с основами математической логики, теории алгоритмов с методами оценки сложности алгоритмов и построения эффективных алгоритмов.

В результате изучения дисциплины обучаемый должен:

уметь находить представление и исследовать свойства булевых и многозначных функций формулами в различных базисах, применять методы математической логики и теории алгоритмов к решению задач математической кибернетики, оценивать сложность алгоритмов и вычислений при решении практических задач;

иметь навыки использования современной символической логики, построения алгоритмов и анализа их сложности. Построения функциональных схем реализации булевых функций, поиска научной информации и работы с реферативной, справочной, периодической и монографической литературой по математической логике;

знать язык и средства современной математической логики, основные свойства булевых и многозначных функций, различные подходы к определению алгоритма и доказательства алгоритмической неразрешимости отдельных массовых задач, методы доказательства оптимальности алгоритма, возможности применения общих логических принципов в математике и специальных науках, историю развития математической логики и вклад отечественных ученых в решении проблем математической логики, теории алгоритмов и их развитие в связи с приложениями к кибернетике и специальной науке.

Дисциплина является основной в системе курсов дискретной математики. Изучаемый материал используется в курсах компьютерного и криптографического циклов.

Наименование тем

Введение

История развития математической логики и теории алгоритмов. Математическая логика и основания математики. Теория алгоритмов и принципиальные возможности вычислительных машин. Сложность алгоритмов и ее значение для практики. (2 часа)

Алгебра высказываний и алгебра предикатов

Основные логические операции и их свойства. Понятие булевой алгебры. Алгебра высказываний и алгебра подмножеств, множества как примеры булевых алгебр. Предикаты на множестве и их связь с отношениями. Логические операции над предикатами. Определение формулы алгебры предикатов. Выполнимые. Тождественно истинные и тождественно ложные формулы. Равносильность формул, основные формулы равносильности и их использование для упрощения формул. Существование для каждой формулы алгебры высказываний приведенной формы, дизъюнктивной и конъюнктивной нормальных форм. (4 часа)

Булевы функции и их обобщения

Понятие булевой функции и функций многозначной логики. Их представление формулами над заданной системой функций. Представление булевых функций формулами алгебры высказываний и многочленами Жегалкина. Критерии полноты для булевых функций и функций многозначной логики. (2 часа)

Исчисление высказываний

Общее понятие о логическом исчислении. Выводимость и доказуемость формул в исчислении высказываний. Непротиворечивость и полнота исчисления высказываний. (2 часа)

Исчисление предикатов

Язык, аксиомы и правила вывода исчисления предикатов. Вспомогательные правила вывода: правило силлогизма, правила разделения и умножения формул, правила умножения и разделения посылок, правило перестановки посылок, правило умножения заключений, правило контрапозиции, правила де Моргана, правила противоречия, закон исключенного третьего. Эквивалентность формул. Приведение формул к нормальным формам. Теорема Гёделя о полноте исчисления предикатов. Применение исчисления предикатов для записи математических утверждений и автоматического доказательства теорем. (2 часа)

Метод резолюций

Применение исчисления предикатов для доказательства теорем. Метод резолюций для логики предикатов. Теорема о полноте метода резолюций для логики предикатов. (2 часа)

Элементы теории алгоритмов

Интуитивное понятие алгоритма и его характерные черты. Необходимость уточнения понятия алгоритма. Определение нормального алгоритма. Примеры. Принцип Маркова. Композиция нормальных алгоритмов. Определение машины Тьюринга - Поста. Нумерация слов в счетном алфавите и арифметизация алгоритмов. Определение рекурсивных и частично рекурсивных функций. Примеры алгоритмически неразрешимых массовых задач. Неразрешимость проблем распознавания самоприменимости нормальных алгоритмов и самоприменимости машин Тьюринга. Теорема Чёрча о неразрешимости исчислений предикатов. Рекурсивные и рекурсивно перечислимые множества. Примеры. Теорема Клини о неподвижной точке. (6 часов)

Сложность алгоритмов и вычислений

Подходы к оценкам сложности алгоритмов и вычислений. Модели вычислений. Сложность вычисления на машине Тьюринга. Меры сложности. Методы построения эффективных алгоритмов. Метод разбиения и рекурсии. Сложность рекурсивных алгоритмов. Умножение чисел и матриц. Быстрое преобразование Фурье. (4 часа)

Теория алгоритмов и задачи использования ЭВМ

Вычислительные возможности современных ЭВМ. Модель ЭВМ - машина произвольного доступа. МПД - вычислимые функции и их связь с частично рекурсивными функциями. (2 часа)

Темы практических занятий

Таблицы истинности формул алгебры высказываний. (2 часа)

Нормальные формы формул. (4 часа)

Булевы функции. (4 часа)

Правила вывода в исчислении высказываний.	(2 часа)
Правила вывода в исчислении предикатов.	(2 часа)
Построение машин Тьюринга.	(2 часа)
Алгоритмы Маркова.	(2 часа)
Частично рекурсивные функции.	(2 часа)
Функции сложности.	(2 часа)
Построение машин произвольного доступа.	(2 часа)
Контрольная работа.	(2 часа)

Самостоятельная работа

№	Наименование работы	Количество час.	Форма контроля
1.	Проработка лекционного материала.	18	Зачет.
2.	Подготовка к практическим занятиям. Выполнение домашних заданий.	8	Опрос на прак. занятиях.
3.	Подготовка к контрольным работам.	12	Проверка контрольной работы
4.	Изучение тем для самостоятельной проработки	7	Зачет. Конспекты.

Всего часов самостоятельной работы - 45 часов.

КОНТРОЛЬНЫЕ ЗАДАНИЯ ПО КУРСУ "МАТЕМАТИЧЕСКАЯ ЛОГИКА И ТЕОРИЯ АЛГОРИТМОВ"

(Упражнение, друзья, дает больше,
чем хорошее (природное) дарование.

Эпихарм

КОНТРОЛЬНАЯ № 1

Вариант 1

1. Найдите множество X , удовлетворяющее следующему условию:
 $A \setminus (A \setminus X) = \emptyset$.
2. Равны ли два множества:
 $\{\{1,2\}, \{2,3\}\}$ и $\{1,2,3\}$?
3. Докажите следующее утверждение: $A \subset B$ и $B \subseteq C \Rightarrow A \subset C$.
4. Пусть $A = \{0, 1\}$. Перечислите элементы множеств A^2, A^3 .
5. Пусть ρ и φ - бинарные отношения, определенные на некотором множестве. Тогда $(\varphi \setminus \rho)^{-1} = \varphi^{-1} \setminus \rho^{-1}$.
6. Укажите все сюръективные отображения множества $A = \{1,2,3\}$ на множество $B = \{a, b\}$.
7. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{A \cap B}(x) = \chi_A(x) + \chi_B(x) - \chi_A(x) \chi_B(x);$$

8. Пусть отношение ρ определено на множестве N^2 (N - множество натуральных чисел $\{1,2,3,\dots\}$): $\langle x,y \rangle \rho \langle u,v \rangle \Leftrightarrow x+v = y+u$. Доказать, что ρ - отношение эквивалентности.
9. Построить линейный порядок: а) на множестве N^2 ; б) на множестве $N \cap N^2 \cap \dots \cap N^n \cap \dots = \{\text{все конечные последовательности из натуральных чисел}\}$.
10. Если ρ и φ - отношения эквивалентности на X , то $\rho \cup \varphi$ также отношение эквивалентности на X .

Вариант 2

1. Докажите следующее утверждение: $A \subset B$ и $B \subseteq C \Rightarrow A \subset C$.
2. Равны ли два множества:
 $\{2,3\}, \{3,4\}$ и $\{2,3,4\}$?
3. Найдите $A \cap B, A \cup B, A \setminus B, B \setminus A, \overline{A}, \overline{B}$ для

$$A=\{1,2,3\}, B=\{2,3,4,5\}, U = \{0,1,\dots,9\};$$

4. Определим упорядоченную пару $\langle a,b \rangle$ как множество $\{\{a\}, \{a,b\}\}$. Убедимся, что такое формальное теоретико-множественное определение вполне соответствует нашему неформальному определению упорядоченной пары. Для этого достаточно доказать, что для любых элементов $\langle a,b \rangle = \langle c,d \rangle \Leftrightarrow a=c, b=d$.
5. Пусть $x \rho y \Leftrightarrow x^2 = y^2$. Определите ρ^{-1} , $\rho \perp \rho$, $\rho^{-1} \perp \rho^{-1}$.
6. Найдите все отображения множества $A=\{1,2\}$ в себя, укажите, какие из них инъективные, сюръективные.
7. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{\bar{A}}(x) = 1 - \chi_A(x);$$

8. Если ρ и φ - отношения эквивалентности на X , то $\rho \cup \varphi$ также отношение эквивалентности на X .
9. Докажите, что отношение делимости на множестве натуральных чисел N является отношением частичного порядка. Является ли это отношение линейным порядком? Является ли отношением частичного порядка отношение делимости на множестве целых чисел Z ?
10. Если ρ и φ - отношения эквивалентности на X , то $\rho \cap \varphi$ - отношение эквивалентности на $X \Leftrightarrow \rho \cap \varphi = \rho \perp \varphi$.

Вариант 3

1. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание:
 $\{1,2\} ? \{1,2, \{1\}, \{2\}\};$
2. Равны ли два множества:
 $\{4,5\}, \{5,6\}$ и $\{4,5,6\}$?
3. Найдите $A \cap B, A \cup B, A \setminus B, B \setminus A, \bar{A}, \bar{B}$ для
 $A=\{x \mid x \text{ делится на } 2\}, B = \{x \mid x \text{ делится на } 3\}, U = N$ - множество натуральных чисел.
4. Пусть $A \subseteq C, B \subseteq C$. Докажите, что $A \times B = (A \times C) \cap (C \times B)$.
5. Пусть ρ - бинарное отношение на R и $e_R = \{\langle x,x \rangle \mid x \in R\}$. Доказать, что $\rho = e_R \Leftrightarrow \rho \perp \varphi = \varphi \perp \rho = \varphi$ для любого отношения φ на R .
6. Пусть X - конечное множество и отображение $f: X \rightarrow X$ инъективно. Доказать, что тогда f биективно.
7. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать: $\chi_{A \setminus B}(x) = \chi_A(x) - \chi_A(x) \chi_B(x)$.

8. Если ρ и φ - отношения эквивалентности на X , то $\rho \cap \varphi$ - отношение эквивалентности на $X \Leftrightarrow \rho \cap \varphi = \rho \perp \varphi$.
9. Докажите, что $M = \{\{1\}, \{2,5\}, \{3\}, \{4,6,7\}\}$ - разбиение множества $A = \{1,2,3,4,5,6,7\}$. Перечислите все элементы отношения эквивалентности ρ , соответствующего разбиению M .
10. Если ρ - частичный порядок на X , то ρ^{-1} также частичный порядок на X .

Вариант 4

1. Докажите следующее утверждение: $A \subset B$ и $B \subseteq C \Rightarrow A \subset C$.
2. Доказать, что если конечное множество A содержит n элементов, то множество-степень $P(A)$ содержит 2^n элементов.
3. Докажите, что $\overline{A \setminus B} = \overline{A} \cap B$.
4. Докажите, что подмножество C множества $A \times B$ является прямым произведением некоторого подмножества A_1 множества A и подмножества B_1 множества B тогда и только тогда, когда для любых $\langle a, b \rangle \in C$, $\langle c, d \rangle \in C$ следует, что $\langle a, d \rangle, \langle c, b \rangle \in C$.
5. Пусть $A = \{0, 1\}$. Перечислите элементы множеств A^2, A^3 .
6. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{A \cup B}(x) = \chi_A(x) \vee \chi_B(x);$$

7. Пусть $f: X \rightarrow Y$ и $A \subseteq X$. Образом множества A при отображении f называется множество $f(A) = \{y \mid y = f(x), x \in A\}$. Пусть $B \subseteq Y$. Прообразом множества B при отображении f называется множество $f^{-1}(B) = \{x \in X \mid f(x) \in B\}$. Доказать, что $f^{-1}(A \cap B) = f^{-1}(A) \cap f^{-1}(B)$.
8. Если ρ - частичный порядок на X , то ρ^{-1} также частичный порядок на X .
9. Пусть $A = \{1, 2, 3\}$. На множестве $P(A)$ задано бинарное отношение $X \rho Y \Leftrightarrow$ "множества X и Y имеют равное количество элементов". Доказать, что это отношение эквивалентности и найдите классы эквивалентности.
10. Докажите, что $M = \{\{1\}, \{2,5\}, \{3\}, \{4,6,7\}\}$ - разбиение множества $A = \{1,2,3,4,5,6,7\}$. Перечислите все элементы отношения эквивалентности ρ , соответствующего разбиению M .

Вариант 5

1. Перечислить все элементы каждого из следующих множеств:

$$\{x \mid x \subseteq \{1\}\};$$

2. Доказать, что для любых множеств A и B имеем $A \cap (A \cup B) = A$.

3. Найдите множество X , удовлетворяющее следующему условию:

$$A \setminus (A \setminus X) = \emptyset.$$

4. Пусть A, B, C, D - непустые множества. Докажите, что

$$A \subseteq B \text{ и } C \subseteq D \Leftrightarrow A \times C \subseteq B \times D,$$

5. Определим упорядоченную пару $\langle a, b \rangle$ как множество $\{\{a\}, \{a, b\}\}$. Убедимся, что такое формальное теоретико-множественное определение вполне соответствует нашему неформальному определению упорядоченной пары. Для этого достаточно доказать, что для любых элементов $\langle a, b \rangle = \langle c, d \rangle \Leftrightarrow a=c, b=d$.

6. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{A \cap B}(x) = \chi_A(x) + \chi_B(x) - \chi_A(x) \chi_B(x);$$

7. Пусть $A = \{a_1, a_2, \dots, a_n\}$ - конечное множество. Определим отображение $f: P(A) \rightarrow \{0, 1\}^n$ следующим образом

$$f(B) = \langle \alpha_1, \alpha_2, \dots, \alpha_n \rangle, \text{ где } \alpha_i = 0, \text{ если } a_i \notin B, \text{ и } \alpha_i = 1, \text{ если } a_i \in B.$$

Докажите, что f - биекция.

8. Пусть $A = \{1, 2, 3\}$. На множестве $P(A)$ задано бинарное отношение $X \rho Y \Leftrightarrow$ "множества X и Y имеют равное количество элементов". Доказать, что это отношение эквивалентности и найдите классы эквивалентности.

9. Если ρ - частичный порядок на X , то ρ^{-1} также частичный порядок на X .

10. Докажите, что отношение делимости на множестве натуральных чисел \mathbb{N} является отношением частичного порядка. Является ли это отношение линейным порядком? Является ли отношением частичного порядка отношение делимости на множестве целых чисел \mathbb{Z} ?

Вариант 6

1. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\emptyset ? \{\emptyset\}$.

2. Перечислить все элементы каждого из следующих множеств:

$$\{x \mid x \subseteq \{1\}\};$$

3. Найдите соответствующую формулу $P(x)$ для каждого множества:

$$\{2, 3, 5, 7, 11, 13, 17, 19\};$$

4. Пусть A, B, C, D - непустые множества. Докажите, что
 $A \times C = B \times D \Leftrightarrow A=B$ и $C=D$.
5. Пусть $A \subseteq C, B \subseteq C$. Докажите, что $A \times B = (A \times C) \cap (C \times B)$.
6. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{\bar{A}}(x) = 1 - \chi_A(x);$$

7. Найдите прообраз множества $\{0\}$ при следующих отображениях $\mathbf{R} \rightarrow \mathbf{R}$:
- $x \rightarrow \sin(x)$;
 - $x \rightarrow \lg(x^2+1)$;
 - $x \rightarrow x^2+2x+1$.
8. Докажите, что $M = \{\{1\}, \{2,5\}, \{3\}, \{4,6,7\}\}$ - разбиение множества $A = \{1,2,3,4,5,6,7\}$. Перечислите все элементы отношения эквивалентности ρ , соответствующего разбиению M .
9. Если ρ и φ - отношения эквивалентности на X , то $\rho \cap \varphi$ - отношение эквивалентности на $X \Leftrightarrow \rho \cap \varphi = \rho \cup \varphi$.
10. Построить линейный порядок: а) на множестве \mathbf{N}^2 ; б) на множестве $\mathbf{N} \cap \mathbf{N}^2 \cap \dots \cap \mathbf{N}^n \cap \dots = \{\text{все конечные последовательности из натуральных чисел}\}$.

Вариант 7

1. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\emptyset ? \{\{\emptyset\}\}$;
2. Перечислите все элементы каждого из следующих множеств:
 $\{x \mid x \subseteq \{1,2\}\}$;
3. Найдите соответствующую формулу $P(x)$ для каждого множества:
 $\{м, о, н, е, ж, т, с, в\}$;
4. Докажите тождество $(A \setminus B) \times C = (A \times C) \setminus (B \times C)$.
5. Докажите, что подмножество C множества $A \times B$ является прямым произведением некоторого подмножества A_1 множества A и подмножества B_1 множества B тогда и только тогда, когда для любых $\langle a,b \rangle \in C, \langle c,d \rangle \in C$ следует, что $\langle a,d \rangle, \langle c,b \rangle \in C$.
6. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{A \setminus B}(x) = \chi_A(x) - \chi_A(x) \chi_B(x).$$

7. Какие отображения инъективны, сюръективны?
 $f: \mathbb{R} \rightarrow \mathbb{R}, x \mapsto x^2 + 3x + 5$;
8. Докажите, что отношение делимости на множестве натуральных чисел \mathbb{N} является отношением частичного порядка. Является ли это отношение линейным порядком? Является ли отношением частичного порядка отношение делимости на множестве целых чисел \mathbb{Z} ?
9. Если ρ и φ - отношения эквивалентности на X , то $\rho \cup \varphi$ также отношение эквивалентности на X .
10. Докажите, что $M = \{\{1\}, \{2,5\}, \{3\}, \{4,6,7\}\}$ - разбиение множества $A = \{1,2,3,4,5,6,7\}$. Перечислите все элементы отношения эквивалентности ρ , соответствующего разбиению M .

Вариант 8

1. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\emptyset ? \{1,2,\{1\},\{\emptyset\}\}$;
2. Перечислите все элементы каждого из следующих множеств:
 $\{x \mid x \subseteq \{1,2,3\}\}$;
3. Найдите соответствующую формулу $P(x)$ для каждого множества:
 $[-2,3]$.
4. Докажите, что $(A \times B) \cap (C \times D) \subseteq (A \cap C) \times (B \cap D)$.
5. Пусть A, B, C, D - непустые множества. Докажите, что
 $A \subseteq B$ и $C \subseteq D \Leftrightarrow A \times C \subseteq B \times D$,
6. Пусть $f: X \rightarrow Y$ и $A \subseteq X$. Образом множества A при отображении f называется множество $f(A) = \{y \mid y = f(x), x \in A\}$.
 Пусть $B \subseteq Y$. Прообразом множества B при отображении f называется множество $f^{-1}(B) = \{x \in X \mid f(x) \in B\}$.
 Доказать, что $f^{-1}(A \cap B) = f^{-1}(A) \cap f^{-1}(B)$.
7. Какие отображения инъективны, сюръективны?
 $f: \mathbb{R} \rightarrow \mathbb{R}, x \mapsto x^{15}(x^2 - 1)$;
8. Построить линейный порядок: а) на множестве \mathbb{N}^2 ; б) на множестве $\mathbb{N} \cap \mathbb{N}^2 \cap \dots \cap \mathbb{N}^n \cap \dots = \{\text{все конечные последовательности из натуральных чисел}\}$.
9. Пусть отношение ρ определено на множестве \mathbb{N}^2 (\mathbb{N} - множество натуральных чисел $\{1,2,3,\dots\}$): $\langle x,y \rangle \rho \langle u,v \rangle \Leftrightarrow x+v = y+u$. Доказать, что ρ - отношение эквивалентности.
10. Пусть $A = \{1,2,3\}$. На множестве $\mathcal{P}(A)$ задано бинарное отношение $X \rho Y \Leftrightarrow$ "множества X и Y имеют равное количество элементов". Доказать, что это отношение эквивалентности и найдите классы эквивалентности.

Вариант 9

1. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\{1,2\} ? \{1, 2, \{1,2\}\}$;
2. Перечислите все элементы каждого из следующих множеств:
 $\{x \mid x \subseteq \emptyset\}$.
3. Приведите пример множеств A , B и C таких, чтобы выполнялись условия $A \in B$, $B \notin C$, $A \subseteq C$.
4. Для бинарного отношения $\rho = \{ \langle x, y \rangle \mid x^2 + y^2 < 1 \}$ найдите D_ρ и R_ρ .
5. Докажите тождество $(A \setminus B) \times C = (A \times C) \setminus (B \times C)$.
6. Пусть $A = \{a_1, a_2, \dots, a_n\}$ - конечное множество. Определим отображение $f: P(A) \rightarrow \{0,1\}^n$ следующим образом
 $f(B) = \langle \alpha_1, \alpha_2, \dots, \alpha_n \rangle$, где $\alpha_i = 0$, если $a_i \notin B$, и $\alpha_i = 1$, если $a_i \in B$.
 Докажите, что f - биекция.
7. Какие отображения инъективны, сюръективны?
 $f: \mathbb{R} \rightarrow \mathbb{R}, x \mapsto 2^{3x+1}$;
8. Пусть отношение ρ определено на множестве \mathbb{N}^2 (\mathbb{N} - множество натуральных чисел $\{1,2,3,\dots\}$): $\langle x, y \rangle \rho \langle u, v \rangle \Leftrightarrow x+v = y+u$. Доказать, что ρ - отношение эквивалентности.
9. Построить линейный порядок: а) на множестве \mathbb{N}^2 ; б) на множестве $\mathbb{N} \cap \mathbb{N}^2 \cap \dots \cap \mathbb{N}^n \cap \dots = \{\text{все конечные последовательности из натуральных чисел}\}$.
10. Если ρ - частичный порядок на X , то ρ^{-1} также частичный порядок на X .

Вариант 10

1. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\{1\} ? \{1, \{1,2\}\}$;
2. Перечислите все подмножества множества A :
 $A = \{\{1,2\}, \{3\}, 1\}$;
3. Доказать тождество $A \setminus (B \setminus C) = (A \setminus B) \cup (A \cap C)$.
4. Пусть $A = \{1,2,3,4,5\}$, $B = \{\{1\}, \{1,2\}, \{2,5\}, \{3\}\}$. Для бинарного отношения $\rho = \{ \langle a, X \rangle \in A \times B \mid a \in X \}$ найдите D_ρ и R_ρ .
5. Докажите, что $(A \times B) \cap (C \times D) \subseteq (A \cap C) \times (B \cap D)$.
6. Найдите прообраз множества $\{0\}$ при следующих отображениях $\mathbb{R} \rightarrow \mathbb{R}$:
 д) $x \mapsto \sin(x)$;
 е) $x \mapsto \lg(x^2+1)$;
 ф) $x \mapsto x^2+2x+1$.
7. Какие отображения инъективны, сюръективны?

$f: \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}, \langle a, b \rangle \rightarrow a+b, \mathbb{Z}$ - множество целых чисел;

8. Если ρ и φ - отношения эквивалентности на X , то $\rho \cup \varphi$ также отношение эквивалентности на X .
9. Докажите, что отношение делимости на множестве натуральных чисел \mathbb{N} является отношением частичного порядка. Является ли это отношение линейным порядком? Является ли отношением частичного порядка отношение делимости на множестве целых чисел \mathbb{Z} ?
10. Если ρ и φ - отношения эквивалентности на X , то $\rho \cap \varphi$ - отношение эквивалентности на $X \Leftrightarrow \rho \cap \varphi = \rho \perp \varphi$.

Вариант 11

1. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\{1,2\} ? \{1,2, \{1\}, \{2\}\}$;
2. Перечислите все подмножества множества A :
 $A = \{\{1\}, \{2\}, 1\}$;
3. Равны ли два множества:
 $\{\{1,2\}, \{2,3\}\}$ и $\{1,2,3\}$?
4. Какими свойствами обладает отношение $x \rho y \Leftrightarrow x^2 + x = y^2 + y$, определенное на множестве действительных чисел?
5. Для бинарного отношения $\rho = \{\langle x, y \rangle \mid x^2 + y^2 < 1\}$ найдите D_ρ и R_ρ .
6. Какие отображения инъективны, сюръективны?
 $f: \mathbb{R} \rightarrow \mathbb{R}, x \mapsto x^2 + 3x + 5$;
7. Какие отображения инъективны, сюръективны?
 $f: \mathbb{Z} \rightarrow \mathbb{Z} \times \mathbb{Z}, a \mapsto \langle a, a \rangle$;
8. Если ρ и φ - отношения эквивалентности на X , то $\rho \cap \varphi$ - отношение эквивалентности на $X \Leftrightarrow \rho \cap \varphi = \rho \perp \varphi$.
9. Докажите, что $M = \{\{1\}, \{2,5\}, \{3\}, \{4,6,7\}\}$ - разбиение множества $A = \{1,2,3,4,5,6,7\}$. Перечислите все элементы отношения эквивалентности ρ , соответствующего разбиению M .
10. Построить линейный порядок: а) на множестве \mathbb{N}^2 ; б) на множестве $\mathbb{N} \cap \mathbb{N}^2 \cap \dots \cap \mathbb{N}^n \cap \dots = \{\text{все конечные последовательности из натуральных чисел}\}$.

Вариант 12

1. Перечислить все элементы каждого из следующих множеств:
 $\{x \mid x \subseteq \emptyset\}$.
2. Перечислите все подмножества множества A :
 $A = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}$.
3. Равны ли два множества:
 $\{2,3\}, \{3,4\}$ и $\{2,3,4\}$?

4. Какими свойствами обладает отношение ρ , определенное на множестве всех прямых плоскости: $x \rho y \Leftrightarrow x$ пересекается с y ?
5. Пусть $A = \{0, 1\}$. Перечислите элементы множеств A^2, A^3 .
6. Какие отображения инъективны, сюръективны?
 $f: \mathbb{R} \rightarrow \mathbb{R}, x \rightarrow x^{15}(x^2-1)$;
7. Укажите все сюръективные отображения множества $A = \{1, 2, 3\}$ на множество $B = \{a, b\}$.
8. Если ρ - частичный порядок на X , то ρ^{-1} также частичный порядок на X .
9. Пусть $A = \{1, 2, 3\}$. На множестве $P(A)$ задано бинарное отношение $X \rho Y \Leftrightarrow$ "множества X и Y имеют равное количество элементов". Доказать, что это отношение эквивалентности и найдите классы эквивалентности.
10. Если ρ и φ - отношения эквивалентности на X , то $\rho \cup \varphi$ также отношение эквивалентности на X .

Вариант 13

1. Перечислить все элементы каждого из следующих множеств:
 $\{x \mid x \subseteq \{1, 2, 3\}\}$;
2. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\{1, 2\} ? \{1, 2, \{1\}, \{2\}\}$;
3. Равны ли два множества:
 $\{4, 5\}, \{5, 6\}$ и $\{4, 5, 6\}$?
4. Какими свойствами обладает отношение ρ , определенное на множестве всех прямых плоскости: $x \rho y \Leftrightarrow x$ не пересекается с y ?
5. Определим упорядоченную пару $\langle a, b \rangle$ как множество $\{\{a\}, \{a, b\}\}$. Убедимся, что такое формальное теоретико-множественное определение вполне соответствует нашему неформальному определению упорядоченной пары. Для этого достаточно доказать, что для любых элементов $\langle a, b \rangle = \langle c, d \rangle \Leftrightarrow a = c, b = d$.
6. Какие отображения инъективны, сюръективны?
 $f: \mathbb{R} \rightarrow \mathbb{R}, x \rightarrow 2^{3x+1}$;
7. Найдите все отображения множества $A = \{1, 2\}$ в себя, укажите, какие из них инъективные, сюръективные.
8. Пусть $A = \{1, 2, 3\}$. На множестве $P(A)$ задано бинарное отношение $X \rho Y \Leftrightarrow$ "множества X и Y имеют равное количество элементов". Доказать, что это отношение эквивалентности и найдите классы эквивалентности.
9. Построить линейный порядок: а) на множестве \mathbb{N}^2 ; б) на множестве $\mathbb{N} \cap \mathbb{N}^2 \cap \dots \cap \mathbb{N}^n \cap \dots = \{\text{все конечные последовательности из натуральных чисел}\}$.

10. Пусть отношение ρ определено на множестве N^2 (N - множество натуральных чисел $\{1, 2, 3, \dots\}$): $\langle x, y \rangle \rho \langle u, v \rangle \Leftrightarrow x+v = y+u$. Доказать, что ρ - отношение эквивалентности.

Вариант 14

1. Перечислить все элементы каждого из следующих множеств:
 $\{x \mid x \subseteq \{1, 2\}\}$;
2. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\{1\} ? \{1, \{1, 2\}\}$;
3. Доказать, что если конечное множество A содержит n элементов, то множество-степень $P(A)$ содержит 2^n элементов.
4. Пусть ρ - отношение на множестве X . Докажите:
 ρ симметрично $\Leftrightarrow \rho^{-1} = \rho$;
5. Пусть $A \subseteq C$, $B \subseteq C$. Докажите, что $A \times B = (A \times C) \cup (C \times B)$.
6. Какие отображения инъективны, сюръективны?
 $f: Z \times Z \rightarrow Z, \langle a, b \rangle \rightarrow a+b$, Z - множество целых чисел;
7. Пусть X - конечное множество и отображение $f: X \rightarrow X$ инъективно. Доказать, что тогда f биективно.
8. Докажите, что $M = \{\{1\}, \{2, 5\}, \{3\}, \{4, 6, 7\}\}$ - разбиение множества $A = \{1, 2, 3, 4, 5, 6, 7\}$. Перечислите все элементы отношения эквивалентности ρ , соответствующего разбиению M .
9. Докажите, что отношение делимости на множестве натуральных чисел N является отношением частичного порядка. Является ли это отношение линейным порядком? Является ли отношением частичного порядка отношение делимости на множестве целых чисел Z ?
10. Если ρ и φ - отношения эквивалентности на X , то $\rho \cap \varphi$ - отношение эквивалентности на $X \Leftrightarrow \rho \cap \varphi = \rho \upharpoonright \varphi$.

Вариант 15

1. Перечислить все элементы каждого из следующих множеств:
 $\{x \mid x \subseteq \{1\}\}$;
2. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\{1, 2\} ? \{1, 2, \{1, 2\}\}$;
3. Доказать, что для любых множеств A и B имеем $A \cap (A \cup B) = A$.
4. Пусть ρ - отношение на множестве X . Докажите:
 ρ транзитивно $\Leftrightarrow \rho \upharpoonright \rho \subseteq \rho$;
5. Докажите, что подмножество C множества $A \times B$ является прямым произведением некоторого подмножества A_1 множества A и подмножества B_1

множества B тогда и только тогда, когда для любых $\langle a, b \rangle \in C$, $\langle c, d \rangle \in C$ следует, что $\langle a, d \rangle$, $\langle c, b \rangle \in C$.

6. Какие отображения инъективны, сюръективны?

$$f: Z \rightarrow Z \times Z, a \rightarrow \langle a, a \rangle;$$

7. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{A \cup B}(x) = \chi_A(x) \vee \chi_B(x);$$

8. Докажите, что отношение делимости на множестве натуральных чисел N является отношением частичного порядка. Является ли это отношение линейным порядком? Является ли отношением частичного порядка отношение делимости на множестве целых чисел Z ?

9. Докажите, что $M = \{\{1\}, \{2,5\}, \{3\}, \{4,6,7\}\}$ - разбиение множества $A = \{1,2,3,4,5,6,7\}$. Перечислите все элементы отношения эквивалентности ρ , соответствующего разбиению M .

10. Если ρ - частичный порядок на X , то ρ^{-1} также частичный порядок на X .

Вариант 16

1. Доказать, что для любых множеств A и B имеем $A \cap (A \cup B) = A$.

2. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\emptyset ? \{1,2,\{1\},\{\emptyset\}\}$;

3. Перечислить все элементы каждого из следующих множеств:

$$\{x \mid x \subseteq \{1\}\};$$

4. Пусть ρ - отношение на множестве X . Докажите:

$$\rho \text{ рефлексивно} \Rightarrow \rho \subseteq \rho \cup \rho;$$

5. Пусть A, B, C, D - непустые множества. Докажите, что

$$A \subseteq B \text{ и } C \subseteq D \Leftrightarrow A \times C \subseteq B \times D,$$

6. Какие отображения инъективны, сюръективны?

$$f: P(A) \rightarrow N, f(X) = \text{количество элементов в } X, N - \text{множество неотрицательных целых чисел, } A - \text{конечное множество.}$$

7. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{A \cap B}(x) = \chi_A(x) \wedge \chi_B(x);$$

8. Построить линейный порядок: а) на множестве N^2 ; б) на множестве $N \cap N^2 \cap \dots \cap N^n \cap \dots = \{\text{все конечные последовательности из натуральных чисел}\}$.
9. Пусть $A = \{1, 2, 3\}$. На множестве $P(A)$ задано бинарное отношение $X \rho Y \Leftrightarrow$ "множества X и Y имеют равное количество элементов". Доказать, что это отношение эквивалентности и найдите классы эквивалентности.
10. Докажите, что $M = \{\{1\}, \{2, 5\}, \{3\}, \{4, 6, 7\}\}$ - разбиение множества $A = \{1, 2, 3, 4, 5, 6, 7\}$. Перечислите все элементы отношения эквивалентности ρ , соответствующего разбиению M .

Вариант 17

1. Доказать, что если конечное множество A содержит n элементов, то множество-степень $P(A)$ содержит 2^n элементов.
2. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\emptyset ? \{\{\emptyset\}\}$;
3. Перечислите все подмножества множества A :
 $A = \{\{1, 2\}, \{3\}, 1\}$;
4. Пусть ρ - отношение на множестве X . Докажите:
 ρ рефлексивно и транзитивно $\Rightarrow \rho = \rho \cup \rho$.
5. Пусть A, B, C, D - непустые множества. Докажите, что
 $A \times C = B \times D \Leftrightarrow A = B \text{ и } C = D$.
6. Укажите все сюръективные отображения множества $A = \{1, 2, 3\}$ на множество $B = \{a, b\}$.
7. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{\bar{A}}(x) = 1 - \chi_A(x);$$

8. Пусть отношение ρ определено на множестве N^2 (N - множество натуральных чисел $\{1, 2, 3, \dots\}$): $\langle x, y \rangle \rho \langle u, v \rangle \Leftrightarrow x + v = y + u$. Доказать, что ρ - отношение эквивалентности.
9. Если ρ - частичный порядок на X , то ρ^{-1} также частичный порядок на X .
10. Построить линейный порядок: а) на множестве N^2 ; б) на множестве $N \cap N^2 \cap \dots \cap N^n \cap \dots = \{\text{все конечные последовательности из натуральных чисел}\}$.

Вариант 18

1. Равны ли два множества:
 $\{4,5\}$, $\{5,6\}$ и $\{4,5,6\}$?
2. Вставьте между множествами символ \in или \subseteq так, чтобы получилось истинное высказывание: $\emptyset ? \{\emptyset\}$.
3. Перечислить все элементы каждого из следующих множеств:
 $\{x \mid x \subseteq \emptyset\}$.
4. Какова характеристическая особенность декартовой диаграммы рефлексивного (симметричного, антисимметричного) отношения, определенного на множестве вещественных чисел \mathbf{R} .
5. Докажите тождество $(A \setminus B) \times C = (A \times C) \setminus (B \times C)$.
6. Найдите все отображения множества $A = \{1,2\}$ в себя, укажите, какие из них инъективные, сюръективные.
7. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{A \setminus B}(x) = \chi_A(x) - \chi_A(x) \chi_B(x).$$

8. Если ρ и φ - отношения эквивалентности на X , то $\rho \cup \varphi$ также отношение эквивалентности на X .
9. Если ρ и φ - отношения эквивалентности на X , то $\rho \cap \varphi$ - отношение эквивалентности на $X \Leftrightarrow \rho \cap \varphi = \rho \perp \varphi$.
10. Докажите, что отношение делимости на множестве натуральных чисел \mathbf{N} является отношением частичного порядка. Является ли это отношение линейным порядком? Является ли отношением частичного порядка отношение делимости на множестве целых чисел \mathbf{Z} ?

Вариант 19

1. Равны ли два множества:
 $\{2,3\}$, $\{3,4\}$ и $\{2,3,4\}$?
2. Докажите следующее утверждение: $A \subset B$ и $B \subseteq C \Rightarrow A \subset C$.
3. Найдите множество X , удовлетворяющее следующему условию:
 $A \setminus (A \setminus X) = \emptyset$.
4. Пусть $\rho_1 = \{ \langle x, y \rangle \in \mathbf{R} \times \mathbf{R} \mid x \square y > 0 \}$, $\rho_2 = \{ \langle x, y \rangle \in \mathbf{R} \times \mathbf{R} \mid x + y \text{ - целое число} \}$.
Найти $\rho_1 \perp \rho_2$.
5. Докажите, что $(A \times B) \cap (C \times D) \subseteq (A \cap C) \times (B \cap D)$.
6. Пусть X - конечное множество и отображение $f: X \rightarrow X$ инъективно. Доказать, что тогда f биективно.
7. Пусть $f: X \rightarrow Y$ и $A \subseteq X$. Образом множества A при отображении f называется множество $f(A) = \{y \mid y = f(x), x \in A\}$.

Пусть $B \subseteq Y$. Прообразом множества B при отображении f называется множество $f^{-1}(B) = \{x \in X \mid f(x) \in B\}$.

Доказать, что $f^{-1}(A \cap B) = f^{-1}(A) \cap f^{-1}(B)$.

8. Если ρ и φ - отношения эквивалентности на X , то $\rho \cap \varphi$ - отношение эквивалентности на $X \Leftrightarrow \rho \cap \varphi = \rho \perp \varphi$.
9. Если ρ и φ - отношения эквивалентности на X , то $\rho \cup \varphi$ также отношение эквивалентности на X .
10. Пусть $A = \{1, 2, 3\}$. На множестве $P(A)$ задано бинарное отношение $X \rho Y \Leftrightarrow$ "множества X и Y имеют равное количество элементов". Доказать, что это отношение эквивалентности и найдите классы эквивалентности.

Вариант 20

1. Равны ли два множества:
 $\{\{1, 2\}, \{2, 3\}\}$ и $\{1, 2, 3\}$?
2. Найдите $A \cap B$, $A \cup B$, $A \setminus B$, $B \setminus A$, \overline{A} , \overline{B} для
 $A = \{1, 2, 3\}$, $B = \{2, 3, 4, 5\}$, $U = \{0, 1, \dots, 9\}$;
3. Приведите пример множеств A , B и C таких, чтобы выполнялись условия $A \in B$, $B \notin C$, $A \subseteq C$.
4. Пусть $\rho_1 = \{\langle x, y \rangle \in \mathbb{R} \times \mathbb{R} \mid x \square y > 0\}$, $\rho_2 = \{\langle x, y \rangle \in \mathbb{R} \times \mathbb{R} \mid x = y^2\}$. Найти $\rho_1 \perp \rho_2$.
5. Для бинарного отношения $\rho = \{\langle x, y \rangle \mid x^2 + y^2 < 1\}$ найдите D_ρ и R_ρ .
6. Характеристическая функция множества A определяется следующим образом

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A \\ 0, & \text{если } x \notin A \end{cases}$$

Доказать:

$$\chi_{A \cup B}(x) = \chi_A(x) \vee \chi_B(x);$$

7. Какие отображения инъективны, сюръективны?
 $f: \mathbb{R} \rightarrow \mathbb{R}, x \mapsto x^2 + 3x + 5$;
8. Если ρ - частичный порядок на X , то ρ^{-1} также частичный порядок на X .
9. Пусть отношение ρ определено на множестве \mathbb{N}^2 (\mathbb{N} - множество натуральных чисел $\{1, 2, 3, \dots\}$): $\langle x, y \rangle \rho \langle u, v \rangle \Leftrightarrow x + v = y + u$. Доказать, что ρ - отношение эквивалентности.
10. Докажите, что $M = \{\{1\}, \{2, 5\}, \{3\}, \{4, 6, 7\}\}$ - разбиение множества $A = \{1, 2, 3, 4, 5, 6, 7\}$. Перечислите все элементы отношения эквивалентности ρ , соответствующего разбиению M .

КОНТРОЛЬНАЯ № 2

Вариант 1

1. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
"Для того, чтобы x было нечетным, достаточно, чтобы x было простым";
2. При каких значениях переменных x, y, z формула $((x \supset (y \& z)) \supset (\neg y \supset \neg x)) \supset \neg y$ ложна?
3. Является ли тавтологией формула $((p \supset q) \& (\neg r \supset \neg q) \& (t \supset \neg r)) \supset (p \supset \neg t)$?
4. Доказать выполнимость формулы $\neg(p \supset \neg p)$.
5. Пусть даны предикаты на множестве целых чисел
 $E(x) \equiv$ "x - четное число" и $D(x, y) \equiv$ "y делится на x"
Переведите на обычный язык формулу
 $\exists x(E(x) \vee D(6, x))$.
6. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv$ "y делится на x";
 $G(x, y, z) \equiv$ "z - наибольший общий делитель x и y".
Запишите утверждения на языке логики предикатов:
"если x делится на y и y делится на z, то x делится на z";
7. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, =) запишите на языке логики предикатов следующие высказывания о действительных числах:
"система уравнений
$$\begin{cases} x + y = 1 \\ 2x + 2y = 0 \end{cases}$$
не имеет решения";
8. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, =), каждое из следующих высказываний запишите при помощи логических символов, определите, истинное оно или ложное:
"для любых действительных чисел x и y , если $x < y$ и $y \neq 0$, то $x/y < 1$ ".

Вариант 2

1. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
"Если идет дождь, то дует ветер".
2. Доказать, что $A \sim B \equiv \neg A \sim \neg B$.
3. Проверьте, что формула $((p \supset q) \& p) \supset q$ является тавтологией.
3. При каких значениях переменных x, y, z формула $((x \supset (y \& z)) \supset (\neg y \supset \neg x)) \supset \neg y$ ложна?
5. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv$ "y делится на x";
 $I(x, y) \equiv$ "x равно y";
 $P(x) \equiv$ "x - простое число".
Переведите на обычный язык формулу
 $\forall x (\neg I(1, x) \supset \exists y (P(y) \& D(y, x)))$.
6. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv$ "y делится на x";
 $G(x, y, z) \equiv$ "z - наибольший общий делитель x и y".
Запишите утверждения на языке логики предикатов:
"если d - наибольший общий делитель a и b, то a и b делятся на d и d делится на любой общий делитель a и b".
7. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, $=$) запишите на языке логики предикатов следующие высказывания о действительных числах:
"существует ровно одно положительное значение квадратного корня из положительного числа".
8. Используя только предикаты " $x = y$ " и $D(x, y) \equiv$ "y делится на x", запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
"x - простое число";

Вариант 3

1. Обосновать метод доказательства "разбором случаев": для того, чтобы доказать формулу $(A_1 \vee A_2 \vee \dots \vee A_n) \supset B$ необходимо и достаточно доказать формулу $(A_1 \supset B) \& (A_2 \supset B) \& \dots \& (A_n \supset B)$.
2. Построить формулу от трех переменных, которая истинна в том и только том случае, когда ровно две переменные ложны.
3. Что можно сказать об истинностном значении высказывания $(\neg p \& \neg q) \sim (p \vee q)$, если $p \supset q$ ложно?
4. Доказать, что $A \sim B \equiv \neg A \sim \neg B$.
5. Пусть даны предикаты на множестве натуральных чисел:

$D(x,y) \equiv$ "у делится на х";

$P(x) \equiv$ "х - простое число".

Переведите на обычный язык формулы:

$\forall x (P(x) \supset \neg D(2,x))$;

и ответьте истинны они или нет.

6. Пользуясь знаками арифметических операций (+, ×) и отношений (<, =) запишите на языке логики предикатов следующие высказывания о действительных числах:
"если произведение двух чисел равно 0, то хотя бы один из сомножителей равен 0";
7. Пользуясь знаками арифметических операций (+, ×) и отношений (<, =), каждое из следующих высказываний запишите при помощи логических символов, определите, истинное оно или ложное:
"существует такое целое х, что $x^2 - 4 = 0$ ";
8. Используя только предикаты " $x = y$ " и $D(x,y) \equiv$ "у делится на х", запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
"а и b - взаимно простые числа".

Вариант 4

1. Что можно сказать об истинностном значении высказывания $(\neg p \& \neg q) \sim (p \vee q)$, если $p \supset q$ ложно?
2. Выразить $A \vee B$ через A, B и символ \supset .
3. Обосновать метод доказательства "разбором случаев": для того, чтобы доказать формулу $(A_1 \vee A_2 \vee \dots \vee A_n) \supset B$ необходимо и достаточно доказать формулу $(A_1 \supset B) \& (A_2 \supset B) \& \dots \& (A_n \supset B)$.
4. Построить формулу от трех переменных, которая истинна в том и только том случае, когда ровно две переменные ложны.
5. Пусть даны предикаты на множестве натуральных чисел:
 $D(x,y) \equiv$ "у делится на х";
 $P(x) \equiv$ "х - простое число".
Переведите на обычный язык формулы:
 $\forall x \forall y (\neg P(x) \supset D(x,y))$
и ответьте истинны они или нет.
6. Пользуясь знаками арифметических операций (+, ×) и отношений (<, =) запишите на языке логики предикатов следующие высказывания о действительных числах:
б) "система уравнений

$$\begin{cases} x + y = 1 \\ 2x + 2y = 0 \end{cases}$$
 не имеет решения";

7. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, $=$), каждое из следующих высказываний запишите при помощи логических символов, определите, истинное оно или ложное:
 "для любых действительных чисел x и y , если $x < y$ и $y \neq 0$, то $x/y < 1$ ".
8. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
 Записать формулу с двумя свободными переменными - истинную тогда и только тогда, когда x и y являются простыми числами-близнецами.

Вариант 5

1. Проверьте, что формула $((p \supset q) \& p) \supset q$ является тавтологией.
2. Мальчик решил в воскресенье закончить чтение книги, сходить в музей или кино, а если будет хорошая погода - пойти на реку выкупаться. В каком случае можно сказать, что решение мальчика не выполнено? Запишите формулу истинную тогда и только тогда, когда решение мальчика не выполнено (отрицания должны содержаться лишь в простых высказываниях).
3. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
 "Если идет дождь, то дует ветер".
4. Выразить $A \vee B$ через A, B и символ \supset .
5. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv "y \text{ делится на } x"$;
 $G(x, y, z) \equiv "z - \text{наибольший общий делитель } x \text{ и } y"$.
 Запишите утверждения на языке логики предикатов:
 "если x делится на y и y делится на z , то x делится на z ";
6. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, $=$) запишите на языке логики предикатов следующие высказывания о действительных числах:
 "существует ровно одно положительное значение квадратного корня из положительного числа".
7. Используя только предикаты " $x = y$ " и $D(x, y) \equiv "y \text{ делится на } x"$, запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
 " x - простое число";
8. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
 Записать предложение, выражающее не существование 1.

Вариант 6

1. Является ли тавтологией формула $((p \supset q) \& (\neg r \supset \neg q) \& (t \supset \neg r)) \supset (p \supset \neg t)$?
- 2.
3. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
"Для того, чтобы x было нечетным, достаточно, чтобы x было простым";
4. Мальчик решил в воскресенье закончить чтение книги, сходить в музей или кино, а если будет хорошая погода - пойти на реку выкупаться. В каком случае можно сказать, что решение мальчика не выполнено? Запишите формулу истинную тогда и только тогда, когда решение мальчика не выполнено (отрицания должны содержаться лишь в простых высказываниях).
5. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv$ "у делится на x ";
 $G(x, y, z) \equiv$ " z - наибольший общий делитель x и y ".
Запишите утверждения на языке логики предикатов:
"если d - наибольший общий делитель a и b , то a и b делятся на d и d делится на любой общий делитель a и b ".
6. Пользуясь знаками арифметических операций $(+, \times)$ и отношений $(<, =)$, каждое из следующих высказываний запишите при помощи логических символов, определите, истинное оно или ложное:
"существует такое целое x , что $x^2 - 4 = 0$ ";
7. Используя только предикаты " $x = y$ " и $D(x, y) \equiv$ "у делится на x ", запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
" a и b - взаимно простые числа".
8. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
Записать высказывание, выражающее бесконечность множества простых чисел-близнецов.

Вариант 7

1. Доказать выполнимость формулы $\neg(p \supset \neg p)$.
2. Проверьте, что формула $((p \supset q) \& p) \supset q$ является тавтологией.
3. Подозреваются в совершении преступления Жан и Пьер. На суде выступили четыре свидетеля со следующими заявлениями:
 - а) Пьер не виноват;
 - б) Жан не виноват;
 - в) из первых двух показаний по меньшей мере одно истинно;
 - г) показания третьего свидетеля ложны.

Следствие установило, что четвертый свидетель прав. Кто преступники?

4. Преобразовать к ДНФ формулу $\neg(x \vee y) \& (x \supset y)$.
5. Пользуясь знаками арифметических операций (+, ×) и отношений (<, =) запишите на языке логики предикатов следующие высказывания о действительных числах:
"если произведение двух чисел равно 0, то хотя бы один из сомножителей равен 0";
6. Пользуясь знаками арифметических операций (+, ×) и отношений (<, =), каждое из следующих высказываний запишите при помощи логических символов, определите, истинное оно или ложное:
"для любых действительных чисел x и y, если $x < y$ и $y \neq 0$, то $x/y < 1$ ".
7. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
Записать формулу с двумя свободными переменными - истинную тогда и только тогда, когда x и y являются простыми числами-близнецами.
8. Пусть на некотором универсальном множестве U задан предикат $Q(x, y) \equiv "x \subseteq y"$. Запишите, что "множество x есть пересечение множеств y и z".

Вариант 8

1. При каких значениях переменных x, y, z формула $((x \supset (y \& z)) \supset (\neg y \supset \neg x)) \supset \neg y$ ложна?
2. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
"Для того, чтобы x было нечетным, достаточно, чтобы x было простым";
3. Преобразовать к ДНФ формулу $\neg(x \vee y) \& (x \supset y)$.
4. Подозреваются в совершении преступления Жан и Пьер. На суде выступили четыре свидетеля со следующими заявлениями:
 - 1) Пьер не виноват;
 - 2) Жан не виноват;
 - 3) из первых двух показаний по меньшей мере одно истинно;
 - 4) показания третьего свидетеля ложны.
 Следствие установило, что четвертый свидетель прав. Кто преступники?
5. Пользуясь знаками арифметических операций (+, ×) и отношений (<, =) запишите на языке логики предикатов следующие высказывания о действительных числах:
 - б) "система уравнений

$$\begin{cases} x + y = 1 \\ \end{cases}$$

$$2x + 2y = 0$$

не имеет решения";

6. Используя только предикаты " $x = y$ " и $D(x, y) \equiv "y \text{ делится на } x"$, запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
" x - простое число";
7. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
Записать предложение, выражающее не существование 1.
8. Пусть даны предикаты на множестве целых чисел
 $E(x) \equiv "x \text{ - четное число}"$ и $D(x, y) \equiv "y \text{ делится на } x"$
Переведите на обычный язык формулу
 $\exists x(E(x) \vee D(6, x))$.

Вариант 9

1. Доказать, что $A \sim B \equiv \neg A \sim \neg B$.
2. Обосновать метод доказательства "разбором случаев": для того, чтобы доказать формулу $(A_1 \vee A_2 \vee \dots \vee A_n) \supset B$ необходимо и достаточно доказать формулу $(A_1 \supset B) \& (A_2 \supset B) \& \dots \& (A_n \supset B)$.
3. Мальчик решил в воскресенье закончить чтение книги, сходить в музей или кино, а если будет хорошая погода - пойти на реку выкупаться. В каком случае можно сказать, что решение мальчика не выполнено? Запишите формулу истинную тогда и только тогда, когда решение мальчика не выполнено (отрицания должны содержаться лишь в простых высказываниях).
4. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
"Для того, чтобы x было нечетным, достаточно, чтобы x было простым";
5. Пользуясь знаками арифметических операций $(+, \times)$ и отношений $(<, =)$ запишите на языке логики предикатов следующие высказывания о действительных числах:
"существует ровно одно положительное значение квадратного корня из положительного числа".
6. Используя только предикаты " $x = y$ " и $D(x, y) \equiv "y \text{ делится на } x"$, запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
" a и b - взаимно простые числа".
7. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
Записать высказывание, выражающее бесконечность множества простых чисел-близнецов.

8. Пусть даны предикаты на множестве натуральных чисел:

$D(x,y) \equiv$ "y делится на x";

$I(x,y) \equiv$ "x равно y";

$P(x) \equiv$ "x - простое число".

Переведите на обычный язык формулу

$\forall x (\neg I(1,x) \supset \exists y (P(y) \& D(y,x)))$.

Вариант 10

1. Построить формулу от трех переменных, которая истинна в том и только том случае, когда ровно две переменные ложны.
2. Что можно сказать об истинностном значении высказывания $(\neg p \& \neg q) \sim (p \vee q)$, если $p \supset q$ ложно?
3. Выразить $A \vee B$ через A, B и символ \supset .
4. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
"Если идет дождь, то дует ветер".
5. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, $=$), каждое из следующих высказываний запишите при помощи логических символов, определите, истинное оно или ложное:
"существует такое целое x, что $x^2 - 4 = 0$ ";
6. На множестве натуральных чисел заданы предикаты $S(x,y,z) \equiv "x + y = z"$ и $P(x,y,z) \equiv "x \times y = z"$.
Записать формулу с двумя свободными переменными - истинную тогда и только тогда, когда x и y являются простыми числами-близнецами.
7. Пусть на некотором универсальном множестве U задан предикат $Q(x,y) \equiv "x \subseteq y"$. Запишите, что "множество x есть пересечение множеств y и z".
8. Пусть даны предикаты на множестве натуральных чисел:
 $D(x,y) \equiv$ "y делится на x";
 $P(x) \equiv$ "x - простое число".
Переведите на обычный язык формулы:
 $\forall x (P(x) \supset \neg D(2,x))$;
и ответьте истинны они или нет.

Вариант 11

1. Выразить $A \vee B$ через A, B и символ \supset .
2. Проверьте, что формула $((p \supset q) \& p) \supset q$ является тавтологией.
3. Построить формулу от трех переменных, которая истинна в том и только том случае, когда ровно две переменные ложны.

4. Обосновать метод доказательства "разбором случаев": для того, чтобы доказать формулу $(A_1 \vee A_2 \vee \dots \vee A_n) \supset B$ необходимо и достаточно доказать формулу $(A_1 \supset B) \& (A_2 \supset B) \& \dots \& (A_n \supset B)$.
5. Пользуясь знаками арифметических операций (+, ×) и отношений (<, =), каждое из следующих высказываний запишите при помощи логических символов, определите, истинное оно или ложное:
"для любых действительных чисел x и y, если $x < y$ и $y \neq 0$, то $x/y < 1$ ".
6. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
Записать предложение, выражающее не существование 1.
7. Пусть даны предикаты на множестве целых чисел
 $E(x) \equiv "x - \text{четное число}"$ и $D(x, y) \equiv "y \text{ делится на } x"$
Переведите на обычный язык формулу
 $\exists x (E(x) \vee D(6, x))$.
8. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv "y \text{ делится на } x"$;
 $P(x) \equiv "x - \text{простое число}"$.
Переведите на обычный язык формулы:
 $\forall x \forall y (\neg P(x) \supset D(x, y))$
и ответьте истинны они или нет.

Вариант 12

1. Мальчик решил в воскресенье закончить чтение книги, сходить в музей или кино, а если будет хорошая погода - пойти на реку выкупаться. В каком случае можно сказать, что решение мальчика не выполнено? Запишите формулу истинную тогда и только тогда, когда решение мальчика не выполнено (отрицания должны содержаться лишь в простых высказываниях).
2. Является ли тавтологией формула $((p \supset q) \& (\neg r \supset \neg q) \& (t \supset \neg r)) \supset (p \supset \neg t)$?
3. Доказать, что $A \sim B \equiv \neg A \sim \neg B$.
4. Что можно сказать об истинностном значении высказывания $(\neg p \& \neg q) \sim (p \vee q)$, если $p \supset q$ ложно?
5. Используя только предикаты $"x = y"$ и $D(x, y) \equiv "y \text{ делится на } x"$, запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
"x - простое число";
6. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
Записать высказывание, выражающее бесконечность множества простых чисел-близнецов.
7. Пусть даны предикаты на множестве натуральных чисел:

$D(x,y) \equiv$ "у делится на x";

$I(x,y) \equiv$ "x равно y";

$P(x) \equiv$ "x - простое число".

Переведите на обычный язык формулу

$\forall x (\neg I(1,x) \supset \exists y (P(y) \& D(y,x)))$.

8. Пусть даны предикаты на множестве натуральных чисел:

$D(x,y) \equiv$ "у делится на x";

$G(x,y,z) \equiv$ "z - наибольший общий делитель x и y".

Запишите утверждения на языке логики предикатов:

"если x делится на y и y делится на z, то x делится на z";

Вариант 13

1. Преобразовать к ДНФ формулу $\neg(x \vee y) \& (x \supset y)$.
2. Доказать выполнимость формулы $\neg(p \supset \neg p)$.
3. При каких значениях переменных x, y, z формула $((x \supset (y \& z)) \supset (\neg y \supset \neg x)) \supset \neg y$ ложна?
4. Проверьте, что формула $((p \supset q) \& p) \supset q$ является тавтологией.
5. Используя только предикаты "x = y" и $D(x,y) \equiv$ "у делится на x", запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
"a и b - взаимно простые числа".
6. Пусть на некотором универсальном множестве U задан предикат $Q(x,y) \equiv$ "x \subseteq y". Запишите, что "множество x есть пересечение множеств y и z".
7. Пусть даны предикаты на множестве натуральных чисел:
 $D(x,y) \equiv$ "у делится на x";
 $P(x) \equiv$ "x - простое число".
Переведите на обычный язык формулы:
 $\forall x (P(x) \supset \neg D(2,x))$;
и ответьте истинны они или нет.
8. Пусть даны предикаты на множестве натуральных чисел:
 $D(x,y) \equiv$ "у делится на x";
 $G(x,y,z) \equiv$ "z - наибольший общий делитель x и y".
Запишите утверждения на языке логики предикатов:
"если d - наибольший общий делитель a и b, то a и b делятся на d и d делится на любой общий делитель a и b".

Вариант 14

1. Подозреваются в совершении преступления Жан и Пьер. На суде выступили четыре свидетеля со следующими заявлениями:
а) Пьер не виноват;

- б) Жан не виноват;
 в) из первых двух показаний по меньшей мере одно истинно;
 г) показания третьего свидетеля ложны.
 Следствие установило, что четвертый свидетель прав. Кто преступники?
2. При каких значениях переменных x, y, z формула $((x \supset (y \& z)) \supset (\neg y \supset \neg x)) \supset \neg y$ ложна?
 3. Доказать выполнимость формулы $\neg(p \supset \neg p)$.
 4. Является ли тавтологией формула $((p \supset q) \& (\neg r \supset \neg q) \& (t \supset \neg r)) \supset (p \supset \neg t)$?
 5. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
 Записать формулу с двумя свободными переменными - истинную тогда и только тогда, когда x и y являются простыми числами-близнецами.
 6. Пусть даны предикаты на множестве целых чисел
 $E(x) \equiv "x - \text{четное число}"$ и $D(x, y) \equiv "y \text{ делится на } x"$
 Переведите на обычный язык формулу
 $\exists x (E(x) \vee D(6, x))$.
 7. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv "y \text{ делится на } x";$
 $P(x) \equiv "x - \text{простое число}"$.
 Переведите на обычный язык формулы:
 $\forall x \forall y (\neg P(x) \supset D(x, y))$
 и ответьте истинны они или нет.
 8. Пользуясь знаками арифметических операций $(+, \times)$ и отношений $(<, =)$ запишите на языке логики предикатов следующие высказывания о действительных числах:
 "если произведение двух чисел равно 0, то хотя бы один из сомножителей равен 0";

Вариант 15

1. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
 "Если идет дождь, то дует ветер".
2. Доказать, что $A \sim B \equiv \neg A \sim \neg B$.
3. Является ли тавтологией формула $((p \supset q) \& (\neg r \supset \neg q) \& (t \supset \neg r)) \supset (p \supset \neg t)$?
4. Доказать выполнимость формулы $\neg(p \supset \neg p)$.
5. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
 Записать предложение, выражающее не существование 1.
6. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv "y \text{ делится на } x";$
 $I(x, y) \equiv "x \text{ равно } y";$

$P(x) \equiv "x - \text{простое число}"$.

Переведите на обычный язык формулу

$\forall x (\neg I(1, x) \supset \exists y (P(y) \& D(y, x)))$.

7. Пусть даны предикаты на множестве натуральных чисел:

$D(x, y) \equiv "y \text{ делится на } x"$;

$G(x, y, z) \equiv "z - \text{наибольший общий делитель } x \text{ и } y"$.

Запишите утверждения на языке логики предикатов:

а) "если x делится на y и y делится на z , то x делится на z ";

8. Пользуясь знаками арифметических операций $(+, \times)$ и отношений $(<, =)$ запишите на языке логики предикатов следующие высказывания о действительных числах:

"система уравнений

$$\begin{cases} x + y = 1 \\ 2x + 2y = 0 \end{cases}$$

не имеет решения";

Вариант 16

1. Обосновать метод доказательства "разбором случаев": для того, чтобы доказать формулу $(A_1 \vee A_2 \vee \dots \vee A_n) \supset B$ необходимо и достаточно доказать формулу $(A_1 \supset B) \& (A_2 \supset B) \& \dots \& (A_n \supset B)$.
2. Построить формулу от трех переменных, которая истинна в том и только том случае, когда ровно две переменные ложны.
3. Проверьте, что формула $((p \supset q) \& p) \supset q$ является тавтологией.
4. При каких значениях переменных x, y, z формула $((x \supset (y \& z)) \supset (\neg y \supset \neg x)) \supset \neg y$ ложна?
5. На множестве натуральных чисел заданы предикаты $S(x, y, z) \equiv "x + y = z"$ и $P(x, y, z) \equiv "x \times y = z"$.
Записать высказывание, выражающее бесконечность множества простых чисел-близнецов.
6. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv "y \text{ делится на } x"$;
 $P(x) \equiv "x - \text{простое число}"$.
Переведите на обычный язык формулы:
а) $\forall x (P(x) \supset \neg D(2, x))$;
и ответьте истинны они или нет.
7. Пусть даны предикаты на множестве натуральных чисел:
 $D(x, y) \equiv "y \text{ делится на } x"$;

$G(x,y,z) \equiv$ "z - наибольший общий делитель x и y".

Запишите утверждения на языке логики предикатов:

"если d - наибольший общий делитель a и b, то a и b делятся на d и d делится на любой общий делитель a и b".

8. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, $=$) запишите на языке логики предикатов следующие высказывания о действительных числах:
"существует ровно одно положительное значение квадратного корня из положительного число".

Вариант 17

1. Что можно сказать об истинностном значении высказывания $(\neg p \& \neg q) \sim (p \vee q)$, если $p \supset q$ ложно?
2. Выразить $A \vee B$ через A, B и символ \supset .
3. Является ли тавтологией формула $((p \supset q) \& (\neg r \supset \neg q) \& (t \supset \neg r)) \supset (p \supset \neg t)$?
4. Доказать, что $A \sim B \equiv \neg A \sim \neg B$.
5. Пусть на некотором универсальном множестве U задан предикат $Q(x,y) \equiv$ "x \subseteq y". Запишите, что "множество x есть пересечение множеств y и z".
6. Пусть даны предикаты на множестве натуральных чисел:
 $D(x,y) \equiv$ "y делится на x";
 $P(x) \equiv$ "x - простое число".
Переведите на обычный язык формулы:
 $\forall x \forall y (\neg P(x) \supset D(x,y))$
и ответьте истинны они или нет.
7. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, $=$) запишите на языке логики предикатов следующие высказывания о действительных числах:
"если произведение двух чисел равно 0, то хотя бы один из сомножителей равен 0";
8. Используя только предикаты "x = y" и $D(x,y) \equiv$ "y делится на x", запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
"x - простое число";

Вариант 18

1. Проверьте, что формула $((p \supset q) \& p) \supset q$ является тавтологией.
2. Мальчик решил в воскресенье закончить чтение книги, сходить в музей или кино, а если будет хорошая погода - пойти на реку выкупаться. В каком случае можно сказать, что решение мальчика не выполнено? Запишите формулу истинную тогда и только тогда, когда решение маль-

чика не выполнено (отрицания должны содержаться лишь в простых высказываниях).

3. Обосновать метод доказательства "разбором случаев": для того, чтобы доказать формулу $(A_1 \vee A_2 \vee \dots \vee A_n) \supset B$ необходимо и достаточно доказать формулу $(A_1 \supset B) \& (A_2 \supset B) \& \dots \& (A_n \supset B)$.
4. Построить формулу от трех переменных, которая истинна в том и только том случае, когда ровно две переменные ложны.
5. Пусть даны предикаты на множестве целых чисел
 $E(x) \equiv$ "x - четное число" и $D(x,y) \equiv$ "y делится на x"
 Переведите на обычный язык формулу
 $\exists x (E(x) \vee D(6,x))$.
6. Пусть даны предикаты на множестве натуральных чисел:
 $D(x,y) \equiv$ "y делится на x";
 $G(x,y,z) \equiv$ "z - наибольший общий делитель x и y".
 Запишите утверждения на языке логики предикатов:
 "если x делится на y и y делится на z, то x делится на z";
7. Пользуясь знаками арифметических операций (+, \times) и отношений (<, =) запишите на языке логики предикатов следующие высказывания о действительных числах: "система уравнений

$$\begin{cases} x + y = 1 \\ 2x + 2y = 0 \end{cases}$$
 не имеет решения";
8. Используя только предикаты " $x = y$ " и $D(x,y) \equiv$ "y делится на x", запишите при помощи логических символов следующие формулы от переменных, принимающих натуральные значения:
 "a и b - взаимно простые числа".

Вариант 19

1. Является ли тавтологией формула $((p \supset q) \& (\neg r \supset \neg q) \& (t \supset \neg r)) \supset (p \supset \neg t)$?
2. Преобразовать к ДНФ формулу $\neg(x \vee y) \& (x \supset y)$.
3. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
 "Если идет дождь, то дует ветер".
4. Выразить $A \vee B$ через A, B и символ \supset .
5. Пусть даны предикаты на множестве натуральных чисел:
 $D(x,y) \equiv$ "y делится на x";
 $I(x,y) \equiv$ "x равно y";
 $P(x) \equiv$ "x - простое число".
 Переведите на обычный язык формулу
 $\forall x (\neg I(1,x) \supset \exists y (P(y) \& D(y,x)))$.
6. Пусть даны предикаты на множестве натуральных чисел:

$D(x,y) \equiv$ "у делится на x";

$G(x,y,z) \equiv$ "z - наибольший общий делитель x и y".

Запишите утверждения на языке логики предикатов:

"если d - наибольший общий делитель a и b, то a и b делятся на d и d делится на любой общий делитель a и b".

7. Пользуясь знаками арифметических операций (+, \times) и отношений (<, =) запишите на языке логики предикатов следующие высказывания о действительных числах:
"существует ровно одно положительное значение квадратного корня из положительного числа".
8. Пусть на некотором универсальном множестве U задан предикат $Q(x,y) \equiv$ " $x \subseteq y$ ". Запишите, что "множество x есть пересечение множеств y и z".

Вариант 20

1. Доказать выполнимость формулы $\neg(p \supset \neg p)$.
2. Подозреваются в совершении преступления Жан и Пьер. На суде выступили четыре свидетеля со следующими заявлениями:
а) Пьер не виноват;
б) Жан не виноват;
в) из первых двух показаний по меньшей мере одно истинно;
г) показания третьего свидетеля ложны.
Следствие установило, что четвертый свидетель прав. Кто преступники?
3. Записать составные высказывания в виде формул, употребляя высказывательные переменные для обозначения простых высказываний:
"Для того, чтобы x было нечетным, достаточно, чтобы x было простым";
4. Мальчик решил в воскресенье закончить чтение книги, сходить в музей или кино, а если будет хорошая погода - пойти на реку выкупаться. В каком случае можно сказать, что решение мальчика не выполнено? Запишите формулу истинную тогда и только тогда, когда решение мальчика не выполнено (отрицания должны содержаться лишь в простых высказываниях).
5. Пусть даны предикаты на множестве натуральных чисел:
 $D(x,y) \equiv$ "у делится на x";
 $P(x) \equiv$ "x - простое число".
Переведите на обычный язык формулы:
 $\forall x (P(x) \supset \neg D(2,x))$;
и ответьте истинны они или нет.

6. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, $=$) запишите на языке логики предикатов следующие высказывания о действительных числах:
"если произведение двух чисел равно 0, то хотя бы один из сомножителей равен 0";
7. Пользуясь знаками арифметических операций (+, \times) и отношений ($<$, $=$), каждое из следующих высказываний запишите при помощи логических символов, определите, истинное оно или ложное:
"существует такое целое x , что $x^2 - 4 = 0$ ";
8. На множестве натуральных чисел заданы предикаты $S(x,y,z) \equiv "x + y = z"$ и $P(x,y,z) \equiv "x \times y = z"$.
Записать высказывание, выражающее бесконечность множества простых чисел-близнецов.