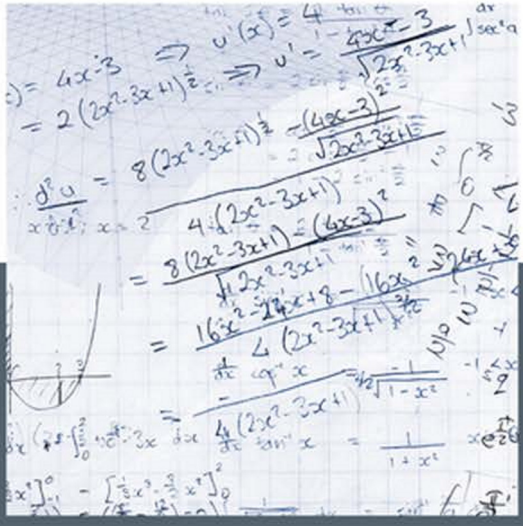
ВЫСШЕЕ ОБРАЗОВАНИЕ

ДИСКРЕТНАЯ МАТЕМАТИКА



И. А. Мальцев

ЛАНЬ

E.LANBOOK.COM

И. А. МАЛЬЦЕВ

ДИСКРЕТНАЯ МАТЕМАТИКА

УЧЕБНОЕ ПОСОБИЕ

Издание четвертое, стереотипное

ЛАНЬ

САНКТ-ПЕТЕРБУРГ · МОСКВА · КРАСНОДАР
2022

УДК 512
ББК 22.176я73

М 21 **Мальцев И. А.** Дискретная математика : учебное пособие для вузов / И. А. Мальцев. — 4-е изд., стер. — Санкт-Петербург : Лань, 2022. — 292 с. : ил. — Текст : непосредственный.

ISBN 978-5-507-45354-2

Книга содержит следующие разделы: теория множеств, комбинаторика, графы, математическая логика, конечные автоматы, теория алгоритмов, теория чисел, алгебраические системы. Поскольку дискретная математика обычно читается студентам младших курсов, материал излагается доступно и иллюстрируется многочисленными примерами.

Книга адресована студентам, аспирантам и преподавателям вузов, а также лицам, желающим самостоятельно познакомиться с основными разделами дискретной математики.

УДК 512
ББК 22.176я73

Обложка
П. И. ПОЛЯКОВА

© Издательство «Лань», 2022
© И. А. Мальцев, 2022
© Издательство «Лань»,
художественное оформление, 2022

ВВЕДЕНИЕ

Термин «дискретный» означает *прерывистый*, *дробный* и происходит от латинского слова *discretus*. Противоположное свойство выражается словом «непрерывный». Когда человек идет по улице, каждая точка его тела описывает в пространстве некоторую непрерывную кривую. Если же рассматривать только следы, оставляемые ногами этого человека на снегу, то, по-видимому, все согласятся, что они образуют дискретную совокупность. В этом примере неявно фигурирует и время, ведь отпечатки ног появляются лишь в определенные моменты, а в промежутке между двумя такими моментами происходит нечто, нас не интересующее. Также не имеет значения, когда именно появляется очередной след, важно лишь, что от момента появления предыдущего отпечатка проходит какое-то время.

Сказанное позволяет уловить разницу между изучаемыми дискретной математикой дискретными процессами и непрерывными, описываемыми непрерывной математикой, хотя, конечно, четкой границы между этими двумя разделами нет. Полвека назад практически все электронные устройства были аналоговые, т. е. используемые в них сигналы являлись непрерывными. Типичными примерами могут служить радиоприемники, телевизоры, магнитофоны, различные усилители того времени. При проектировании таких устройств использовались главным образом методы непрерывной математики. Исключениями являлись относительно немногочисленные громоздкие компьютеры.

Теперь трудно найти человека, который не сталкивался бы с устройствами, использующими цифровые технологии и, следовательно, достижения дискретной математики. Достаточно упомянуть ставшие массовыми компьютеры, диски с записями музыки и фильмов, цифровые фотоаппараты и видеокамеры. Цифровыми являются все современные специализированные устройства, используемые в системах управления и связи, например интернет, сотовые телефоны и спутниковое телевидение. В ближайшие годы на цифровое вещание перейдут эфирные радио и телевидение.

Дискретная математика в эти годы также быстро развивалась, и в настоящее время огромное количество специалистов проводит

исследования в различных ее областях. Она служит основой для создания интеллектуальных систем информации, защиты информации, нахождения методов принятия решения на основе анализа неполной и частично противоречивой информации.

В этой книге затрагиваются следующие ее разделы: комбинаторика, теория графов, теория автоматов, теория алгоритмов, математическая логика, теория чисел, алгебраические системы. Каждый из них представляет собой обширную и хорошо разработанную теорию, и читатель найдет здесь лишь начальные понятия и простейшие теоремы, которые позволят ему составить представление о предмете и облегчат чтение специальной литературы.

Основой книги послужили лекции, прочитанные автором в Сибирском государственном университете телекоммуникаций и информатики (СибГУТИ), однако она содержит также материал, в лекции не вошедший. Специфика аудитории сделала необходимым очень подробное и доходчивое изложение материала. Это, однако, не означает, что пришлось прибегать к каким-то упрощениям и жертвовать строгостью определений и доказательств. Напротив, аккуратные формулировки приучают студентов к осторожности и внимательности в обращении с математикой, заставляют следить за своей речью, убирать из высказываний все лишнее и не забывать сказать существенное, говорить так, чтобы быть понятным собеседнику.

Курс дискретной математики читается на разных потоках за один или два семестра. В последнем случае он обычно делится на курс дискретной математики и курс математической логики. Книга построена таким образом, чтобы ею могли пользоваться студенты, обучающиеся по любому из этих вариантов. В семестровый курс входят параграфы 1.1, 1.2, 3.1–3.4, 3.7–3.9, 4.2–4.6, 4.9, 5.4, 5.5, 6.2–6.4. Годовой курс включает в себя почти целиком главы 1–6. Последние главы содержат дополнительный материал, изучаемый факультативно, и в случае необходимости могут служить справочником.

Практика показывает, что многие студенты ограничиваются заучиванием перед экзаменом всех определений и формулировок теорем. Такая работа необходима, однако, чтобы владеть предметом, этого недостаточно. Надо обязательно разбирать примеры, следующие за определениями, и доказательства теорем — иначе можно знать формулировку, но не понимать, что за ней кроется, и тем более не уметь использовать соответствующее понятие или теорему для решения задач. Разбор доказательств приучает правильно рассуждать, мыслить логически, разбивать рассуждение на этапы, отличать доказанное от правдоподобно-

го. Все эти важные качества, совершенно необходимые как исследователю, так и инженеру, не являются врожденными, они приобретаются на основе жизненного опыта и, главное, систематического обучения.

Встречающийся в книге знак □ означает, что доказательство окончено, а при отсутствии доказательства утверждения — что утверждение очевидно. Он также указывает конец примеров, чтобы отделить их от последующего текста. Определения набраны *наклонным* шрифтом, определяемые слова — также наклонным, но несколько ***жирнее***, утверждения — *курсивом*.

Определения и утверждения имеют номера, состоящие из трех чисел. Первое означает номер главы, второе — номер раздела в этой главе, третье — номер определения или утверждения в разделе.

При изложении материала в формулах используются буквы латинского и греческого алфавитов. Последний многим не знаком, поэтому ниже приводится список букв этого алфавита и их названия:

A, α а́льфа	B, β бе́та	Γ, γ га́мма
Δ, δ де́льта	E, ε э́псилон	Z, ζ дзе́та
H, η э́та	Θ, θ тэ́та	I, ι йо́та
K, $\varkappa$ ка́ппа	Λ, λ ля́мбда	M, μ мю
N, ν ню	Ξ, ξ кси	O, o о́микрон
Π, π пи	P, ρ ро	Σ, σ си́гма
T, τ та́у	Υ, υ и́псилон	Φ, φ фи
X, χ хи	Ψ, ψ пси	Ω, ω оме́га

Глава 1

МНОЖЕСТВА

1.1. ОПЕРАЦИИ С МНОЖЕСТВАМИ

Рассуждая о чем-либо, мы всегда имеем в виду некоторую определенную совокупность объектов. Например, мы можем изучать какие-либо свойства натуральных чисел, свойства многочленов с действительными коэффициентами и т. п. В первом случае мы ограничиваемся рассмотрением совокупности натуральных чисел и не рассматриваем другие числа. Это, однако, означает, что какие-то свойства позволяют отличить числа натуральные от прочих. Такие свойства естественно назвать **характеристическими**.

Множеством *мы называем совокупность каких-либо объектов, обладающих общим свойством, характеризующим эти объекты.*

Сказанное не является определением множества, так как понятие множества является первичным и слово «совокупность» ничем не лучше слова «множество». Напомним, что в курсе геометрии, изучаемой в школе, также встречались первичные понятия, не имеющие определения: точки, прямые и плоскости. Считается, что наш опыт позволяет в каждом конкретном случае понять, является ли некоторый объект множеством. Чтобы не повторять слово «множество» несколько раз в одном предложении, употребляют также слова «совокупность», «система», «набор» и т. д., считая их синонимами слова «множество».

1.1.1. *Объекты, образующие множество, называются его* ***элементами****. Запись $u \in B$ означает, что u является элементом множества B (принадлежит B). Чтобы показать, что v не является элементом множества B, пишут $v \notin B$. Через $\{a_1, a_2, \ldots\}$ обозначается множество, состоящее из элементов $a_1,\ a_2,\ \ldots$.*

Задать множество можно различными способами. Если число его элементов небольшое, то можно их перечислить.

Пример. $\{2, 4, 6\}$ — множество с тремя элементами. Множество {лошадь, корова, овца} также образовано тремя элементами. В множестве {Вася} всего один элемент. □

Характеристическое свойство позволяет выделить элементы множества среди прочих потенциальных элементов. Этому свойству удовлетворяют элементы множества, и только они. Иногда достаточно только это свойство и указать.

Пример. Множествами являются совокупность $\mathbb{R}$ всех действительных чисел, совокупность $\mathbb{Z} = \{0, \pm 1, \pm 2, \pm 3, \ldots\}$ всех целых чисел и совокупность $\mathbb{N} = \{1, 2, 3, \ldots\}$ всех целых положительных чисел, называемых также **натуральными**. Свойство «быть действительным числом» является характеристическим в первом случае, «быть целым числом» — во втором, «быть целым положительным числом» — в последнем случае.

Можно говорить о множестве всех прямых на плоскости, проходящих через данную точку, о множестве женатых студентов в университете и т. д. □

Иногда характеристическое свойство записывается внутри фигурных скобок:

$$\{x \in \mathbb{R} \mid x \text{ делится на } 3\}.$$

Знак | здесь заменяет слова «таких, что», и вся формула читается следующим образом: множество элементов x из $\mathbb{R}$ таких, что x делится на 3. Конечно, можно сказать и более гладко: множество действительных чисел, делящихся на 3.

Стоит особо отметить, что каждый элемент множества должен по каким-то признакам отличаться от любого другого элемента того же множества. Так, $\{1, 2, 3\}$ — множество с тремя элементами, а $\{3, 3, 3\}$ трехэлементным множеством не является, поскольку фигурирующие в записи тройки ничем, кроме порядка следования, друг от друга не отличаются.

1.1.2. *Множество, не содержащее ни одного элемента, называется* ***пустым*** *и обозначается символом* $\varnothing$.

На первый взгляд кажется странным рассматривать объект, которого вроде бы и нет, — ведь в пустом множестве нет ни одного элемента. Однако иногда, выделяя то или иное множество, мы еще не знаем, есть ли в этом множестве элементы. При выполнении некоторых операций с множествами также может оказаться, что полученный результат ни одного элемента не содержит. Тем не менее очень удобно считать, что результатом операции всегда является множество.

1.1.3. *Если каждый элемент множества A принадлежит множеству B, то A называется* ***подмножеством*** *множества B. Утверждение «A является подмножеством множества B» записывается формулой $A \subseteq B$, а его отрицание формулой $A \not\subseteq B$. Считается, что пустое множество является подмножеством любого множества.*

Согласно приведенному определению каждое множество является своим подмножеством. Вместо слова «подмножество» иногда употребляют слово «часть».

Пример. Множество $\{1, 2\}$ имеет следующие подмножества:

$$\{1, 2\},\ \{1\},\ \{2\},\ \varnothing\,. \qquad \square$$

1.1.4. *Множества A и B называются* ***равными****, если $A \subseteq B$ и $B \subseteq A$. Утверждение «множества A и B равны» записывается обычным образом: $A = B$. Запись $A \neq B$ означает, что какое-то из множеств A, B содержит хотя бы один элемент, не принадлежащий другому множеству.*

Пример. Из этого определения, в частности, следует, что множества

$$\{a, b, c\}, \quad \{b, c, a\}, \quad \{c, a, b\}$$

являются равными, т. е. что перед нами три различные записи одного и того же множества. $\square$

1.1.5. *Подмножество C множества B называется* ***собственным****, если $C \neq \varnothing$ и $C \neq B$. Чтобы подчеркнуть, что C — собственное подмножество множества B, будем использовать запись $C \subset B$. Два подмножества любого непустого множества называются* ***несобственными****: само множество и пустое подмножество.*

Отметим, что у многих авторов символ $\subset$ используется вместо $\subseteq$ и, встречая этот символ в текстах, надо выяснять точное его значение.

1.1.6. *Пересечением*

$$A_1 \cap A_2 \cap \ldots \cap A_n = \bigcap_{i=1}^{n} A_i$$

множеств $A_1, \ldots, A_n$ называется множество, состоящее из тех и только тех элементов, которые принадлежат каждому множеству A_i, $i = \overline{1, n}$. Если пересечение множеств A и B пусто, то множества A и B называются ***непересекающимися****.*

Пример. Пусть

$$A = \{1, 2, 3, 4, 5\}, \quad B = \{2, 4, 6, 8\}, \quad C = \{8, 9, 10\}.$$

Из определения 1.1.6 следует, что

$$A \cap B = \{2, 4\}, \quad B \cap C = \{8\}, \quad A \cap C = \varnothing. \qquad \square$$

1.1.7. ***Объединением*** *множеств $A_1, \ldots, A_n$ называется множество*

$$A_1 \cup A_2 \cup \ldots \cup A_n = \bigcup_{i=1}^{n} A_i,$$

состоящее из тех и только тех элементов, которые принадлежат хотя бы одному из множеств A_i, $i = \overline{1, n}$.

Пример. Объединением множеств A и B, где

$$A = \{1, -1, 2, -2\}, \quad B = \{0, 1, 2, 3, 4\},$$

является множество

$$A \cup B = \{-2, -1, 0, 1, 2, 3, 4\}. \qquad \square$$

1.1.8. ***Разностью*** *множеств A и B называется множество $A \setminus B$, состоящее из всех элементов, принадлежащих A, но не принадлежащих B.*

Пример. Если

$$M = \{a, b, c\}, \quad N = \{b, c, d\}, \quad P = \{a, b, c, d, e\},$$

то

$$M \setminus N = \{a\}, \quad M \setminus P = \varnothing, \quad P \setminus M = \{d, e\},$$
$$N \setminus M = \{d\}, \quad N \setminus P = \varnothing, \quad P \setminus N = \{a, e\}. \qquad \square$$

1.1.9. ***Симметрической разностью*** *$A \triangle B$ множеств A и B называется множество $(A \setminus B) \cup (B \setminus A)$.*

Пример. Если $A = \{1, 2, 3, 4\}$, $B = \{3, 4, 5\}$, то

$$A \setminus B = \{1, 2\}, \quad B \setminus A = \{5\},$$

$$A \triangle B = (A \setminus B) \cup (B \setminus A) = \{1, 2, 5\}. \qquad \square$$

1.1.10. *Если множество A является подмножеством множества B, то разность $B \setminus A$ называется* ***дополнением множества*** A ***до*** B.

Пример. Множество $M = \{a, b, c\}$ является дополнением множества $K = \{d, e, f\}$ до множества $L = \{a, b, c, d, e, f\}$. □

Часто предполагается, что все элементы рассматриваемых множеств принадлежат некоторому множеству U, называемому **универсальным** или **универсом**.

1.1.11. *Если все рассматриваемые множества являются подмножествами некоторого универсального множества U, то вместо «дополнение множества A до U» говорят «****дополнение множества*** A*». Дополнение множества A обозначается через $\overline{A}$.*

Пример. Если

$$A = \{0, 2, 4, 6, 8\}, \quad B = \{4, 5, 6\},$$

а множество $U = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ является универсом, то

$$\overline{A} = \{1, 3, 5, 7, 9\}, \quad \overline{B} = \{1, 2, 3, 7, 8, 9\}. \qquad \square$$

Весьма наглядно можно проиллюстрировать результат применения к множествам определенных выше операций с помощью **диаграмм Эйлера.** Будем предполагать, что элементами множеств являются точки плоскости. Каждое множество будем изображать некоторым кругом на плоскости, а результат выполнения операций — заштрихованной частью плоскости. Получим рис. 1.1–1.4.

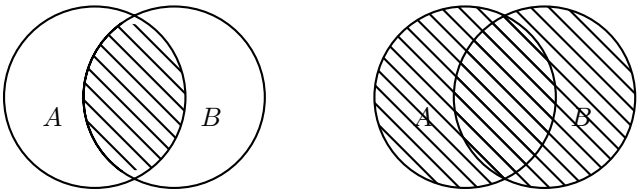


Рис. 1.1
$A \cap B$; $A \cup B$

Рассмотрим следующее равенство:

$$\overline{A \cap B} = \overline{A} \cup \overline{B}. \tag{1.1.1}$$

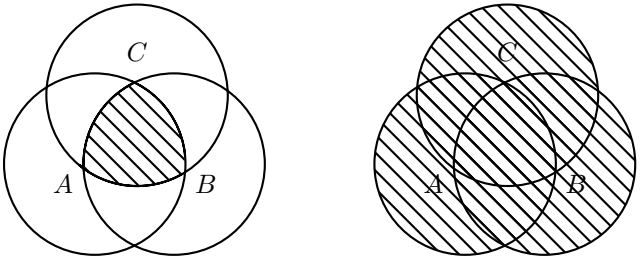


Рис. 1.2
$A \cap B \cap C;\ \ A \cup B \cup C$

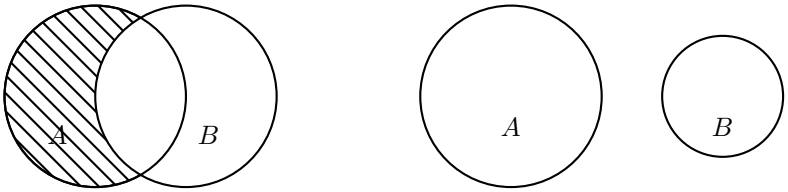


Рис. 1.3
$A \setminus B;\ \ A \cap B = \varnothing$

Как убедиться в его истинности? Нужно доказать, что элемент x принадлежит множеству $\overline{A \cap B}$ тогда и только тогда, когда он принадлежит множеству $\overline{A} \cup \overline{B}$. Можно рассуждать следующим образом.

Если $x \notin A$ и $x \notin B$, то $x \notin A \cap B$ и $x \in \overline{A \cap B}$. Также $x \in \overline{A}$, $x \in \overline{B}$ и $x \in \overline{A} \cup \overline{B}$.

Если $x \notin A$ и $x \in B$, то $x \notin A \cap B$ и $x \in \overline{A \cap B}$. Также $x \in \overline{A}$, $x \notin \overline{B}$ и $x \in \overline{A} \cup \overline{B}$.

Рассуждения продолжаем до тех пор, пока не рассмотрим все случаи.

Эти рассуждения можно формализовать и упростить записи, введя особые функции. Для каждого непустого подмножества M универсального множества U зададим функцию $I_M(x)$, определенную на U,

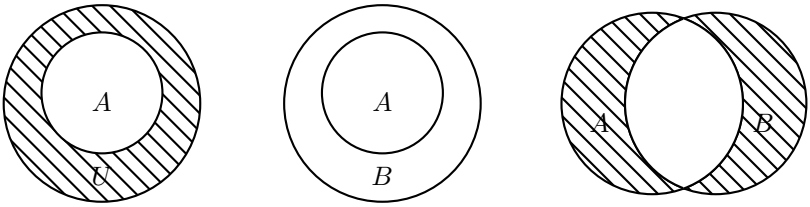


Рис. 1.4
$\overline{A}$; $A \subset B$; $A \triangle B$

следующим образом:

$$I_M(x) = \begin{cases} 1, \text{ если } x \in M\,, \\ 0, \text{ если } x \notin M\,. \end{cases}$$

Это означает, что утверждения $x \in M$ и $I_M(x) = 1$ равнозначны так же, как и утверждения $x \notin M$ и $I_M(x) = 0$. Функция $I_M(x)$ иногда называется **индикатором принадлежности элемента x множеству M**.

Первое из приведенных выше рассуждений теперь можно записать следующим образом:

$$\begin{gathered} \textit{Если } I_A(x) = 0 \textit{ и } I_B(x) = 0\,, \textit{ то} \\ I_{A\cap B}(x) = 0\,,\ I_{\overline{A\cap B}}(x) = 1\,, \\ I_{\overline{A}}(x) = 1\,,\ I_{\overline{B}}(x) = 1\,,\ I_{\overline{A}\cup\overline{B}}(x) = 1\,. \end{gathered} \tag{1.1.2}$$

Значения индикаторов принадлежности элемента x для всех рассматриваемых случаев удобно свести в одну таблицу. Для рассматриваемого соотношения (1.1.1) получаем табл. 1.1, в которой использованы следующие обозначения:

$$A \cap B = C, \quad \overline{A} \cup \overline{B} = D\,.$$

Первая из ее строк, состоящих из нулей и единиц, составлена из значений индикаторов принадлежности, указанных в (1.1.2). Совпадение значений индикаторов принадлежности в четвертом и седьмом столбцах табл. 1.1 означает, что равенство (1.1.1) верное. □

1.1.12. *Множество пар*

$$\{(1, a), (2, b), (3, c), \ldots, (n, u)\}\,,$$

Таблица 1.1

$x \in A$	$x \in B$	$x \in C$	$x \in \overline{C}$	$x \in \overline{A}$	$x \in \overline{B}$	$x \in D$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

в котором a, b, $c, \ldots,$ u — элементы какого-нибудь множества A, не обязательно различные, называется ***конечной последовательностью длины*** *n. Пары $(1, a)$, $(2, b)$ и т. д. называются* ***членами последовательности.***

Название «последовательность» указывает на то, что члены последовательности следуют друг за другом: сначала стоит пара $(1, a)$, затем пара $(2, b)$ и т. д. Иначе говоря, в конечной последовательности всегда есть первая пара, вторая, ..., последняя. Это наводит на мысль о том, что запись последовательности можно сделать более рациональной.

1.1.13. *Пусть $\{(1, a), (2, b), (3, c), \ldots, (n, u)\}$ — некоторая конечная последовательность, в которой a, b, c, ... — элементы множества A. Обозначая пару $(1, a)$ через a_1, пару $(2, b)$ — через a_2 и т. д., эту последовательность можно записать в виде $(a_1, \ldots, a_n)$. Такую запись будем называть* ***n-кой*** *или* ***набором из n элементов множества*** *A. При малых n говорят о* ***двойках (парах)*** *элементов,* ***тройках*** *и т. д.*

Каждая последовательность строится из элементов некоторого множества A. Нижние индексы в записи этой последовательности в виде n-ки просто указывают номер места соответствующего элемента и часто могут быть опущены. Например, если множество A состоит из натуральных чисел, то (2, 4, 6, 8) является (числовой) последовательностью длины 4. Ту же длину имеет последовательность (0, 1, 0, 1). В обоих случаях перед нами четверки чисел.

Поскольку в n-ке существенную роль играет порядок, в котором перечисляются элементы множества, то две n-ки, образованные одними и теми же элементами, расположенными в разном порядке, считаются не совпадающими. Так, при $a \neq b$ двойки (a, b) и (b, a) являются различными.

1.1.14. *Декартовым произведением множеств* $A_1, \ldots, A_n$ *называется множество* $A_1 \times A_2 \times \ldots \times A_n$*, образованное всеми* n*-ками* $(b_1, \ldots, b_n)$*, где* $b_i \in A_i$*,* $i = \overline{1,n}$*. Если один из сомножителей является пустым множеством, то и произведение является пустым множеством. Состоящее из* n *сомножителей декартово произведение, в котором каждый из сомножителей равен множеству* A*, обозначается через* A^n*.*

Пример. Пусть $A = \{a, b, c\}$ и $B = \{1, 2\}$, тогда

$$A \times B = \{(a,1), (a,2), (b,1), (b,2), (c,1), (c,2)\},$$

$$B \times A = \{(1,a), (2,a), (1,b), (2,b), (1,c), (2,c)\},$$

$$B^2 = \{(1,1), (1,2), (2,1), (2,2)\}. \qquad \square$$

1.2. ОТНОШЕНИЯ И ФУНКЦИИ

1.2.1. *Подмножество* R *множества* $A \times B$ *называется* ***бинарным отношением*** *между элементами множеств* A *и* B*. Если* $(u, v) \in R$*, то говорят, что* ***элементы*** u ***и*** v ***находятся в отношении*** R*. Подмножество множества* $A \times A$ *называется* ***бинарным отношением на*** A*.*

Пример. Элементами множества $\mathbb{N}^2 = \mathbb{N} \times \mathbb{N}$ являются всевозможные пары натуральных чисел. Обозначим через R множество, образованное теми парами (x, y) из $\mathbb{N}^2$, у которых $x < 5$ и $y < 3$. Очевидно,

$$R = \{(1,1), (1,2), (2,1), (2,2), (3,1), (3,2), (4,1), (4,2)\}.$$

Согласно 1.2.1 R является отношением на $\mathbb{N}$. $\square$

Вместо «отношение между элементами множеств A и B» можно сказать также «отношение на паре множеств A, B». Указать на принадлежность пары (a, b) отношению R можно несколькими способами:

$$(a,b) \in R, \quad R(a,b), \quad aRb.$$

Последний способ может показаться странным, однако на самом деле им обычно и пользуются: $2 < 7$; $5 = 5$.

Задать бинарное отношение можно различными способами. Можно перечислить все пары, принадлежащие отношению, или указать какое-то свойство, специфичное именно для этих пар. Иногда полезно задать отношение матрицей.

1.2.2. *Пусть* R *— бинарное отношение между элементами множеств* $A = \{\alpha_1, \ldots, \alpha_m\}$*,* $B = \{\beta_1, \ldots, \beta_n\}$*. Матрица* $A = (\gamma_{ij})_m^n$*, имеющая* m

строк и n столбцов, называется ***матрицей отношения*** *R, если ее элементы удовлетворяют следующему условию:*

$$\gamma_{ij} = \begin{cases} 1, & \text{если } (\alpha_i, \beta_j) \in R, \\ 0, & \text{если } (\alpha_i, \beta_j) \notin R. \end{cases}$$

Пример. Пусть $A = 1, 2, 3, 4$, $B = 3, 4, 5$ и

$$R = \{(1,3), (1,4), (2,4), (2,5), (3,3), (3,4), (4,4), (4,5)\}.$$

Матрица этого отношения имеет вид

$$\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}. \qquad \square$$

1.2.3. *Пусть R — отношение на паре множеств A и B.* ***Областью определения*** *бинарного отношения R называется множество*

$$\{x \in A \mid \text{существует такое } y \in B, \text{ что } (x,y) \in R\}.$$

Обозначение: $\operatorname{Dom} R$.

Областью (множеством) значений *отношения R называется множество*

$$\{y \in B \mid \text{существует такое } x \in A, \text{ что } (x,y) \in R\}.$$

Обозначение: $\operatorname{Im} R$.

Пример. Если $R = \{(0,a), (0,b), (2,a), (2,b)\}$ — отношение между элементами множеств $\{0,1,2\}$ и $\{a,b,c\}$, то

$$\operatorname{Dom} R = \{0,2\}, \quad \operatorname{Im} R = \{a,b\}. \qquad \square$$

1.2.4. *Множество $\overline{R} = (A \times B) \setminus R$ называется* ***дополнением бинарного отношения*** *$R \subseteq A \times B$.*

Пример. Пусть

$$A = \{0,1,2\}, \quad B = \{a,b\}, \quad R = \{(0,a), (0,b), (2,a)(2,b)\},$$

тогда $\overline{R} = \{(1,a), (1,b)\}$. $\square$

1.2.5. ***Произведением*** *(или* ***композицией*****) отношений** *$R_1 \subseteq A \times B$ и $R_2 \subseteq B \times C$ называется отношение $R_1 \circ R_2 \subseteq A \times C$, образованное следующим образом: пара $(x,y) \in A \times C$ принадлежит $R_1 \circ R_2$ тогда и только тогда, когда в B существует такой элемент z, для которого $(x,z) \in R_1$ и $(z,y) \in R_2$.*

Пример. Возьмем множества

$$A = \{a, b, c\}, \quad B = \{1, 2, 3, 4\}, \quad C = \{f, g, h\}.$$

Множества пар

$$R_1 = \{(a, 1), (a, 2), (b, 2), (c, 3)\},$$
$$R_2 = \{(2, f), (2, g), (4, f), (4, g), (4, h)\}$$

являются отношениями между элементами множеств A, B и B, C соответственно. Из определения 1.2.5 следует, что

$$R_1 \circ R_2 = \{(a, f), (a, g), (b, f), (b, g)\}. \qquad \square$$

1.2.6. *Обратным отношением для бинарного отношения R называется множество R^{-1}, состоящее из тех пар (a, b), для которых $(b, a) \in R$.*

Пример. Если $R = \{(a, 1), (a, 2), (b, 2), (c, 3)\}$, то

$$R^{-1} = \{(1, a), (2, a), (2, b), (3, c)\}. \qquad \square$$

Некоторые виды отношений встречаются особенно часто и потому заслуживают упоминания.

1.2.7. *Бинарное отношение R на множестве A называется*

а) ***рефлексивным*** *(или* ***иррефлексивным****), если $(x, x) \in R$ для каждого $x \in A$;*

б) ***антирефлексивным****, если $(x, x) \notin R$ для каждого $x \in A$;*

в) ***симметричным****, если для любых $x, y \in A$ из $(x, y) \in R$ следует, что и $(y, x) \in R$;*

г) ***антисимметричным****, если для любых $x, y \in A$ из $(x, y) \in R$ и $x \neq y$ следует, что $(y, x) \notin R$;*

д) ***транзитивным****, если для любых $x, y, z \in A$ если $(x, y) \in R$ и $(y, z) \in R$, то и $(x, z) \in R$.*

Примеры. 1. Пусть множество $M = \{в, к, м\}$ состоит из волка, кошки и мышки. Обозначим через R_1 множество тех пар зверей (i, j), в которых зверь i может съесть зверя j, а через R_2 — множество тех пар (p, q), в которых p не может съесть q. Очевидно,

$$R_1 = \{(в,к),(в,м),(к,м)\},$$

$$R_2 = \{(в,в),(к,к),(м,м),(к,в),(м,в),(м,к)\}.$$

Отношение R_1 антирефлексивное, так как не содержит пар $(в,в)$, $(к,к)$, $(м,м)$, антисимметричное, поскольку в нем нет пар $(к,в)$, $(м,в)$, $(м,к)$,

транзитивное. Отношение R_2 рефлексивное, антисимметричное, транзитивное.

2. Зададим на множестве всех жителей нашего города отношение ϱ следующим образом: $(a, b) \in \varrho$ тогда и только тогда, когда человек по имени a является сыном человека по имени b. Нетрудно убедиться в том, что это отношение антирефлексивное, антисимметричное, не транзитивное. □

Заметим, что отношение R на множестве A не является рефлексивным, если хотя бы для одного элемента x из A пара (x, x) не принадлежит R. Сходным образом определяются отношения, не симметричные и не транзитивные. Отношение R на множестве A не является симметричным, если в A найдутся такие элементы x и y, что $(x, y) \in R$ и $(y, x) \notin R$. Отношение R не является транзитивным, если в A имеются такие элементы x, y и z, что

$$(x, y) \in R, \quad (y, z) \in R, \quad (x, z) \notin R\,.$$

Замечания. Одного нарушения необходимого условия достаточно для того, чтобы отношение не обладало каким-либо из свойств, отмеченных в 1.2.7. Так, отношение

$$\{(1, 1), (1, 2), (2, 2), (2, 3), (1, 3)\}$$

на множестве $\{1, 2, 3\}$ не рефлексивно, потому что не содержит пары $(3, 3)$.

При проверке транзитивности отношения R для каждой пары $(x, y) \in R$ мы ищем пару $(y, z) \in R$ и смотрим, есть ли в R пара (x, z). Может оказаться, что для каждой пары (x, y) в R пары вида (y, z) нет. В этом случае отношение также считается транзитивным. Так, транзитивным является отношение $\{(1, 2), (1, 3), (1, 4)\}$ на множестве $\{1, 2, 3, 4\}$.

По свойствам матрицы бинарного отношения можно судить о некоторых свойствах самого отношения.

1.2.8. *Пусть $M = (\alpha_{ij})_n^n$ — матрица бинарного отношения на некотором множестве. Это отношение является*

1) симметричным, если матрица M симметричная: $\alpha_{ij} = \alpha_{ji}$ для любых i и j;

2) антисимметричным, если матрица M антисимметричная: из $\alpha_{ij} = \alpha_{ji}$ следует $i = j$;

3) рефлексивным, если на главной диагонали матрицы M нет нулей;

4) антирефлексивным, если на главной диагонали матрицы M нет единиц. □

1.2.9. *Бинарное отношение R на множестве A называется* ***диагональю****, если оно состоит из всех пар вида (x, x), где $x \in A$. Диагональ на множестве A обозначается через* id_A.

Пример. Если $A = \{0, 1, 2\}$, то $\mathrm{id}_A = \{(0, 0), (1, 1), (2, 2)\}$. □

Нетрудно заметить, что бинарное отношение на множестве A является рефлексивным тогда и только тогда, когда оно содержит все пары из id_A, антирефлексивным тогда и только тогда, когда оно не содержит ни одной пары из id_A.

1.2.10. *Бинарное отношение R на множестве A называется* ***отношением эквивалентности****, или* ***эквивалентностью****, если оно рефлексивно, симметрично и транзитивно.*

Пример. Свойство «быть ровесником» можно считать отношением эквивалентности на множестве всех студентов, обучающихся в университете. □

1.2.11. *Рефлексивное транзитивное антисимметричное бинарное отношение называется* ***отношением частичного порядка (частичным порядком, частичным упорядочением)****. Множество, на котором задано отношение частичного порядка, называется* ***частично упорядоченным****.*

Обычно для отношения частичного порядка используется знак $\leq$ и вместо $(a, b) \in \leq$ пишут $a \leq b$.

Пример. Пусть M — множество всех пар натуральных чисел. Будем писать $(a_1, b_1)R(a_2, b_2)$, если $a_1 \leq a_2$ и $b_1 \leq b_2$. Легко убедиться в том, что R является отношением частичного порядка на M. □

1.2.12. *Транзитивное антисимметричное бинарное отношение называется* ***отношением строгого частичного порядка (строгим частичным порядком, строгим частичным упорядочением)****.*

Примеры. 1. Множество M состоит из четырех точек, обозначенных буквами a, b, c, d (рис. 1.5). Рассмотрим отношение

$$R = \{(a, b), (a, c), (a, d), (b, d), (c, d)\}.$$

Пара точек принадлежит R тогда и только тогда, когда на рис. 1.5 первая из них находится выше второй. Нетрудно проверить, что R — отношение строгого частичного порядка.

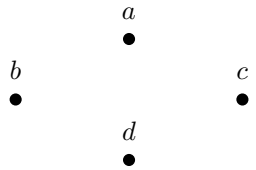


Рис. 1.5
Множество M

2. Если M — множество всех студентов, обучающихся в университете, то M^2 — множество всевозможных пар студентов. Отметим, что пары (a, b) и (b, a), образованные студентами a и b, считаются различными и обе входят в M^2. По определению это множество также содержит все пары вида (a, a). Составим отношение R из тех и только тех пар (a, b), в которых студент a в данный момент старше студента b. Очевидно, R является отношением строгого частичного порядка. □

В курсе элементарной математики обычным является следующее определение функции.

Говорят, что задана ***функция, отображающая множество*** A ***в множество*** B*, если указано правило, сопоставляющее каждому элементу множества* A *один и только один элемент множества* B*.*

Обозначим функцию, удовлетворяющую этому определению, через $f(x)$, и рассмотрим множество R всех пар вида $(x, f(x))$. Очевидно, это множество является подмножеством множества $A \times B$, т. е. бинарным отношением между элементами множеств A и B. Таким образом, каждая имеющая один аргумент функция может рассматриваться как некоторое особое бинарное отношение. Такое отношение не должно содержать пар (a, b) и (a, c), если $b \neq c$, так как иначе окажется, что

$$f(a) = b, \quad f(a) = c, \quad b \neq c.$$

Кроме того, для каждого u из области определения функции f должно найтись такое v, что пара (u, v) принадлежит отношению R, поскольку значение $f(u)$ существует.

Возникает следующее определение.

1.2.13. *Отношение* R *между элементами множеств* A *и* B*, обладающее следующими двумя свойствами:*

а) *если* $(x, y_1) \in R$ *и* $(x, y_2) \in R$*, то* $y_1 = y_2$*,*

б) *для каждого* $x \in A$ *существует такой элемент* $y \in B$*, что* $(x, y) \in R$*,*

называется ***отображением множества*** A ***в множество*** B*, или* ***функцией из*** A ***в*** B***.***

Отображение f *множества* A *в множество* B *будет обозначаться через* $f : A \to B$*. Вместо* $(x, y) \in f$ *будем писать* $f(x) = y$*.*

Пример. Отношение

$$\{(0,2),(3,5),(1,2),(2,5)\}$$

между элементами множеств

$$A = \{0,1,2,3\}\,, \quad B = \{0,1,2,3,4,5\}$$

является отображением множества A в множество B.

Отношение

$$\{(0,2),(0,5),(3,5),(1,2),(1,5),(2,5)\}$$

между элементами тех же множеств функцией не является хотя бы потому, что содержит пары $(0,2)$ и $(0,5)$. □

1.2.14. *Отображение* $f : A \to B$ *называется* ***отображением на*** *(все множество)* B*, или* ***сюръекцией****, если для каждого* $y \in B$ *существует такой элемент* $x \in A$*, для которого* $f(x) = y$*.*

Пример. 1. Функция $f(x)$, определенная равенствами

$$f(0) = f(1) = 2\,, \quad f(2) = f(3) = 5\,,$$

отображает множество $M = \{0,1,2,3\}$ на множество $\{2,5\}$, т. е. является сюръекцией.

2. Функция $f : M \to M$, определенная теми же равенствами, сюръекцией не является, так как, например, $f(x) \neq 0$ ни для какого $x \in M$.

3. После звонка 28 студентов вошли в аудиторию и уселись за 18 столов. Обозначим множество студентов через A, а множество столов — через B. Если за один стол могут сесть не более двух студентов, то получаем сюръекцию $h : A \to B$, так как за каждый стол сел хотя бы один студент. □

1.2.15. *Отображение* $f : A \to B$ *называется* ***взаимно однозначным****, или* ***инъекцией****, или* $1-1$***-отображением****, если для любых* $x_1, x_2 \in A$ *из* $x_1 \neq x_2$ *следует, что* $f(x_1) \neq f(x_2)$*.*

Латинское слово *injectio* означает *вбрасывание*.

1.2.16. *Отображение $f : A \to B$ называется* ***взаимно однозначным отображением множества A на множество B****, или* ***биекцией*** *A на B, если оно является одновременно инъекцией и сюръекцией.*

Часть «би» слова «биекция» происходит от латинского слова *bis* — дважды. Чтобы стать биекцией, отображение должно обладать двумя свойствами — быть инъекцией и сюръекцией.

Пример. 1. Пусть

$$A = \{0, 1, 2\}, \quad B = \{0, 2, 4\}, \quad C = \{0, 1, 2, 3, 4\}.$$

Функция $f(x) = 2x$ взаимно однозначно отображает A на B, т. е. является биекцией. Функция $g : A \to C$, также задаваемая формулой $g(x) = 2x$, взаимно однозначно отображает A в C, т. е. является инъекцией.

2. Для проведения контрольной работы преподаватель приготовил 28 листков с задачами, среди которых не было одинаковых. После звонка он раздал задания двадцати восьми студентам. Здесь мы имеем дело с биекцией $h : A \to B$, где A — множество листков с задачами, а B — множество студентов, поскольку каждый студент получил задание и все эти задания разные.

3. Обозначим через A множество билетов на спектакль, через B — множество проданных билетов, через C — множество кресел в зале. Очевидно, каждому билету соответствует свое кресло, поэтому возникает отображение $f : A \to C$, являющееся биекцией. Функция $g : B \to C$ также биекция, если проданы все билеты. В противном случае она инъективна, но не сюръективна. □

1.2.17. *Подмножество множества A^n называется* ***n-арным (n-местным) отношением на множестве A****. Одноместные отношения называются* ***унарными****, двуместные —* ***бинарными****, трехместные —* ***тернарными****.*

Пример. Из этого определения следует, что подмножества любого непустого множества являются унарными отношениями. Так, совокупность $\{1, 2, 4\}$ — унарное отношение на множестве $E_6 = \{0, 1, 2, 3, 4, 5\}$, поскольку является его подмножеством.

Множество $\{(0, 0, 0), (1, 2, 3), (3, 4, 5)\}$ служит примером тернарного отношения на том же множестве E_6. □

1.2.18. *Отображение $f : A^n \to B$ называется* ***n-арной (или n-местной) функцией из A в B****. Подобно отношениям, одноместные функции называются* ***унарными****, двуместные —* ***бинарными****, трехместные —* ***тернарными****.*

Пример. Отношение $\{((0,0),0),((0,1),1),((1,0),1),((1,1),2)\}$ задает согласно определениям 1.2.13 и 1.2.14 отображение множества

$$A = \{(0,0),(0,1),(1,0),(1,1)\}$$

на множество $B = \{0,1,2\}$. Вследствие 1.2.18 это отображение можно также назвать **унарной функцией**. Обозначая эту функцию через f, запишем ее в более привычном виде: $f(x)$. Значениями переменной x являются элементы множества A, т. е. пары чисел.

Легко заметить, что $A = C^2$, где $C = \{0,1\}$. Превратим отношение (1.2.) в тернарное, отбросив часть скобок:

$$\{(0,0,0),(0,1,1),(1,0,1),(1,1,2)\}.$$

Оно задает бинарную функцию g, отображающую множество C на множество B. Эта функция может быть записана в следующем виде:

$$g(x,y) = x + y\,.$$

Теперь значениями переменных x и y служат числа 0 и 1. □

1.2.19. *Пусть $f : A \to B$ — унарная функция, $x \in A$. Если $f(x) = y$, то y называется **образом элемента** x, а x — **прообразом элемента** y **при отображении** f. Для любого непустого подмножества C множества A его **образом относительно** f называется множество $f(C) = \{f(x) \mid x \in C\}$.*

Из определения 1.2.19 следует, что при отображении $f : A \to B$ у каждого элемента $x \in A$ имеется только один образ. Если f не инъективно, то элемент $f(x) \in B$ может иметь несколько прообразов.

Пример. Пусть функция $f : \mathbb{N} \to \mathbb{N}$ определена следующим образом: $f(x) = 2x$. Так как $f(3) = 6$, то при этом отображении 6 является образом числа 3, а 3 — прообразом числа 6. Образом множества $\{3,5,7\}$ относительно f является множество $\{6,10,14\}$. □

1.3. МОЩНОСТЬ МНОЖЕСТВА

Вопрос о том, в каком из двух конечных множеств элементов больше, обычно не вызывает затруднений. Каждый согласится с тем, что семья, в которой десять детей, больше семьи с тремя детьми. Однако сумеете ли вы сказать, где больше элементов: в множестве $\mathbb{N}$ натуральных чисел или в множестве $\mathbb{N} \times \mathbb{N}$ пар натуральных чисел? Чтобы найти ответ, надо понять, как можно сравнивать количество элементов в множествах.

Предположим, что перед нами находятся две кучки каких-то предметов, например кучка орехов и кучка камешков. Может ли некий человек установить, что число орехов совпадает с числом камешков, если он не умеет считать? Оказывается, это совсем несложно. Надо отложить в сторону один орех и один камешек, затем еще один орех и один камешек и т. д. Если для последнего ореха найдется парный ему камешек и при этом в кучке камешков ничего не останется, то орехов было столько же, сколько камешков.

Составляя пары орех–камешек, мы строили инъективное отображение множества всех орехов в множество камешков. Число орехов равно числу камешков тогда и только тогда, когда такое отображение существует и является сюръекцией.

Подобным же способом математики сравнивают между собой бесконечные множества.

1.3.1. *Два множества называются* ***равномощными*** *(или* ***эквивалентными****), если одно из них можно взаимно однозначно отобразить на другое.*

Сказанное можно сформулировать и несколько иначе: *множества* A *и* B ***равномощны****, если существует биекция*

$$f : A \to B\,.$$

То общее, что присуще всем равномощным множествам, называется **мощностью множества**. Мощность конечного множества — это число его элементов.

1.3.2. *Множество, равномощное множеству* $\mathbb{N}$ *натуральных чисел, называется* ***счетным****.*

Название прямо указывает на то, что элементы такого множества можно «пересчитать», т. е. сопоставить числам 1, 2 и т. д. так, чтобы получилось взаимно однозначное отображение.

Теперь можно ответить на вопрос, поставленный в начале параграфа.

1.3.3. *Множество всех пар натуральных чисел счетно.*

Доказательство. Надо отобразить множество натуральных чисел на множество пар таких чисел. Достаточно расположить все пары натуральных чисел таким образом, чтобы были первая пара, вторая пара, третья пара и т. д. Рассмотрим такую последовательность пар:

$$(1,1),\ (2,1),\ (1,2),\ (3,1),\ (2,2),\ (1,3),\ (4,1),\ldots\,. \qquad (1.3.1)$$

Она построена по следующим правилам.

1. Пара с меньшей суммой чисел стоит левее пары с большей суммой чисел.

2. Если суммы чисел каждой из двух пар равны, то левее стоит пара, у которой первое число больше.

Эти свойства легче проследить, если пары расположить следующим образом:

$$(1,1),$$
$$(2,1),\ (1,2),$$
$$(3,1),\ (2,2),\ (1,3),$$
$$\ldots\ldots\ldots\ldots\ldots$$

Нетрудно заметить, что последовательность (1.3.1) содержит все пары натуральных чисел. □

1.3.4. *Каждое бесконечное множество содержит счетное подмножество.*

Доказательство. Если множество A бесконечно, то оно непусто и в нем есть элемент, который мы обозначим через a_1. Очевидно, множество $A \setminus \{a_1\}$ также бесконечно, следовательно, непусто, и в нем есть элемент, который мы обозначим через a_2. Продолжая этот процесс, получим счетное подмножество $\{a_1, a_2, a_3, \ldots\}$ множества A. □

1.3.5. *Если множество A равномощно какому-то собственному подмножеству множества B, а множество B не равномощно никакому подмножеству множества A, то говорят, что* ***мощность множества*** A ***меньше мощности множества*** B.

1.3.6. *Множество $\mathbb{Q}$ всех рациональных чисел счетно.*

Доказательство. Каждое рациональное число представимо в виде дроби $\frac{m}{n}$, где m, n — целые числа и $n \neq 0$. Это означает, что все натуральные числа являются рациональными, и потому множество $\mathbb{Q}$, по крайней мере, счетно. Каждая пара натуральных чисел (m, n) соответствует рациональному числу $\frac{m}{n}$. Ранее все такие пары были расположены в последовательность (см. 1.3.3), поэтому все положительные рациональные числа также можно расположить в последовательность

$$\alpha_1,\ \alpha_2,\ \alpha_3,\ \ldots\ .$$

Но тогда и все рациональные числа, как положительные, так и неположительные, можно расположить в последовательность

$$0,\ \alpha_1,\ -\alpha_1,\ \alpha_2,\ -\alpha_2,\ \ldots\ .$$ □

1.3.7. *Если множество A непустое, а множество B содержит более одного элемента, то мощность множества всех различных унарных функций, отображающих A в B, больше мощности множества A.*

Доказательство. Обозначим множество функций, о которых говорится в теореме, через F.

1. Пусть b_1 и b_2 — два фиксированных элемента из множества B. Сопоставим каждому элементу $u \in A$ функцию $f_u(x) \in F$ следующим образом:

$$f_u(x) = \begin{cases} b_1, \text{ если } x = u\,, \\ b_2, \text{ если } x \neq u\,. \end{cases}$$

Получим инъекцию множества A в множество F.

2. Докажем, что инъекции $F \to A$ не существует. Предположим, что это неверно и инъекция $\psi : F \to A$ нашлась. Область определения отображения ψ состоит из унарных функций, отображающих A в B, и должна содержать все такие функции. Покажем, что хотя бы одна из функций, отображающих A в B, в эту область не попала.

Обозначим искомую функцию через g. Чтобы ее задать, надо для каждого $x \in A$ указать элемент $g(x) \in B$. Сделаем это так. Прообразом элемента $x \in A$ при отображении ψ является некоторая функция из F, которую мы обозначим через h_x. Поскольку h_x отображает A в B, элемент $h_x(x)$ принадлежит B. По условию в множестве B найдется элемент y_x, отличный от $h_x(x)$. Положим $g(x) = y_x$.

Построенная функция g отображает A в B. Она не принадлежит области определения отображения ψ. Действительно, пусть это не так и g совпадает с какой-то функцией p из этой области. Если $\psi(p) = z \in \in A$ и $p(z) = d$, то, используя введенное выше обозначение, получаем равенство $p = h_d$. Но тогда $g(d) = y_d \neq h_d(d) = p(d)$ и g не может совпадать с p. □

1.3.8. *Мощность множества всех подмножеств непустого множества A больше мощности множества A.*

Доказательство. Сопоставим каждому подмножеству B множества A следующую функцию:

$$f_B(x) = \begin{cases} 1, \text{ если } x \in B\,, \\ 0, \text{ если } x \notin B\,. \end{cases}$$

Очевидно, полученное соответствие между множеством $U(A)$ всех подмножеств множества A и множеством F всех функций, отображающих

A в множество $\{0,1\}$, является взаимно однозначным. Согласно 1.3.1 мощности множеств $U(A)$ и F совпадают. Из 1.3.7 следует, что мощность множества F больше мощности множества A. □

1.3.9. *Мощность множества $\mathbb{R}$ действительных чисел больше мощности множества $\mathbb{N}$ натуральных чисел.*

Доказательство. Достаточно показать, что мощность какого-то подмножества множества действительных чисел больше мощности множества $\mathbb{N}$. Рассмотрим числа $u \in \mathbb{R}$, представимые в виде бесконечной дроби:

$$u = 0, x_1 x_2 x_3 \ldots, \quad \text{где} \quad x_i \in \{0,1\}, \quad i = 1,2,3,\ldots. \tag{1.3.2}$$

Множество всех таких чисел обозначим через M. Числа дроби (1.3.2), стоящие после запятой, можно рассматривать как значения функции f_u, отображающей множество $\mathbb{N}$ в множество $\{0,1\}$:

$$u = 0, f_u(1) f_u(2) f_u(3) \ldots .$$

Таким образом, мощность множества M равна мощности множества всех функций, отображающих $\mathbb{N}$ в $\{0,1\}$, и потому больше мощности множества $\mathbb{N}$ (см. 1.3.7). □

1.3.10. *Мощность множества всех действительных чисел называется* ***мощностью континуума****.*

Латинское слово *continuum* означает *непрерывное, сплошное.*

Глава 2

КОМБИНАТОРИКА

2.1. ВЫБОРКИ

При решении многих математических проблем теоретического или прикладного характера возникает задача выбора и расположения каких-либо объектов в соответствии с заданными правилами и отыскание числа различных вариантов решения этой задачи. Раздел математики, в котором подобные задачи рассматриваются, называется **комбинаторикой**.

2.1.1. *Число элементов конечного множества A обозначается через $|A|$. Множество, состоящее из n элементов, называется **n-множеством**.*

Следующее утверждение кажется вполне очевидным.

2.1.2. *Если множества A и B не пересекаются и содержат m и n элементов соответственно, то множество $A \cup B$ имеет $m + n$ элементов.*

2.1.3. *Если множества A и B содержат m и n элементов соответственно, то множество $A \times B$ состоит из mn элементов.*

Доказательство. Пусть $C = A \times B$ и $A = \{a_1, \ldots, a_m\}$. Обозначим через C_i множество, образованное всеми парами из C, у которых на первом месте стоит a_i. Очевидно, для любого $i = \overline{1, m}$ множество C_i содержит столько элементов, сколько существует различных пар вида (a_i, z), $z \in B$. Отсюда следует, что число элементов множества C_i равно числу элементов множества B, т. е. n.

Так как в парах, принадлежащих множеству C_i, на первом месте находится a_i, а в парах, принадлежащих множеству C_j, на первом месте находится a_j, то $C_i \cap C_j = \varnothing$, если $i \neq j$. Множество C является объединением множеств $C_1, \ldots, C_m$. Согласно 2.1.2 число элементов

множества C равно сумме чисел элементов множеств C_i, $i = \overline{1,m}$. Видим, что $|C| = |C_1| + \ldots + |C_m| = mn$. □

Представим себе, что у продавца имеется 50 лотерейных билетов и вы хотите купить один из них. Обычно продавец предлагает вам выбрать билет. Очевидно, имеется 50 различных вариантов выбора. Подобные соображения приводят нас к следующему правилу: *один элемент из n-множества можно выбрать n способами.*

Предположим теперь, что у продавца имеется две пачки билетов двух различных лотерей, в одной из них содержится 30, а во второй — 20 билетов. Если мы хотим купить один билет и нам все равно, из какой он будет пачки, то мы можем взять билет из первой пачки (имеется 30 вариантов) либо отказаться от выбора в первой пачке и взять билет во второй пачке (20 вариантов). Общее число различных вариантов выбора равно $30 + 20 = 50$. В этих подсчетах мы использовали следующее правило.

2.1.4. ***Правило суммы.*** *Если объект α можно выбрать m способами, а объект β, отличный от α, — n способами, причем α и β нельзя выбрать одновременно, то осуществить выбор «либо α, либо β» можно $m + n$ способами.*

Пусть некто хочет купить в магазине фотоаппарат и для этого фотоаппарата чехол. В продаже имеется четыре модели фотоаппаратов и три вида чехлов, пригодных для любой модели фотоаппаратов. Сколькими способами можно выбрать фотоаппарат и чехол к нему?

Очевидно, имеется четыре варианта выбора фотоаппарата. Выбрав фотоаппарат, можно тремя способами выбрать чехол. Всего имеется 12 способов выбора.

2.1.5. ***Правило произведения.*** *Если объект α можно выбрать m способами, а после каждого такого выбора можно выбрать n способами объект β, то выбор обоих объектов α и β в указанном порядке можно осуществить mn способами.*

Сравнивая правила 2.1.4 и 2.1.5 с утверждениями 2.1.2 и 2.1.3 соответственно, нетрудно заметить, что в них речь идет об одних и тех же закономерностях, только формулировки используют разные термины.

2.1.6. *Совокупность $(a_1, a_2, \ldots, a_r)$ элементов (не обязательно различных) некоторого n-множества называется* ***выборкой объема r из n элементов****, или* ***(n, r)-выборкой****, или* ***r-выборкой****.*

Пример. Тройка $(1,4,1)$ и четверка $(5,3,4,1)$ являются соответственно $(5,3)$-выборкой и $(5,4)$-выборкой из 5-множества $\{1,2,3,4,5\}$. □

2.1.7. *Выборка называется* ***упорядоченной****, если в ней задан порядок следования элементов, т. е. известно, какой элемент является первым, какой — вторым и т. д. Если такой порядок не определен, то выборка называется* ***неупорядоченной****.*

Пример. Рассмотрим множество $\{a,b,c,d\}$. Если выборка (a,c) из этого множества упорядоченная, то она не совпадает с выборкой (c,a), так как первые элементы в них различны. Если же считать обе выборки неупорядоченными, то они совпадают, так как содержат одни и те же элементы. Неупорядоченная выборка, в которой все элементы различны, является подмножеством множества, из которого взяты элементы. □

2.1.8. *Если элементы упорядоченной (n,r)-выборки попарно различны, то она называется* (n,r)***-перестановкой****, если же среди ее элементов имеются равные, то она называется* (n,r)***-перестановкой с повторениями****. Иногда (n,r)-перестановки называют* ***размещениями из*** n ***элементов по*** r*.*

Из этого определения следует, что две (n,r)-перестановки, составленные из одних и тех же элементов, но различающиеся порядком этих элементов, считаются разными.

Пример. Выборка (a,b,a) является $(5,3)$-перестановкой с повторениями из 5-множества $\{a,b,c,d,e\}$.

Выборки (a,b,e,d) и (b,e,a,d) представляют собой различные $(5,4)$-перестановки из того же множества. □

2.1.9. *Неупорядоченная (n,r)-выборка, все элементы которой попарно различны, называется* (n,r)***-сочетанием (сочетанием из*** n ***элементов по*** r***)****. Если же среди элементов такой выборки имеются равные, то она называется* (n,r)***-сочетанием с повторениями****.*

Таким образом, (n,r)-сочетаниями являются неупорядоченные r-элементные подмножества заданного n-множества. Два таких подмножества считаются разными тогда и только тогда, когда одно из них содержит элемент, отсутствующий во втором.

Пример. В множестве $\{a,b,c,d,e\}$, состоящем из пяти элементов, произведены три выборки объема 4:

$$\{a,c,e,d\}, \quad \{a,e,d,c\}, \quad \{a,b,e,d\}.$$

Выборки неупорядоченные и потому являются $(5,4)$-сочетаниями. Первые два сочетания совпадают, а третье отлично и от первого, и от второго. □

Следующие примеры могут помочь лучше запомнить термины, введенные в определениях 2.1.6–2.1.9. Имеется ящик с n кубиками одного размера. Все грани каждого кубика окрашены в один и тот же цвет, и среди кубиков нельзя найти двух одинаково окрашенных.

Примеры. 1. Возьмем коробку и переложим в нее r кубиков. Получим **выборку**. Эта выборка является **неупорядоченной**, так как мы не отмечали, в каком порядке вынимали кубики из ящика. Такая выборка является **сочетанием**. Поскольку в ящике все кубики были разные, то и в коробке нет одинаковых кубиков. Это означает, что мы имеем **сочетание без повторений**.

2. Вынимая кубики из ящика, будем выстраивать их в один ряд: первый кубик, второй кубик и т. д. Получим **упорядоченную** выборку, т. е. **перестановку**. Поменяв в ней местами любые кубики, получим другую перестановку, хотя составлена она будет из тех же кубиков.

3. **Выборки с повторениями** можно получить следующим образом. Последовательно вынимая кубики из ящика, будем записывать цвет каждого кубика, а кубик возвращать в ящик. Очевидно, теперь мы можем один и тот же кубик вынуть несколько раз. Если нас не интересует, в каком порядке вынимались кубики, то полученную выборку считаем **сочетанием с повторениями**. Если же учитывать порядок появления кубиков, то выборка является **перестановкой с повторениями**. □

2.1.10. *Число $P(n,r)$ различных (n,r)-перестановок равно*

$$n(n-1)\dots(n-r+1)\,.$$

Доказательство. Пусть мы рассматриваем всевозможные (n,r)-перестановки элементов n-множества A. Первым членом (n,r)-перестановки может быть любой элемент из A, следовательно, его можно выбрать n способами. Как только он фиксирован, второй член уже выбирается из $(n-1)$-множества, так как в перестановке элементы множества A повторяться не могут. Согласно правилу произведения 2.1.5 первые два члена можно выбрать $n(n-1)$ способами.

Теперь под объектом α мы понимаем первые два члена перестановки, их можно выбрать $n(n-1)$ способами, а третий член — $n-2$ способами, так как он не должен совпадать ни с первым, ни со вторым. Видим, что первые три члена перестановки можно выбрать $n(n-1)(n-2)$ способами и т. д. Для выбора всех элементов (n,r)-перестановки имеется $n(n-1)\dots(n-r+1)$ различных способов. □

Отметим один важный случай. Число (n,n)-перестановок, называемых обычно перестановками из n элементов, равно $n!$, т. е. произведению $1{\cdot}2{\cdot}3{\cdot}\ldots{\cdot}n$. Для удобства рассуждений полагают $0!=1$. Очевидно, для любого целого $n\geqslant 0$ справедлива формула $n!(n+1)=(n+1)!$.

Примеры. 1. Студентки одной из групп решили подарить к 23 февраля каждому из трех студентов, обучающихся в той же группе, по галстуку. В магазине им показали 10 галстуков с различными рисунками. Сколькими способами можно сформировать подарки так, чтобы у всех трех студентов рисунки на галстуках были разные?

Искомое число совпадает с числом $(10,3)$-перестановок и равно $10\cdot 9\cdot 8=720$.

2. Молодожены пригласили в гости две супружеские пары. Сколькими способами они могут рассадить гостей за столом и разместиться сами? Предполагается, что места за столом пронумерованы числами от единицы до шести и что супруги не обязательно должны сидеть рядом.

Легко подсчитать, что число способов разместиться за столом равно $6!$. □

2.1.11. *Число всевозможных (n,r)-перестановок с повторениями равно n^r.*

Доказательство. Первым членом (n,r)-перестановки может быть любой элемент заданного n-множества, и для его выбора имеется n возможностей. Вторым членом тоже может быть любой элемент того же n-множества, и его можно выбрать также n способами. То же самое можно сказать и об остальных членах (n,r)-перестановки. Применяя правило произведения 1.3.5, видим, что для выбора первых двух членов (n,r)-перестановки имеется n^2 способов, первых трех — n^3 способов и т. д. □

Пример. Студенты одной из групп решили подарить к 8 Марта каждой из 15 студенток той же группы по плитке шоколада. В продаже имеется 6 видов шоколадок. Сколькими способами ими можно одарить студенток?

Число способов совпадает с числом $(6,15)$-перестановок с повторениями и равно 6^{15}, так как первой студентке можно подарить любую из шести видов шоколадок, второй также любую из шести видов и т. д. □

2.1.12. *Число различных подмножеств n-множества A равно 2^n.*

Доказательство. Пусть $A=\{a_1,\ldots,a_n\}$. Каждому подмножеству B множества A сопоставим $(2,n)$-перестановку с повторениями, состоящую из чисел 0 и 1. Для $i=\overline{1,n}$ на i-м месте в этой перестановке

поставим 1, если $a_i \in B$, и 0, если $a_i \notin B$. Получим взаимно однозначное отображение множества всех подмножеств множества A на множество всех $(2,n)$-перестановок с повторениями чисел $0,1$.

Согласно 2.1.11 число различных $(2,n)$-перестановок с повторениями равно 2^n. Столько же различных подмножеств имеет множество A. □

Следует обратить внимание на то, что при подсчете числа подмножеств множества A учитывались пустое подмножество (ему соответствует $(2,n)$-перестановка, состоящая из нулей) и само множество A (которому соответствует $(2,n)$-перестановка, состоящая из единиц).

Пример. Рассмотрим множество $\{0,1,2\}$. Перечислим все его подмножества и соответствующие им наборы из нулей и единиц:

$$\{0,1,2\} \to (1,1,1),\ \{0,1\} \to (1,1,0),\ \{0,2\} \to (1,0,1),$$

$$\{1,2\} \to (0,1,1), \{0\} \to (1,0,0),\ \{1\} \to (0,1,0),$$

$$\{2\} \to (0,0,1),\ \varnothing \to (0,0,0). \qquad \square$$

В рассмотренном примере и при доказательстве утверждения 2.1.12 был использован следующий прием. Элементы n-множества A некоторым образом были упорядочены: $A = \{a_1, a_2, \ldots, a_n\}$, и каждому подмножеству $B \subseteq A$ была взаимно однозначно сопоставлена строка

$$(b_1, b_2, \ldots, b_n)$$

из нулей и единиц по следующему правилу:

$$b_i = \begin{cases} 1, & \text{если } a_i \in B\,, \\ 0, & \text{если } a_i \notin B\,. \end{cases}$$

Сказанное наводит на мысль о том, что можно **перечислить (породить, сгенерировать)** все подмножества множества A, перечислив некоторым регулярным способом все наборы из нулей и единиц длины n. Это можно сделать, например, таким способом.

За основу возьмем двоичные числа. Напомним, что двоичная запись чисел преобразуется в десятичную следующим образом:

$$0 = 0,\ 1 = 1,\ 10 = 2^1 + 0 = 2,\ 11 = 2^1 + 1 = 3,$$
$$100 = 2^2 + 0 \cdot 2 + 0 = 4,\ 101 = 2^2 + 0 \cdot 2 + 1 = 5,$$
$$110 = 2^2 + 2^1 + 0 = 6,\ \ldots.$$

Получаем следующее правило прибавления единицы:

$$0+1=1,\ 1+1=10,\ 10+1=11,\ 11+1=100,$$
$$100+1=101,\ 101+1=110,\ 110+1=111,\ \ldots.$$

Будем считать двоичным числом каждый набор длины n из нулей и единиц. Берем число $0\ldots000$, прибавляем к нему единицу, получаем число $0\ldots001$, прибавляем к нему единицу, получаем число $0\ldots010$ и т. д. Через конечное число шагов получим все наборы из нулей и единиц длины n.

Пример. Возьмем множество $\{a,b,c\}$ и сгенерируем все его подмножества указанным выше способом:

$$\begin{aligned}
000 &\to \varnothing,\\
000+001=001 &\to \{c\},\\
001+001=010 &\to \{b\},\\
010+001=011 &\to \{b,c\},\\
011+001=100 &\to \{a\},\\
100+001=101 &\to \{a,c\},\\
101+001=110 &\to \{a,b\},\\
110+001=111 &\to \{a,b,c\}.
\end{aligned}$$

□

2.2. БИНОМИАЛЬНЫЕ КОЭФФИЦИЕНТЫ

2.2.1. *Биномиальным коэффициентом* называется число всевозможных (n,k)-сочетаний элементов n-множества. Существует три варианта обозначения этого числа:

$$\binom{n}{k},\quad C(n,k),\quad C_n^k.$$

Для удобства выкладок считают, что $C_n^0=1$.

Как правило, число (n,k)-сочетаний мы будем обозначать через C_n^k.

2.2.2. *Число* C_n^k *всевозможных* (n,k)*-сочетаний элементов* n*-множества равно*

$$\frac{n(n-1)\ldots(n-k+1)}{k!}=\frac{n!}{k!(n-k)!}.$$

Доказательство. Согласно 2.1.10 число $P(n,k)$ различных (n,k)-перестановок элементов некоторого n-множества равно $n(n-1)\dots$ $\dots(n-k+1)$. Каждая из этих перестановок является упорядоченным множеством, и две перестановки, образованные одними и теми же элементами, расположенными в различном порядке, считаются разными. Однако нас интересуют сочетания, которые являются различными только тогда, когда они образованы разными множествами элементов. Чтобы найти число различных (n,k)-сочетаний, следует $P(n,k)$ разделить на число попарно различных (n,k)-перестановок, составленных из одних и тех же элементов. Это число совпадает с $P(k,k)$ и равно $k!$. Таким образом,

$$C_n^k = \frac{n(n-1)\dots(n-k+1)}{k!} = \frac{n!}{k!(n-k)!}\,. \qquad \square$$

Пример. Для встречи с ректором от группы, состоящей из 25 студентов, требуется отправить делегацию из трех человек. Сколькими способами можно сформировать такую делегацию?

Очевидно, количество способов выбрать делегацию совпадает с числом сочетаний из 25 элементов по 3. Вычислим его:

$$C_{25}^3 = \frac{25\cdot 24\cdot 23}{3!} = 25\cdot 4\cdot 23 = 2300. \qquad \square$$

2.2.3. *Справедлива формула*

$$C_n^0 + C_n^1 + \ldots + C_n^n = 2^n\,. \tag{2.2.1}$$

Доказательство. Для каждого $k \in \{0,1,\dots,n\}$ число C_n^k совпадает с числом различных подмножеств n-множества, содержащих по k элементов. Общее число различных подмножеств каждого n-множества равно 2^n (см. 2.1.12), следовательно, формула (2.2.1) верна. $\square$

2.2.4. *При $n \geqslant k$ справедливы равенства*

$$C_n^k = C_n^{n-k}\,, \quad C_n^k = C_{n-1}^k + C_{n-1}^{k-1}\,.$$

Доказательство. Справедливость первого равенства почти очевидна:

$$C_n^{n-k} = \frac{n!}{(n-k)!(n-(n-k))!} = \frac{n!}{k!(n-k)!} = C_n^k\,.$$

Докажем второе равенство. Пусть A — некоторое n-множество и $x \in A$. Разделим совокупность всех (n,k)-сочетаний из множества A на два класса. В первый включим все (n,k)-сочетания, содержащие x,

а во второй — не содержащие x. Каждое сочетание из первого класса получается добавлением к элементу x дополнительно $(k-1)$-го элемента из множества $A \setminus \{x\}$, содержащего $(n-1)$ элемент. Ввиду этого в первом классе содержится столько выборок, сколько существует $(n-1, k-1)$-сочетаний, а их согласно 2.2.1 имеется C_{n-1}^{k-1}.

Число выборок во втором классе равно числу C_{n-1}^{k} различных сочетаний по k элементов из множества $A \setminus \{x\}$, содержащего $n-1$ элемент. Получаем равенство $C_n^k = C_{n-1}^k + C_{n-1}^{k-1}$. □

Формулы из утверждения 2.2.4 дают новый способ последовательного вычисления биномиальных коэффициентов. Следующая таблица называется **треугольником Паскаля**:

$$
\begin{array}{ccccccccc}
 & & & & 1 & & & & \\
 & & & 1 & & 1 & & & \\
 & & 1 & & 2 & & 1 & & \\
 & 1 & & 4 & & 6 & & 4 & 1 \\
1 & & 5 & & 10 & & 10 & & 5 & & 1 \\
\multicolumn{9}{c}{\dots\dots\dots\dots\dots\dots\dots\dots\dots}
\end{array}
$$

Будем считать, что в этой таблице вверху расположена строка с номером 0, затем с номером 1 и т. д. При соблюдении этого соглашения строка с номером n состоит из биномиальных коэффициентов C_n^k, $k = \overline{0, n}$. Действительно, для начальных строк в этом легко убедиться прямой проверкой. Элементы строки с номером $n > 1$ получены из элементов предыдущей строки следующим образом. Первый элемент равен 1, т. е. C_n^0. Элемент a_k, $k = \overline{2, n}$, стоящий на месте k, получается сложением двух элементов строки с номером $n-1$, расположенных над ним левее и правее. Эти элементы находятся в строке на местах $k-1$ и k и равны по предположению C_{n-1}^{k-1} и C_{n-1}^k соответственно. Таким образом,

$$a_k = C_{n-1}^k + C_{n-1}^{k-1},$$

что соответствует второй формуле из 2.2.4. Последний элемент строки равен 1, т. е. C_n^n. Первая из формул в 2.2.4 означает, что достаточно вычислить лишь половину элементов каждой строки, остальные повторяют найденные, располагаясь в порядке убывания.

2.2.5. *Число появлений элемента в выборке называется* ***кратностью*** *этого элемента.*

2.2.6. *Число всевозможных (n, r)-сочетаний с повторениями равно* $C_{n+r-1}^{n-1} = C_{n+r-1}^{r}$.

Доказательство. Очевидно, достаточно подсчитать число различных (n,r)-сочетаний с повторениями из множества чисел $\{1,2,\ldots,n\}$. Любое такое сочетание представляет собой выборку, состоящую из r чисел, причем некоторые из них могут повторяться. Порядок чисел в выборке не имеет значения, важен лишь ее состав. Ввиду этого каждую такую выборку объема r будем записывать в виде

$$(a_1, a_2, \ldots, a_r)\,, \tag{2.2.2}$$

где

$$a_1 \leqslant a_2 \leqslant \ldots \leqslant a_r\,.$$

Сначала столько раз записывается наименьшее среди чисел выборки, какова его кратность, затем следующее по величине число тоже столько раз, какова его кратность, и т. д.

В сочетании (2.2.2) соседние числа могут оказаться равными. Перейдем к выборкам, в которых числа попарно различны. Воспользуемся следующим наблюдением. Рассмотрим конечную неубывающую последовательность натуральных чисел

$$p_1, p_2, \ldots, p_k, \quad p_1 \leqslant p_2 \leqslant \ldots \leqslant p_k\,,$$

и строго возрастающую последовательность натуральных чисел

$$q_1, q_2, \ldots, q_k, \quad q_1 < q_2 < \ldots < q_k\,.$$

Складывая соответствующие члены этих последовательностей, всегда получим строго возрастающую последовательность. Действительно, если $s_i = p_i + q_i$, $i = \overline{1,k}$, то

$$s_1 < s_2 < \ldots < s_k\,.$$

Заметим, что числа в сочетании (2.2.2) образуют неубывающую последовательность. Прибавим к каждому члену этой последовательности соответствующий член строго возрастающей последовательности

$$0, 1, 2, \ldots, r-1\,.$$

Положим

$$b_i = a_i + (i-1)\,, \quad i = \overline{0, r-1}\,.$$

Выборке (2.2.2) сопоставим выборку

$$(b_1, b_2, \ldots, b_r) \tag{2.2.3}$$

из множества чисел $\{0, 1, \ldots, n+r-1\}$.

Из способа получения выборки (2.2.3) следует, что образующие ее числа удовлетворяют неравенствам

$$b_1 < b_2 < \ldots < b_r\,.$$

Другими словами, в (2.2.3) все числа попарно различны, и эта выборка является $(n+r-1,r)$-сочетанием. То, что числа в ней выписаны в порядке возрастания, несущественно: так можно записать числа, принадлежащие любому $(n+r-1,r)$-сочетанию. Важно то, что выборки (2.2.2) и (2.2.3) находятся во взаимно однозначном соответствии: если дана выборка (2.2.2), из нее однозначно строится выборка (2.2.3) указанным выше способом, а из выборки (2.2.3) восстанавливается набор (2.2.2), если воспользоваться формулами $a_i = b_i - i + 1$. Таким образом, число всевозможных (n,r)-сочетаний с повторениями совпадает с числом различных $(n+r-1,r)$-сочетаний (без повторений), равным C^r_{n+r-1}. Осталось заметить, что согласно 2.2.4

$$C^r_{n+r-1} = C^{n+r-1-r}_{n+r-1} = C^{n-1}_{n+r-1}\,. \qquad \square$$

Пример. Вася хочет купить себе и своей подруге по мороженому. В киоске имеется 6 видов мороженого. Сколькими способами Вася может выбрать 2 мороженых?

Обозначим виды мороженого через $s_1,\ldots,s_6$. Очевидно, если Вася купил первое мороженое вида s_i, а второе — вида s_j, то его покупка не будет отличаться от покупки, в которой первое мороженое оказалось вида s_j, а второе — вида s_i. Это означает, что речь идет о сочетаниях с повторениями. Число всевозможных $(6,2)$-сочетаний с повторениями равно

$$C^2_7 = 21\,. \qquad \square$$

2.2.7. *Для любых $a,b\in\mathbb{R}$ и любого $n\in\mathbb{N}$ справедлива формула*

$$(a+b)^n = \sum_{k=0}^{n} C^k_n a^{n-k} b^k\,. \tag{2.2.4}$$

Доказательство. Левая часть в (2.2.4) представляет собой сумму произведений вида

$$c_1 c_2 \ldots c_n\,, \tag{2.2.5}$$

в которых для каждого $i\in\{1,\ldots,n\}$ множитель c_i равен либо a, либо b. Эту сумму обозначим через S. Процесс образования произведений

(2.2.5) можно описать следующим образом. Представляем левую часть в (2.2.4) в виде произведения

$$\underbrace{(a+b)(a+b)\dots(a+b)}_{n \text{ раз}}.$$

Двигаясь слева направо и выбирая из каждой скобки либо a, либо b, получаем произведение вида (2.2.5).

Каждому образованному таким образом произведению сопоставим последовательность

$$(\alpha_1, \alpha_2, \dots, \alpha_n) \tag{2.2.6}$$

из нулей и единиц, ставя на i-м месте 0, если на i-м шаге мы выбрали a, единицу, если выбрали b. Очевидно, разным произведениям (2.2.6), т. е. разным слагаемым в сумме S, соответствуют разные n-ки (2.2.6). Перестановкой сомножителей каждое произведение (2.2.5) можно привести к виду $a^{n-k}b^k$. Это означает, что среди слагаемых в S есть совпадающие по величине, но различающиеся порядком сомножителей. Обозначим через S_k, $k \in \{0, \dots, n\}$, часть суммы S, составленную из всех тех слагаемых, которые по величине равны $a^{n-k}b^k$. Каждому слагаемому поставлена в соответствие n-ка (2.2.6) из нулей и единиц, содержащая $n-k$ нулей и k единиц. Число таких n-к равно числу способов выбора из n мест тех k, в которые попадут единицы. Согласно 2.2.2 оно равно C_n^k. Видим, что S_k содержит C_n^k слагаемых, а каждое слагаемое равно $a^{n-k}b^k$. Это означает, что

$$S_k = C_n^k a^{n-k} b^k .$$

Осталось сложить все суммы S_k:

$$S = S_0 + S_1 + \ldots + S_n = C_n^0 a^n + C_n^1 a^{n-1} b + \ldots + C_n^n a^0 b^n = \\ = \sum_{k=0}^{n} C_n^k a^{n-k} b^k . \qquad \square$$

Формулу (2.2.4) обычно называют **биномом Ньютона**. Считается, однако, что эта формула была известна знаменитому ученому и поэту Омару Хайяму (1048–1131) и другим математикам, жившим задолго до Ньютона. Слово «бином» образовано из латинской приставки *bi*, которую в данном случае можно перевести как *состоящий из двух*, и греческого слова *nome*, означающего *доля, часть*. Таким образом, слову бином соответствует слово двучлен. Стоит упомянуть, что многочлен часто называют **полиномом**.

2.3. ПОЛИНОМИАЛЬНЫЕ КОЭФФИЦИЕНТЫ

2.3.1. *Неупорядоченная система $\{A_1, A_2, \ldots, A_k\}$ непустых подмножеств множества A называется (неупорядоченным)* ***разбиением множества*** *A, если она обладает двумя свойствами:*

1) *объединение всех подмножеств совпадает с A;*

2) *образующие эту систему подмножества не пересекаются друг с другом.*

Термин «разбиение» легко запомнить, если представить себе керамический горшок, который уронили столь удачно, что при сложении осколков можно восстановить весь горшок. Очевидно, различные осколки общих частей не имеют.

Пример. Пусть $A = \{a_1, a_2, a_3\}$. Следующие системы подмножеств множества A являются разбиениями:

$$\{\{a_1, a_2, a_3\}\}, \quad \{\{a_1\}, \{a_2, a_3\}\}, \quad \{\{a_2\}, \{a_1, a_3\}\},$$

$$\{\{a_3\}, \{a_1, a_2\}\}, \quad \{\{a_1\}, \{a_2\}, \{a_3\}\}. \qquad \square$$

2.3.2. *Упорядоченная система $\{A_1, A_2, \ldots, A_k\}$ подмножеств множества A называется $(r_1, r_2, \ldots, r_k)$-****разбиением****, если выполняются два условия:*

а) *подмножества $A_1, A_2, \ldots, A_k$ образуют разбиение множества A;*

б) $|A_1| = r_1,\ |A_2| = r_2, \ldots, |A_k| = r_k$.

Пример. Пусть $A = \{a_1, a_2, a_3\}$.

а) Есть только одно (3)-разбиение множества A:

$$\{\{a_1, a_2, a_3\}\}.$$

б) Существует три $(1, 2)$-разбиения:

$$\{\{a_1\}, \{a_2, a_3\}\}, \quad \{\{a_2\}, \{a_1, a_3\}\}, \quad \{\{a_3\}, \{a_1, a_2\}\}.$$

в) Имеется $3! = 6$ различных $(1, 1, 1)$-разбиений:

$$\{\{a_1\}, \{a_2\}, \{a_3\}\}, \quad \{\{a_2\}, \{a_3\}, \{a_1\}\}, \quad \{\{a_3\}, \{a_1\}, \{a_2\}\},$$
$$\{\{a_3\}, \{a_2\}, \{a_1\}\}, \quad \{\{a_2\}, \{a_1\}, \{a_3\}\}, \quad \{\{a_1\}, \{a_3\}, \{a_2\}\}. \qquad \square$$

2.3.3. *Число $C_n^{r_1, r_2, \ldots, r_k}$ различных $(r_1, r_2, \ldots, r_k)$-разбиений n-множества равно*

$$\frac{n!}{r_1! r_2! \ldots r_k!}.$$

Доказательство. Предположим для определенности, что требуется разбить n-множество A на непересекающиеся подмножества A_1, A_2, …, A_k, содержащие соответственно r_1, r_2, …, r_k элементов. Согласно 2.2.2 множество A_1 можно выбрать $C_n^{r_1}$ способами. Когда оно выбрано, элементы множества A_2 выбираются среди оставшихся $n-r_1$ элементов множества A. Этот выбор можно осуществить $C_{n-r_1}^{r_2}$ способами. Затем выбираем множество A_3, и т. д. По правилу произведения общее число способов выбора подмножеств $A_1, A_2, \ldots, A_k$ равно

$$C_n^{r_1} \cdot C_{n-r_1}^{r_2} \cdot \ldots \cdot C_{n-r_1-r_2-\ldots-r_{k-1}}^{r_k} = \\ = \frac{n!}{r_1!(n-r_1)!} \cdot \frac{(n-r_1)!}{r_2!(n-r_1-r_2)!} \cdot \frac{(n-r_1-r_2)!}{r_3!(n-r_1-r_2-r_3)!} \cdot \ldots \\ \ldots \cdot \frac{(n-r_1-\ldots-r_{k-1})!}{r_k!(n-r_1-\ldots-r_k)!}.$$

В этой цепочке дробей числитель каждой дроби начиная со второй содержится в знаменателе предыдущей дроби. После сокращений получаем дробь

$$\frac{n!}{r_1!r_2!\ldots r_k!(n-r_1-\ldots-r_k)!}.$$

Так как $r_1+\ldots+r_k=n$, то $(n-r_1-\ldots-r_k)!=0!=1$ и потому

$$C_n^{r_1,r_2,\ldots,r_k} = \frac{n!}{r_1!r_2!\ldots r_k!} \qquad \square$$

2.3.4. *Величина*

$$C_n^{r_1,r_2,\ldots,r_k} = \frac{n!}{r_1!r_2!\ldots r_k!}$$

называется ***полиномиальным коэффициентом****. Для нее существует также другое обозначение:*

$$\binom{n}{r_1, r_2, \ldots, r_k}.$$

Пример 1. Четырех человек надо разделить на две команды по два игрока. Сколькими способами это можно сделать?

Находим число различных $(2,2)$-разбиений множества из четырех элементов:

$$C_4^{2,2} = \frac{4!}{2!2!} = \frac{24}{4} = 6.$$

Этот результат легко проверить. Обозначим спортсменов буквами a, b, c, d. Первую команду можно сформировать следующими способами:

$$\{a,b\}, \quad \{a,c\}, \quad \{a,d\}, \quad \{b,c\}, \quad \{b,d\}, \quad \{c,d\}.$$

В каждом случае оставшиеся два спортсмена образуют вторую команду. □

Пример 2. Компания из шести человек зашла в кафе и обнаружила, что свободны три столика. За каждый столик могут сесть два человека. Сколькими способами они могут сесть за столики?

Шесть человек можно разделить на пары $C_6^2 = 15$ способами. Пары можно рассадить за три столика $3! = 6$ способами. Всего же имеется $15 \cdot 6 = 90$ вариантов рассадки.

Найденное число совпадает с числом различных $(2,2,2)$-разбиений множества из шести элементов:

$$C_6^{2,2,2} = \frac{6!}{2!2!2!} = \frac{6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1}{8} = 90\,. \qquad \square$$

2.3.5. *Для любых $a_1, \ldots, a_k \in \mathbb{R}$ и любого $n \in \mathbb{N}$ справедлива формула*

$$(a_1 + \ldots + a_k)^n = \sum_{\substack{r_1 \geqslant 0, \ldots, r_k \geqslant 0, \\ r_1 + \ldots + r_k = n}} \frac{n!}{r_1! r_2! \ldots r_k!} a_1^{r_1} \ldots a_k^{r_k}\,. \tag{2.3.1}$$

Доказательство. Левая часть в (2.3.1) представляет собой сумму произведений вида

$$c_1 c_2 \ldots c_n\,, \tag{2.3.2}$$

в которых для каждого $i \in \{1, \ldots, n\}$ множитель c_i равен одному из чисел $a_1, \ldots, a_k$. Эту сумму обозначим через S. Через $T(r_1, \ldots, r_k)$ обозначим множество всех тех слагаемых из S, в которых a_1 является сомножителем r_1 раз, a_2 — сомножителем r_2 раз, и т. д. Любые два элемента из $T(r_1, \ldots, r_k)$ отличаются только порядком сомножителей.

Чтобы задать одно произведение, входящее в $T(r_1, \ldots, r_k)$, достаточно указать номера мест в произведении (2.3.2), на которых стоит число a_1, затем номера мест, на которых стоит число a_2, и т. д. Иначе говоря, необходимо произвести некоторое $(r_1, r_2, \ldots, r_k)$-разбиение множества $\{1, 2, \ldots, n\}$. Очевидно, число элементов в $T(r_1, \ldots, r_k)$ равно числу $C_n^{r_1, r_2, \ldots, r_k}$ различных $(r_1, r_2, \ldots, r_k)$-разбиений множества $\{1, 2, \ldots, n\}$. Поскольку перестановкой сомножителей каждый элемент из $T(r_1, \ldots, r_k)$ можно превратить в произведение $a_1^{r_1} \cdot \ldots \cdot a_k^{r_k}$, сумма всех элементов из $T(r_1, \ldots, r_k)$ равна

$$C_n^{r_1, r_2, \ldots, r_k} a_1^{r_1} \ldots, a_k^{r_k} = \frac{n!}{r_1! r_2! \ldots r_k!} a_1^{r_1} \ldots a_k^{r_k}\,.$$

Так как при различных наборах

$$r_1^{(1)}, \ldots, r_k^{(1)}, \quad r_1^{(2)}, \ldots, r_k^{(2)}$$

множества

$$T(r_1^{(1)},\dots,r_k^{(1)}),\quad T(r_1^{(2)},\dots,r_k^{(2)})$$

не пересекаются и

$$S=\bigcup_{\substack{r_1\geqslant 0,\dots,r_k\geqslant 0,\\ r_1+\dots+r_k=n}} T(r_1,\dots,r_k),$$

справедливо равенство (2.3.1). □

Отметим, что при $k=2$ равенство (2.3.1) превращается в уже знакомую формулу бинома Ньютона:

$$(a_1+a_2)^n=\sum_{\substack{r_1\geqslant 0,r_2\geqslant 0,\\ r_1+r_2=n}}\frac{n!}{r_1!r_2!}a_1^{r_1}a_2^{r_2}=\sum_{k=0}^{n}C_n^k a_1^{n-k}a_2^k\,.$$

Последняя сумма получена из предыдущей заменой r_1 на $n-k$ и r_2 на k, так как $r_1+r_2=n$.

2.4. ФОРМУЛА ВКЛЮЧЕНИЙ И ИСКЛЮЧЕНИЙ

2.4.1. *Пусть имеется N предметов и n свойств $\alpha_1,\dots,\alpha_n$. Каждый из предметов может обладать, а может и не обладать любым из этих свойств. Через $N(\alpha_{i_1},\dots,\alpha_{i_k})$ обозначим число предметов, обладающих по крайней мере свойствами $\alpha_{i_1},\dots,\alpha_{i_k}$. Число $\overline{N}$ предметов, не обладающих ни одним из свойств $\alpha_1,\dots,\alpha_n$, удовлетворяет равенству*

$$\begin{aligned}\overline{N}=N-N(\alpha_1)-N(\alpha_2)-\dots-N(\alpha_n)+\\ +N(\alpha_1,\alpha_2)+N(\alpha_1,\alpha_3)+\dots+N(\alpha_{n-1},\alpha_n)-\\ -N(\alpha_1,\alpha_2,\alpha_3)-\dots-N(\alpha_{n-2},\alpha_{n-1},\alpha_n)+\dots\\ \dots+(-1)^n N(\alpha_1,\alpha_2,\dots,\alpha_n)\,.\end{aligned}\tag{2.4.1}$$

Доказательство. Докажем утверждение индукцией по числу рассматриваемых свойств. Если имеется только одно свойство α, то каждый предмет либо обладает этим свойством, либо не обладает, следовательно,

$$\overline{N}=N-N(\alpha).$$

Предположим, что утверждение справедливо, если рассматривается $n-1$ свойств. Перепишем равенство (2.4.1) для этого случая, обозначив

через $\overline{N}_1$ число предметов, не обладающих свойствами $\alpha_1, \ldots, \alpha_{n-1}$:

$$\begin{aligned}\overline{N}_1 = N - N(\alpha_1) - N(\alpha_2) - \ldots - N(\alpha_{n-1}) + \\ + N(\alpha_1, \alpha_2) + N(\alpha_1, \alpha_3) + \ldots + N(\alpha_{n-2}, \alpha_{n-1}) - \\ - N(\alpha_1, \alpha_2, \alpha_3) - \ldots - N(\alpha_{n-3}, \alpha_{n-2}, \alpha_{n-1}) + \ldots \\ \ldots + (-1)^{n-1} N(\alpha_1, \alpha_2, \ldots, \alpha_{n-1}) .\end{aligned} \tag{2.4.2}$$

Докажем, что теорема верна и в случае, когда рассматривается n свойств.

Если никакой из рассматриваемых предметов не обладает ни одним из свойств α_i, $i = \overline{1,n}$, то $\overline{N} = N$. В этом случае теорема верна.

Рассмотрим случай, когда каким-то из рассматриваемых свойств обладает по крайней мере один предмет. Для упрощения записей будем считать, что этим свойством является α_n. Обозначим через $\overline{N}(\alpha_n)$ число предметов, обладающих свойством α_n, но не обладающих свойствами $\alpha_1, \ldots, \alpha_{n-1}$. Для $(n-1)$-го свойства утверждение 2.4.1 справедливо, поэтому

$$\begin{aligned}\overline{N}(\alpha_n) = N(\alpha_n) - N(\alpha_1, \alpha_n) - N(\alpha_2, \alpha_n) - \ldots - N(\alpha_{n-1}, \alpha_n) + \\ + N(\alpha_1, \alpha_2, \alpha_n) + N(\alpha_1, \alpha_3, \alpha_n) + \ldots \\ \ldots + N(\alpha_{n-2}, \alpha_{n-1}, \alpha_n) - \\ - N(\alpha_1, \alpha_2, \alpha_3, \alpha_n) - \ldots - N(\alpha_{n-3}\alpha_{n-2}, \alpha_{n-1}, \alpha_n) + \ldots \\ \ldots + (-1)^{n-1} N(\alpha_1, \alpha_2, \ldots, \alpha_n) .\end{aligned} \tag{2.4.3}$$

Вычитая (2.4.3) из (2.4.2), получим равенство, правая часть которого совпадает с правой частью в (2.4.1), а левая часть равна

$$\overline{N}_1 - \overline{N}(\alpha_n) .$$

В этой разности $\overline{N}_1$ — число предметов, не обладающих свойствами $\alpha_1, \ldots, \alpha_{n-1}$, а $\overline{N}(\alpha_n)$ — число предметов, также не обладающих свойствами $\alpha_1, \ldots, \alpha_{n-1}$, но обладающих свойством α_n. Очевидно,

$$\overline{N}_1 - \overline{N}(\alpha_n) = \overline{N} .$$

Видим, что равенство (2.4.1) получается как разность истинных равенств (2.4.2) и (2.4.3), значит, оно также справедливо. □

Пример. В деревне Сидоровке живет 15 семей и только одна из них не имеет скотины. Одиннадцать семей держат коров, пять — овец, восемь — свиней. Три семьи держат коров и овец, четыре — коров

и свиней, четыре — овец и свиней. Сколько семей держит и коров, и овец, и свиней?

Всего имеется 15 семей (всего предметов). Сложив количества семей, имеющих хотя бы один вид скотины (обладающих одним свойством), получим число

$$11 + 5 + 8 = 24\,.$$

Суммируя количества семей, имеющих не менее двух видов скотины (обладающих двумя свойствами), получим 11. Не имеет скотины (не обладает ни одним свойством) 1 семья. Пусть X семей держит и коров, и овец, и свиней. Применяем формулу включений и исключений: $1 = 15 - 24 + 11 - X$, откуда $X = 1$. □

2.5. РЕКУРРЕНТНЫЕ СООТНОШЕНИЯ

Свойства решений однородных систем линейных уравнений

$$\begin{cases} \alpha_{11}x_1 + \ldots + \alpha_{1n}x_n = 0, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \alpha_{m1}x_1 + \ldots + \alpha_{mn}x_n = 0 \end{cases}$$

с n неизвестными и m уравнениями, а также способы отыскания этих решений изучаются в курсе линейной алгебры. Здесь мы рассмотрим те же проблемы для однородных систем, составленных из бесконечного числа уравнений и имеющих бесконечное число неизвестных. При этом мы ограничимся только случаем, когда система имеет следующий вид:

$$\begin{cases} \alpha_1x_1 + \ldots + \alpha_{k+1}x_{k+1} = 0, \\ \alpha_1x_2 + \ldots + \alpha_{k+1}x_{k+2} = 0, \\ \alpha_1x_3 + \ldots + \alpha_{k+1}x_{k+3} = 0, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \alpha_1x_n + \ldots + \alpha_{k+1}x_{n+k} = 0, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \end{cases} \tag{2.5.1}$$

Будем предполагать, что коэффициенты во всех уравнениях и значения неизвестных являются действительными числами. Каждое решение такой системы представляет собой бесконечную совокупность чисел. При подстановке первого из этих чисел вместо x_1, второго вместо x_2 и т. д. каждое из уравнений должно превратиться в тождество. Когда мы говорим «первое число a», «второе число b» и т. д., то тем самым неявно предполагаем, что существует функция, сопоставляющая единице число a, двойке число b и т. д. Это служит основанием для следующего определения.

2.5.1. *Последовательностью элементов множества* A *называется функция, отображающая множество* $\mathbb{N} = \{1, 2, 3, \ldots\}$ *в* A.

Элементом последовательности $f : \mathbb{N} \to A$ *является упорядоченная пара* (n, a), *в которой* $a = f(n)$. *Обычно эта пара обозначается через* a_n, *а последовательность* $f : \mathbb{N} \to A$ *— через* $\{a_n\}$ *или через* $a_1, a_2, a_3, \ldots$.

Согласно этому определению в последовательности есть первый член, второй и т. д. Однако часто нумерацию начинают с нуля, и тогда **последовательностью** называют функцию, определенную на множестве $\mathbb{N}_0 = \{0, 1, 2, \ldots\}$. От способа нумерации зависит вид некоторых формул, что следует принимать во внимание.

Примеры. Полагая

$$f(1) = 1, \quad f(2) = \frac{1}{2}, \quad f(3) = \frac{1}{3}, \ldots, \ f(n) = \frac{1}{n}, \ldots,$$

задаем функцию $f(n)$, отображающую множество $\mathbb{N}$ в множество $\mathbb{R}$. Согласно 2.5.1 эта функция называется последовательностью. Ту же последовательность можно изобразить, перечисляя лишь значения этой функции:

$$1, \quad \frac{1}{2}, \quad \frac{1}{3}, \quad \frac{1}{4}, \quad \frac{1}{5}, \ldots.$$

Ее можно записать и таким образом: $\{\frac{1}{n}\}$.

Значения функции, определенной на множестве $\mathbb{N}$, т. е. элементы последовательности, можно назначать произвольным образом. Ввиду этого следующие функции также являются последовательностями:

$$-2, \ -1, \ 0, \ 1, \ 2, \ 1, \ 0, \ -1, \ -2, \ -1, \ 0, \ldots,$$

$$\sin x, \ \cos x, \ 2\sin x, \ 2\cos x, \ 3\sin x, \ 3\cos x, \ldots. \qquad \square$$

Уравнения в системе (2.5.1) отличаются только нумерацией неизвестных, коэффициенты, стоящие на одинаковых местах, во всех уравнениях совпадают. Вследствие этого можно всю систему записать в виде одного уравнения

$$\alpha_1 x_n + \ldots + \alpha_{k+1} x_{n+k} = 0\,,$$

имея в виду, что если брать n равным $1, 2, \ldots$, то получатся все уравнения системы. Запись этого уравнения также можно упростить, если предполагать, что коэффициент α_{k+1} не равен нулю (в противном случае последние члены во всех уравнениях можно не писать). Поделив на

него обе части и записав все слагаемые в обратном порядке, получим равенство

$$x_{n+k} + \frac{\alpha_k}{\alpha_{k+1}} x_{n+k-1} + \ldots + \frac{\alpha_1}{\alpha_{k+1}} x_n = 0\,.$$

Обозначая дроби новыми буквами, придадим ему следующий вид:

$$x_{n+k} + p_1 x_{n+k-1} + \ldots + p_k x_n = 0\,. \tag{2.5.2}$$

2.5.2. *Соотношение* (2.5.2) *называется* ***возвратным уравнением порядка*** k.

2.5.3. *Любая последовательность, удовлетворяющая возвратному уравнению* (2.5.2), *называется его* ***решением***.

Пример. Арифметические прогрессии

$$5,\ 7,\ 9,\ 11\ldots,\quad 2,\ 6,\ 10,\ 14\ldots$$

удовлетворяют возвратному уравнению

$$a_{n+2} - 2a_{n+1} + a_n = 0$$

и потому являются его решениями. □

2.5.4. *Если при фиксированном* k *члены* $a_n, a_{n+1}, \ldots, a_{n+k}$ *последовательности* $a_1, a_2, \ldots, a_n, \ldots$ *для каждого* $n \in \mathbb{N}$ *удовлетворяют уравнению* (2.5.2), *то последовательность называется* ***возвратной*** *(или* ***рекуррентной***) ***порядка*** k.

Иначе говоря, последовательность является возвратной тогда и только тогда, когда она является решением подходящей системы вида (2.5.1). Если порядок такой последовательности равен k, то каждое из уравнений системы содержит $k+1$ неизвестных.

Примеры. 1. Последовательность $a_1, a_2, \ldots$ является геометрической прогрессией со знаменателем $p \neq 0$, если $a_{n+1} = p \cdot a_n$ для каждого n. Ее члены a_{n+1}, a_n удовлетворяют возвратному уравнению $x_{n+1} - p \cdot x_n = 0$. В нем число неизвестных равно двум, поэтому геометрическая прогрессия является возвратной последовательностью первого порядка.

2. Если последовательность $a_1, a_2, \ldots$ является арифметической прогрессией и r — ее разность, то

$$a_{n+1} = a_n + r, \quad a_{n+2} = a_{n+1} + r, \quad n = 1, 2, \ldots.$$

Вычитая первое равенство из второго, получаем

$$a_{n+2} - a_{n+1} = a_{n+1} - a_n, \quad a_{n+2} - 2a_{n+1} + a_n = 0\,.$$

Последнее равенство показывает, что арифметическая прогрессия является возвратной последовательностью второго порядка. □

Замечание. Термин «прогрессия» образован от латинского слова *progressio* — движение вперед, возрастание.

2.5.5. *Соотношение вида*

$$a_{n+k} = F(n, a_n, a_{n+1}, \ldots, a_{n+k-1})\,, \tag{2.5.3}$$

где F — некоторая фиксированная функция, называется ***рекуррентным порядка*** k.

Название соотношения происходит от латинского слова *recurrere* — возвращаться.

Пример. Соотношение

$$a_{n+2} = n + a_n + a_{n+1}^2$$

является рекуррентным порядка 2. В нем $F(n, a_n, a_{n+1}) = n + a_n + a_{n+1}^2$. □

2.5.6. *Рекуррентное соотношение порядка k позволяет вычислить любой член последовательности $a_1, a_2, a_3, \ldots$, если заданы ее первые k членов.*

Доказательство. Пусть соотношение имеет вид (2.5.3) и известны члены $a_1, a_2, \ldots, a_k$. Полагая в (1) $n = 1$, находим a_{k+1}:

$$a_{k+1} = F(1, a_1, a_2, \ldots, a_k)\,.$$

Теперь нам известны члены $a_2, \ldots, a_{k+1}$ и мы можем вычислить член a_{k+2}:

$$a_{k+2} = F(2, a_2, a_3, \ldots, a_{k+1})\,.$$

Далее вычисляем член a_{k+3} и т. д. □

Примеры. 1. Соотношение в предыдущем примере имело вид

$$a_{n+2} = n + a_n + a_{n+1}^2\,.$$

Если взять $a_1 = 0$ и $a_2 = 1$, то все остальные члены последовательности, задаваемой этими условиями и указанным соотношением, можно вычислить:

$$a_3 = 1 + a_1 + a_2^2 = 1 + 0 + 1^2 = 2\,,$$
$$a_4 = 2 + a_2 + a_3^2 = 2 + 1 + 2^2 = 7\,,$$
$$a_5 = 3 + a_3 + a_4^2 = 3 + 2 + 7^2 = 54\,,$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

2. Для геометрической прогрессии

$$a_1,\ a_2 = 3a_1,\ a_3 = 3^2 a_1,\ \ldots$$

рекуррентным является соотношение $a_{n+1} = 3 \cdot a_n$. Из этого соотношения можно найти все ее члены, как только задан член a_1. Например, если $a_1 = 2$, то $a_2 = 6$, $a_3 = 18$ и т. д.

3. Для арифметической прогрессии

$$a_0,\ a_1 = a_0 + 5,\ a_2 = a_0 + 10,\ \ldots$$

рекуррентным является соотношение $a_{n+1} = a_n + 5$, также позволяющее найти все ее члены, если задан член a_0. Так, при $a_0 = 4$ получаем прогрессию $4, 9, 14, 19, \ldots$. □

Возвратное уравнение (2.5.2) легко превратить в рекуррентное соотношение, перенося все слагаемые, начиная со второго, из левой части в правую:

$$x_{n+k} = -p_1 x_{n+k-1} - \ldots - p_k x_n\,. \tag{2.5.4}$$

Если числа $a_1, a_2, \ldots, a_k$ являются начальными в последовательности, удовлетворяющей уравнению (2.5.4), то согласно 2.5.6 все остальные члены этой последовательности можно вычислить, пользуясь этим соотношением. Другими словами, решение возвратного уравнения (2.5.2) порядка k определено однозначно, как только известны первые k его членов. Эти члены можно выбирать совершенно произвольно, и после каждого выбора будет задано некоторое частное решение уравнения (2.5.2). Это также означает, что существует бесконечное множество различных последовательностей, удовлетворяющих этому уравнению.

Теперь мы постараемся каким-то образом охарактеризовать всю совокупность решений уравнения (2.5.2). Поскольку последовательности являются функциями, можно говорить о сумме последовательностей и о произведении числа на последовательность, имея в виду сумму функций и произведение числа на функцию. Тем не менее приведем отдельные определения для этих операций.

2.5.7. *Произведением числа* $\alpha \in \mathbb{R}$ *на последовательность* $\{x_i\}$ *называется последовательность* $\{\alpha x_i\}$*, каждый член которой получен умножением соответствующего члена последовательности* $\{x_i\}$ *на* α*.*

Суммой последовательностей $\{x_i\}$ *и* $\{y_i\}$ *называется последовательность* $\{x_i + y_i\}$*, каждый член которой равен сумме соответствующих членов последовательностей* $\{x_i\}$ *и* $\{y_i\}$*.*

Примеры. Умножая на 3 последовательность $0, 1, 2, 3, \ldots$, получим последовательность $0, 3, 6, 9, \ldots$.

При сложении последовательности

$$0, 1, 2, 3, \ldots$$

с последовательностью

$$0, -1, -2, -3, \ldots$$

получается последовательность $0, 0, 0, 0, \ldots$. □

2.5.8. *Относительно введенных в* 2.5.7 *операций сложения и умножения на действительное число последовательности действительных чисел образуют векторное пространство.*

Доказательство. Нетрудно убедиться в справедливости всех аксиом векторного пространства. □

Поскольку последовательности действительных чисел образуют векторное пространство, мы можем в дальнейшем пользоваться стандартными терминами и рассматривать линейные комбинации таких последовательностей, линейно независимые системы, подпространства и т. д. Следует только иметь в виду, что пространство, в котором векторами служат бесконечные последовательности действительных чисел, не является конечномерным. Это означает, что в нем не существует конечной системы последовательностей, через которые можно линейно выразить любую другую последовательность.

Если рассматривается несколько последовательностей, то для их различия при записи удобно использовать верхние индексы. Так, три последовательности можно обозначить следующим образом:

$$\{a_i^{(1)}\}, \quad \{a_i^{(2)}\}, \quad \{a_i^{(3)}\}.$$

Умножая их соответственно на γ_1, γ_2 и γ_3, получим последовательности

$$\{\gamma_1 a_i^{(1)}\}, \quad \{\gamma_2 a_i^{(2)}\}, \quad \{\gamma_3 a_i^{(3)}\},$$

складывая — одну последовательность

$$\{\gamma_1 a_i^{(1)} + \gamma_2 a_i^{(2)} + \gamma_3 a_i^{(3)}\},$$

являющуюся линейной комбинацией исходных последовательностей.

2.5.9. *Если возвратные последовательности*

$$\{a_i^{(1)}\},\ \{a_i^{(2)}\},\ \ldots,\ \{a_i^{(s)}\} \tag{2.5.5}$$

удовлетворяют возвратному уравнению (2.5.2)*, то для любых чисел* $\alpha_1, \alpha_2, \ldots, \alpha_s \in \mathbb{R}$ *этому уравнению удовлетворяет также последовательность*

$$\{\alpha_1 a_i^{(1)} + \alpha_2 a_i^{(2)} + \ldots + \alpha_s a_i^{(s)}\}, \tag{2.5.6}$$

и потому множество всех решений уравнения (2.5.2) *образует подпространство в пространстве всех числовых последовательностей.*

Доказательство. По условию последовательности (2.5.5) являются решениями уравнения (2.5.2). Это означает, что для каждого $j = \overline{1,s}$ набор чисел

$$a_{n+k}^{(j)}, a_{n+k-1}^{(j)}, \ldots, a_n^{(j)}$$

является решением уравнения (2.5.2):

$$a_{n+k}^{(j)} + p_1 a_{n+k-1}^{(j)} + \ldots + p_k a_n^{(j)} = 0, \quad j = \overline{1,s}\,.$$

Подставляя в (2.5.2) числа

$$\alpha_1 a_{n+k}^{(1)} + \alpha_2 a_{n+k}^{(2)} + \ldots + \alpha_s a_{n+k}^{(s)},\ \ldots,\ \alpha_1 a_n^{(1)} + \alpha_2 a_{n+k}^{(2)} + \ldots + \alpha_s a_n^{(s)},$$

убеждаемся в том, что они также образуют решение уравнения (2.5.2):

$$\begin{aligned}
&(\alpha_1 a_{n+k}^{(1)} + \alpha_2 a_{n+k}^{(2)} + \ldots + \alpha_s a_{n+k}^{(s)}) + \\
&\quad + p_1(\alpha_1 a_{n+k-1}^{(1)} + \alpha_2 a_{n+k-1}^{(2)} + \ldots + \alpha_s a_{n+k-1}^{(s)}) + \ldots \\
&\qquad\qquad \ldots + p_k(\alpha_1 a_n^{(1)} + \alpha_2 a_n^{(2)} + \ldots + \alpha_s a_n^{(s)}) = \\
&= (a_{n+k}^{(1)} + p_1 a_{n+k-1}^{(1)} + \ldots + p_k a_n^{(1)}) + \ldots \\
&\qquad\qquad \ldots + (a_{n+k}^{(s)} + p_1 a_{n+k-1}^{(s)} + \ldots + p_k a_n^{(s)}) = 0\,.
\end{aligned}$$

Поскольку это справедливо при любом n, последовательность (2.5.6) является решением возвратного уравнения (2.5.2). □

В курсе линейной алгебры доказывается, что *множество решений однородной системы, состоящей из m независимых линейных уравнений с n неизвестными, является векторным пространством размерности $n-m$*. Похожее утверждение справедливо и для возвратных последовательностей, только размерность пространства в этом случае определяется иначе.

2.5.10. *Размерность пространства решений возвратного уравнения* (2.5.4) *порядка k равна k.*

Доказательство. Положим в (2.5.4) $n=0$ и выпишем получившееся уравнение:

$$x_k + p_1 x_{k-1} + \ldots + p_k x_0 = 0\,. \tag{2.5.7}$$

Напомним, что *любой базис в пространстве всех решений однородной системы линейных уравнений называется* ***фундаментальной системой решений***. Выше уже говорилось о том, что размерность пространства решений однородной системы линейных уравнений равна разности между числом неизвестных и числом независимых уравнений. Для системы (2.5.7), состоящей из единственного линейного уравнения (2.5.7) с $k+1$ неизвестными $x_0,\ldots,x_k$, эта разность равна k. Из этого следует, что фундаментальная система решений уравнения (2.5.7) состоит из k решений. Выпишем эти решения:

$$\begin{gathered} u_0^{(1)},\ldots,u_k^{(1)},\\ \ldots\ldots\ldots\ldots\\ u_0^{(k)},\ldots,u_k^{(k)}\,. \end{gathered} \tag{2.5.8}$$

Пусть $\{b_i\}$ — возвратная последовательность, удовлетворяющая уравнению (2.5.2). Ее члены $b_0,\ldots,b_k$ образуют решение системы (2.5.7). Это решение можно представить в виде линейной комбинации решений, принадлежащих базису (2.5.8):

$$(b_0,\ldots,b_k) = \alpha_1(u_0^{(1)},\ldots,u_k^{(1)}) + \ldots + \alpha_k(u_0^{(k)},\ldots,u_k^{(k)})\,.$$

Получаем формулы, позволяющие находить члены $b_0,\ldots,b_k$ последовательности $\{b_i\}$:

$$b_i = \alpha_1 u_i^{(1)} + \ldots + \alpha_k u_i^{(k)}\,,\quad i=\overline{0,k}\,.$$

Ранее уже упоминалось о том, что, как только заданы первые k членов возвратной последовательности, определяемой уравнением (2.5.2)

порядка k, любой другой член этой последовательности может быть найден и потому является однозначно определенным. Обозначим через

$$\{u_i^{(1)}\}, \ldots, \{u_i^{(k)}\}$$

возвратные последовательности, задаваемые начальными условиями (2.5.8) и уравнением (2.5.2). Они образуют линейно независимую систему, так как по предположению линейно независима система (2.5.8), состоящая из конечных последовательностей, образованных первыми k членами.

Рассмотрим последовательность $\{\alpha_1 u_i^{(1)} + \ldots + \alpha_k u_i^{(k)}\}$. Поскольку ее первые k членов совпадают с соответствующими членами последовательности $\{b_i\}$, окажутся равными и все остальные члены этих последовательностей. Видим, что последовательность $\{b_i\}$ является линейной комбинацией последовательностей $\{u_i^{(1)}\}, \ldots, \{u_i^{(k)}\}$, следовательно, последние образуют базис в пространстве всех решений уравнения (2.5.7). □

2.5.11. *Если набор последовательностей*

$$(\{a_i^{(1)}\}, \ldots, \{a_i^{(s)}\})$$

образует базис в пространстве всех решений уравнения (2.5.4)*, то любая последовательность* $\{u_i\}$*, удовлетворяющая этому уравнению, является линейной комбинацией последовательностей, принадлежащих базису, и потому каждый ее член* u_n *представим в виде*

$$u_n = \alpha_1 a_n^{(1)} + \ldots + \alpha_s a_n^{(s)}\,.$$

Это выражение называется ***общим решением возвратного уравнения*** (2.5.2).

2.5.12. *Базис в пространстве всех решений возвратного уравнения назовем* ***базисом*** *этого* ***уравнения****.*

2.5.13. *Пусть последовательность* $\{a_n\}$ *является решением возвратного уравнения*

$$x_{n+k} + p_1 x_{n+k-1} + \ldots + p_k x_n = 0\,. \tag{2.5.9}$$

Характеристическим многочленом возвратной последовательности $\{a_n\}$ *называется многочлен*

$$P(x) = x^k + p_1 x^{k-1} + \ldots + p_k\,. \tag{2.5.10}$$

2.5.14. *Пусть* $\alpha, q \in \mathbb{R}$, $\alpha \neq 0$ *и* $q \neq 0$. *Геометрическая прогрессия* $\alpha, \alpha q, \alpha q^2, \ldots$ *удовлетворяет возвратному уравнению* (2.5.9) *тогда и только тогда, когда ее знаменатель* q *является корнем характеристического многочлена* (2.5.10).

Доказательство. «$\Leftarrow$» Предположим, что q — корень многочлена (2.5.10). Докажем, что последовательность

$$a_1 = \alpha, \quad a_2 = \alpha q, \quad a_3 = \alpha q^2, \ldots$$

является решением уравнения (2.5.9). Член a_n этой последовательности равен αq^{n-1}, поэтому после подстановки чисел $a_n, a_{n+1}, \ldots, a_{n+k}$ в (2.5.9) получаем следующие равенства:

$$\begin{aligned} a_{n+k} + p_1 a_{n+k-1} + \ldots + p_k a_n = \\ = q^{n+k-1} + p_1 q^{n+k-2} + \ldots + p_k q^{n-1} = \\ = p_k q^{n-1}(q^k + p_1 q^{k-1} + \ldots + p_k) = 0\,. \end{aligned}$$

«$\Rightarrow$» Обозначим n-й член геометрической прогрессии через a_n. Так как $a_0 = \alpha$, $a_1 = \alpha q$ и т. д., то

$$a_{n+k} = \alpha q^{n+k},\ a_{n+k-1} = \alpha q^{n+k-1}, \ldots, a_n = \alpha q^n\,.$$

Полагая в (2.5.9)

$$x_{n+k} = a_{n+k},\ x_{n+k-1} = a_{n+k-1}, \ldots, x_n = a_n\,,$$

получим равенство

$$\alpha q^{n+k} + p_1 \alpha q^{n+k-1} + \ldots + p_k \alpha q^n = 0\,,$$

а из него после сокращения на $\alpha q^n \neq 0$ — равенство

$$q^k + p_1 q^{k-1} + \ldots + p_k = 0\,,$$

из которого и следует, что q — корень многочлена (2.5.10). □

2.5.15. *Если все корни характеристического многочлена* (2.5.10) *действительны и попарно различны, то существуют* k *попарно различных геометрических прогрессий, образующих базис возвратного уравнения* (2.5.9).

Доказательство. Пусть $\gamma_1, \ldots, \gamma_k$ — попарно различные корни характеристического многочлена (2.5.10). Заметим, что свободный член p_k этого многочлена отличен от нуля, так как иначе мы имели бы дело с возвратной последовательностью порядка $k-1$. Сказанное означает, что нуль не является корнем многочлена (2.5.10).

Составим k геометрических прогрессий:

$$\begin{array}{l} 1,\ \gamma_1,\ \gamma_1^2,\ \gamma_1^3,\ \ldots, \\ 1,\ \gamma_2,\ \gamma_2^2,\ \gamma_2^3,\ \ldots, \\ \ldots\ldots\ldots\ldots\ldots \\ 1,\ \gamma_k,\ \gamma_k^2,\ \gamma_k^3,\ \ldots. \end{array} \tag{2.5.11}$$

Докажем, что данная система последовательностей образует базис уравнения (2.5.9). Пусть $\{b_n\}$ — произвольная последовательность, удовлетворяющая этому уравнению. Нужно убедиться в том, что ее можно представить в виде линейной комбинации последовательностей из системы (2.5.11), т. е. в существовании чисел $\alpha_1, \ldots, \alpha_k$ таких, что для каждого $n \in \{0, 1, 2, \ldots\}$

$$b_n = \alpha_1 \gamma_1^n + \ldots + \alpha_k \gamma_k^n .$$

Для этого достаточно проверить, что равенство справедливо при $n = \overline{0, k-1}$, т. е. что

$$\begin{cases} \alpha_1 & + \ldots + \alpha_k & = b_0, \\ \alpha_1\gamma_1 & + \ldots + \alpha_k\gamma_k & = b_1, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \alpha_1\gamma_1^{k-1} + \ldots + \alpha_k\gamma_k^{k-1} & = b_{k-1} . \end{cases} \tag{2.5.12}$$

Мы получили неоднородную систему из k линейных уравнений с k неизвестными $\alpha_1, \ldots, \alpha_k$. Докажем, что она имеет решение. Так как число неизвестных в системе равно числу уравнений, достаточно показать, что ранг матрицы системы, составленной из коэффициентов при неизвестных, равен k, т. е. что эта матрица невырожденная. Это можно сделать различными способами. Например, можно вычислить определитель матрицы системы. Мы поступим иначе.

Рассмотрим однородную систему

$$\begin{cases} \alpha_1 & + \ldots + \alpha_k & = 0, \\ \alpha_1\gamma_1 & + \ldots + \alpha_k\gamma_k & = 0, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \alpha_1\gamma_1^{k-1} + \ldots + \alpha_k\gamma_k^{k-1} & = 0 . \end{cases} \tag{2.5.13}$$

Докажем, что она обладает единственным решением

$$\alpha_1 = \ldots = \alpha_k = 0\,.$$

Рассмотрим многочлен

$$Q(x) = \frac{(x-\gamma_2)(x-\gamma_3)\ldots(x-\gamma_k)}{(\gamma_1-\gamma_2)(\gamma_1-\gamma_3)\ldots(\gamma_1-\gamma_k)}\,.$$

Его степень равна $k-1$. Нетрудно заметить, что числа

$$\gamma_2, \gamma_3, \ldots, \gamma_k$$

являются его корнями и что

$$Q(\gamma_1) = \frac{(\gamma_1-\gamma_2)(\gamma_1-\gamma_3)\ldots(\gamma_1-\gamma_k)}{(\gamma_1-\gamma_2)(\gamma_1-\gamma_3)\ldots(\gamma_1-\gamma_k)} = 1\,.$$

Раскрывая скобки и приводя подобные члены, его можно представить в стандартном виде

$$Q(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_{k-1}x^{k-1}\,.$$

Уравнения системы (2.5.13) умножим на $\beta_0, \beta_1, \ldots, \beta_{k-1}$ соответственно и сложим. После приведения подобных членов получим уравнение

$$\begin{aligned}\alpha_1(\beta_0 + \beta_1\gamma_1 + \beta_2\gamma_1^2 + \ldots + \beta_{k-1}\gamma_1^{k-1}) + \\ + \alpha_2(\beta_0 + \beta_1\gamma_2 + \beta_2\gamma_2^2 + \ldots + \beta_{k-1}\gamma_2^{k-1}) + \ldots \\ \ldots + \alpha_k(\beta_0 + \beta_1\gamma_k + \beta_2\gamma_k^2 + \ldots + \beta_{k-1}\gamma_k^{k-1}) = 0\,,\end{aligned}$$

которое кратко можно записать в виде

$$\alpha_1 Q(\gamma_1) + \alpha_2 Q(\gamma_2) + \ldots + \alpha_k Q(\gamma_k) = 0\,.$$

Однако $\gamma_2, \gamma_3, \ldots, \gamma_k$ — корни многочлена $Q(x)$, а $Q(\gamma_1) = 1$, поэтому $\alpha_1 = 0$.

Аналогично доказывается, что $\alpha_2 = 0, \ldots, \alpha_k = 0$, т. е. что система (2.5.13) обладает единственным решением. Но тогда и система (2.5.12) обладает единственным решением.

Каждое решение уравнения (2.5.9) можно представить в виде линейной комбинации k выбранных прогрессий, и размерность пространства решений этого уравнения также равна k. Это означает, что ранг системы, в которой векторами служат эти прогрессии, равен k, т. е. что эта система линейно независима. □

2.5.16. *Если корни* $\gamma_1, \gamma_2, \ldots, \gamma_k$ *характеристического многочлена* (2.5.10) *действительны и попарно различны, то общее решение возвратного уравнения* (2.5.9) *может быть записано в виде*

$$a_n = \alpha_1\gamma_1^n + \alpha_2\gamma_2^n + \ldots + \alpha_k\gamma_k^n .$$

Доказательство. Утверждение непосредственно следует из 2.5.14 и 2.5.9. □

Следующее утверждение мы примем без доказательства, ввиду большого объема последнего.

2.5.17. *Предположим, что* $\gamma_1, \gamma_2, \ldots, \gamma_m$ *— все попарно различные действительные корни характеристического многочлена* (2.5.10) *и степень этого многочлена равна* k. *Если кратности этих корней равны соответственно* $s_1, \ldots, s_m$, *то общее решение возвратного уравнения* (2.5.9) *может быть записано в виде*

$$a_n = \sum_{i=1}^{m} (\alpha_{i1} + \alpha_{i2}n + \ldots + \alpha_{is_i}n^{s_i-1})\gamma_i^n . \qquad \square$$

Применим теперь полученные сведения о свойствах возвратных последовательностей для исследования одной из таких последовательностей.

2.5.18. *Последовательность, в которой каждый член начиная с третьего равен сумме двух предыдущих, а первые два члена равны единице, называется* ***последовательностью Фибоначчи****. Ее члены называются* ***числами Фибоначчи****.*

Согласно этому определению последовательность Фибоначчи выглядит следующим образом:

$$1, \quad 1, \quad 2, \quad 3, \quad 5, \quad 8, \ldots . \tag{2.5.14}$$

Она удовлетворяет рекуррентному соотношению

$$a_{n+2} = a_{n+1} + a_n . \tag{2.5.15}$$

Это соотношение позволяет последовательно находить все числа Фибоначчи. Оно также показывает, что последовательность Фибоначчи является возвратной последовательностью второго порядка.

Выведем формулу, позволяющую вычислять каждый член последовательности Фибоначчи по его номеру без обращения к предшествующим членам.

2.5.19. *Если* $\{a_i\}$ *— последовательность Фибоначчи, то*

$$a_n = \frac{\sqrt{5}+1}{2\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n + \frac{\sqrt{5}-1}{2\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n =$$

$$= \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^{n+1} - \left(\frac{1-\sqrt{5}}{2}\right)^{n+1}\right)$$

для любого $n \in \{0, 1, 2, \ldots\}$.

Доказательство. Последовательность Фибоначчи удовлетворяет соотношению (2.5.15), поэтому она является решением возвратного уравнения

$$a_{n+2} - a_{n+1} - a_n = 0\,. \tag{2.5.16}$$

Видим, что характеристический многочлен последовательности Фибоначчи равен

$$x^2 - x - 1\,.$$

Числа

$$\frac{1}{2} + \frac{1}{2}\sqrt{5}, \quad \frac{1}{2} - \frac{1}{2}\sqrt{5}$$

являются корнями этого многочлена.

Применяя утверждение 2.5.16, записываем общее решение уравнения (2.5.16):

$$a_n = \alpha_1\left(\frac{1+\sqrt{5}}{2}\right)^n + \alpha_2\left(\frac{1-\sqrt{5}}{2}\right)^n. \tag{2.5.17}$$

Поскольку нас интересует только одна конкретная последовательность, ищем коэффициенты α_1 и α_2, пользуясь тем, что значения начальных членов последовательности нам известны. Полагая в (2.5.17) $n = 0, 1$ и учитывая, что $a_0 = a_1 = 1$, из (2.5.17) получаем систему, состоящую из двух уравнений:

$$\begin{cases} \alpha_1 + \alpha_2 = 1, \\ \dfrac{1+\sqrt{5}}{2}\alpha_1 + \dfrac{1-\sqrt{5}}{2}\alpha_2 = 1\,. \end{cases}$$

Решая ее, находим α_1 и α_2:

$$\alpha_1 = \frac{\sqrt{5}+1}{2\sqrt{5}}, \quad \alpha_2 = \frac{\sqrt{5}-1}{2\sqrt{5}}.$$

Подставляя найденные значения в (2.5.15), получаем нужную формулу:

$$a_n = \frac{\sqrt{5}+1}{2\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n + \frac{\sqrt{5}-1}{2\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n =$$
$$= \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^{n+1} - \left(\frac{1-\sqrt{5}}{2}\right)^{n+1}\right). \qquad \square$$

Представляет интерес происхождение названия последовательности (2.5.14). В 1202 году итальянский математик Фибоначчи (Леонардо Пизанский) опубликовал книгу, в которой среди прочих сведений была приведена следующая задача.

Пара зрелых кроликов приносит раз в месяц приплод из двух крольчат, самца и самки. Зрелыми кролики становятся в возрасте одного месяца, а в возрасте двух месяцев они впервые дают потомство. Сколько будет пар зрелых кроликов через год, если сначала была одна зрелая пара?

Пусть $\{a_n\}$ — последовательность, i-й член которой равен числу пар зрелых кроликов через i месяцев. В начальный момент имеется только одна зрелая пара, поэтому $a_0 = 1$. Через месяц появится приплод, однако зрелыми будут только родители, следовательно, $a_1 = 1$. Так как через два месяца будем иметь две зрелые пары и одну пару крольчат, то $a_2 = 2$.

По условию через n месяцев имеется a_n пар зрелых кроликов. Также имеется сколько-то пар крольчат. Обозначим их количество через b_n. Каждая зрелая пара даст приплод, поэтому через $n+1$ месяц будем иметь a_n пар крольчат. Число зрелых пар нам известно из условия, оно равно a_{n+1}. В это количество входит и число b_n пар молодых кроликов, которые к этому моменту станут зрелыми. Еще через месяц станут зрелыми вновь народившиеся крольчата и мы будем иметь $a_n + a_{n+1}$ зрелых пар. Но число зрелых пар через $n + 2$ месяцев равно a_{n+2}. Получаем рекуррентное соотношение (2.5.15), показывающее, что мы имеем дело с последовательностью Фибоначчи.

Как уже говорилось, уравнение (2.5.15) позволяет поочередно вычислять числа Фибоначчи, и несложные подсчеты показывают, что $a_{12} = 233$. Это является ответом на задачу Фибоначчи. Разумеется, число a_{12} можно найти и по формуле из 2.5.19.

2.6. ПРОИЗВОДЯЩИЕ ФУНКЦИИ

2.6.1. *Числовым рядом называется пара таких последовательностей* $\{a_n\}$ *и* $\{s_n\}$ *действительных или комплексных чисел, что*

$$s_n = a_1 + a_2 + \ldots + a_n, \quad n = 1, 2, \ldots.$$

Ряд обозначается следующим образом:

$$a_1 + a_2 + \ldots + a_n + \ldots \quad \text{или} \quad \sum_{n=1}^{\infty} a_n \,.$$

Элементы последовательности $\{a_n\}$ *называются* ***членами*** *ряда, а элементы последовательности* $\{s_n\}$ *— его* ***частичными суммами****.*

2.6.2. *Ряд называется* ***сходящимся****, если последовательность его частичных сумм имеет конечный предел*

$$s = \lim_{n\to\infty} s_n \,,$$

называемый ***суммой ряда****. Если последовательность частичных сумм ряда не имеет конечного предела, то он называется* ***расходящимся****.*

Пример. Рассмотрим ряд

$$\frac{1}{1\cdot 2} + \frac{1}{2\cdot 3} + \ldots + \frac{1}{n(n+1)} + \ldots = \sum_{n=1}^{\infty} \frac{1}{n(n+1)} \,.$$

Так как

$$s_n = \frac{1}{1\cdot 2} + \frac{1}{2\cdot 3} + \ldots + \frac{1}{n(n+1)} =$$
$$= \left(1 - \frac{1}{2}\right) + \left(\frac{1}{2} - \frac{1}{3}\right) + \ldots + \left(\frac{1}{n} - \frac{1}{n+1}\right) = 1 - \frac{1}{n+1},$$

то $\lim\limits_{n\to\infty} s_n = 1$. Ряд сходится, и его сумма равна 1. □

2.6.3. *Ряд вида*

$$a_0 + a_1 x + a_2 x^2 + \ldots + a_k x^k + \ldots \tag{2.6.1}$$

называется ***степенным****.*

Полагая $x = c \in \mathbb{R}$ в (2.6.1), получим числовой ряд

$$a_0 + a_1 c + a_2 c^2 + \ldots + a_k c^k + \ldots, \tag{2.6.2}$$

сходящийся или расходящийся. Если для какого-то c ряд (2.6.2) окажется сходящимся, то существует такой интервал $(-r, r)$, $r > 0$, называемый **промежутком сходимости**, что ряд (2.6.1) сходится, как только $-r < x < r$ (это доказывается в курсе математического анализа).

Пример. Рассмотрим степенной ряд

$$1 + x + x^2 + x^3 + \ldots + x^n + \ldots = \sum_{k=0}^{\infty} x^k \,. \tag{2.6.3}$$

Он образован геометрической прогрессией, поэтому при $x \neq 1$

$$s_n = 1 + x + x^2 + x^3 + \ldots + x^{n-1} = \frac{1 - x^n}{1 - x} \,.$$

Если $|x| < 1$, то

$$\lim_{n\to\infty} x^n = 0, \quad \lim_{n\to\infty} s_n = \frac{1}{1 - x}$$

и ряд (2.6.3) сходится.

Так как при $x > 1$ справедливо $\lim\limits_{n\to\infty} x^n = \infty$ и величины $1 - x^n$ и $1 - x$ отрицательны, частичные суммы

$$s_n = \frac{1 - x^n}{1 - x}$$

положительны и $\lim\limits_{n\to\infty} s_n = \infty$. Согласно 2.6.2 ряд (2.6.3) расходящийся.

Если же $x < -1$, то число x^n положительное при n четном, отрицательное при n нечетном. К тому же $\lim\limits_{n\to\infty} |x|^n = \infty$. Это означает, что $\lim\limits_{n\to\infty} s_n$ не существует и ряд (2.6.3) расходящийся.

Итак, ряд (2.6.3) сходится только при $|x| < 1$, и в этом случае его сумма равна $\dfrac{1}{1 - x}$. □

2.6.4. *Производящей функцией последовательности* $\{a_n\}$ *называется сумма*

$$F(z) = \sum_{n=0}^{\infty} a_n z^n \tag{2.6.4}$$

степенного ряда

$$a_0 + a_1 z + a_2 z^2 + \ldots + a_k z^k + \ldots . \tag{2.6.5}$$

Для конечной последовательности $a_0, a_1, \ldots, a_n$ *функция*

$$F(z) = \sum_{i=0}^{n} a_i z^i$$

также называется ***производящей***.

Примеры 1. Для конечной последовательности $a_k = C_n^k$, $k = \overline{0,n}$, ввиду 2.2.7 производящей является такая функция:

$$F(z) = \sum_{k=0}^{n} C_n^k z^k = (1+z)^n . \tag{2.6.6}$$

2. Рассмотрим последовательность $a^0, a^1, a^2, \ldots$. При $|az| < 1$ сходится ряд

$$\sum_{n=0}^{\infty} a^n z^n$$

и его сумма равна $\dfrac{1}{1-az}$. Это и есть производящая функция исходной последовательности. □

Производящие функции можно использовать для вывода комбинаторных тождеств.

3. Полагая в (2.6.6) $z = -1$, получаем формулу

$$0 = C_n^0 - C_n^1 + C_n^2 - C_n^3 + \ldots + (-1)^n C_n^n .$$

4. Перемножим соответствующие части равенств

$$(1+z)^n = C_n^0 + C_n^1 z + C_n^2 z^2 + \ldots + C_n^n z^n ,$$

$$(1+z)^m = C_m^0 + C_m^1 z + C_m^2 z^2 + \ldots + C_m^m z^m .$$

Слева получим выражение

$$(1+z)^{n+m} .$$

Возведем $1+z$ в степень $n+m$, воспользовавшись формулой бинома Ньютона. Перемножив правые части, приведем подобные члены. После таких преобразований и в правой части, и в левой будут записаны многочлены от переменной z. Приравнивая коэффициенты при одинаковых степенях этой переменной и считая, что $C_p^q = 0$ при $q < 0$, получим формулы

$$C_{n+m}^s = C_n^0 C_m^s + C_n^1 C_m^{s-1} + \ldots + C_n^k C_m^{s-k} + \ldots + C_n^n C_m^{s-n} . \qquad \square$$

Важной является также такая задача. Задана производящая функция. Нужно разложить ее в степенной ряд, т. е. определить, может ли она в некотором интервале быть суммой степенного ряда, а в случае положительного ответа требуется найти коэффициенты этого ряда. Следующая теорема доказывается в курсе математического анализа.

2.6.5. *Если ряд* (2.6.5) *сходится в интервале* $(-r, r)$, $r > 0$, *то его сумма* (2.6.4) *имеет в любой точке этого интервала производные всех порядков, причем функция* $F^{(n)}(z)$ (*производная порядка* n *функции* $F(z)$) *для любого* $n = 1, 2, \ldots$ *является суммой степенного ряда, получаемого* n*-кратным почленным дифференцированием ряда* (2.6.5)*: при* $-r < z < r$

$$F^{(n)}(z) = \sum_{k=n}^{\infty} k(k-1)\ldots(k-n+1)a_k z^{k-n}\,. \tag{2.6.7}$$

Видим, что пытаться разложить функцию $F(z)$ в степенной ряд имеет смысл только тогда, когда она имеет производные всех порядков в каждой точке некоторого интервала $(-r, r)$, $r > 0$. Если функция $F(z)$ удовлетворяет этому требованию и разложена в степенной ряд (2.6.5), то для любого $n \in \mathbb{N}$ выполняется также равенство (2.6.7). Полагая $z = 0$ и учитывая, что $0^0 = 1$, убеждаемся в том, что

$$F^{(n)}(0) = n!a_n, \quad a_n = \frac{F^{(n)}(0)}{n!} \quad (n = 0, 1, 2, \ldots)\,. \tag{2.6.8}$$

Делаем следующий вывод.

2.6.6. *Коэффициенты* a_n *степенного ряда, суммой которого служит функция* $F(z)$, *однозначно определяются через эту функцию с помощью формул* (2.6.8).

Пример. Пусть $F(z) = \dfrac{1}{1-2z}$. Последовательно дифференцируя, убеждаемся в том, что

$$F^{(n)}(z) = (-1)(-2)\ldots(-n)(-2)^n(1-2z)^{-1-n}\,.$$

Из (2.6.8) следует, что

$$a_0 = F(0) = 1, \quad a_1 = \frac{F'(0)}{1!} = 2\,,$$

$$a_2 = \frac{F''(0)}{2!} = 2^2, \ldots, \quad a_n = \frac{F^n(0)}{n!} = 2^n$$

и потому

$$F(z) = 1 + 2z + 2^2z^2 + \ldots + 2^k z^k + \ldots\,. \qquad \square$$

Глава 3

ГРАФЫ

3.1. ВИДЫ ГРАФОВ

Термин «граф» происходит от греческого *grapho* — «пишу». Он возник тогда, когда графы изображали рисунками. Пример такого изображения дает рис. 3.1. На нем можно увидеть основные элементы графа: точки, называемые вершинами, и линии, их соединяющие, называемые ребрами. В данном примере граф — это множество точек на плоскости, некоторые из этих точек соединены кривыми. Однако мы не примем это описание за определение графа по следующим причинам. Граф может иметь очень много вершин. Трудно представить себе рисунок, на котором изображен граф, имеющий 250 000 вершин и 50 000 ребер. Более того, математики изучают также графы с бесконечным числом вершин. Вследствие этого примем следующее абстрактное определение графа.

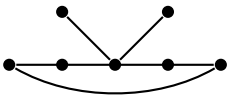


Рис. 3.1

3.1.1. *Система, состоящая из непустого множества V и бинарного отношения E, определенного на V, называется* ***графом****. Эту систему будем обозначать через $G(V, E)$. Элементы множества V называются* ***вершинами графа*** *$G(V, E)$, а элементы отношения E — его* ***ребрами****.*

3.1.2. *Вершины и ребра графа называются его* ***элементами****. Граф, содержащий конечное число элементов, называется* ***конечным****. Число вершин конечного графа G называется его* ***порядком*** *и обозначается через $|G|$. Граф порядка n, имеющий m ребер, называется (n, m)-****графом****.*

Когда порядок графа невелик, бывает полезно вернуться к изображению графа в виде рисунка, состоящего из точек, соответствующих

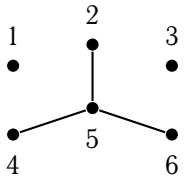


Рис. 3.2

его вершинам, и отрезков или кривых, соединяющих пары точек тогда и только тогда, когда эти точки являются концами одного ребра, принадлежащего графу.

Пример. На рис. 3.2 изображен граф порядка 6 с тремя ребрами, т. е. $(6,3)$-граф. Он образован множеством $\{1,2,3,4,5,6\}$ вершин и множеством ребер $\{(4,5),(2,5),(5,6)\}$. □

3.1.3. *Если пара (u,v) является ребром графа, то вершины u и v называются* ***концами*** *этого* ***ребра****, а про ребро говорят, что оно* ***соединяет вершины*** *u и v.*

3.1.4. *Ребро с совпадающими концами называется* ***петлей****. Граф без петель называется* ***простым****.*

Далее в этой главе будут рассматриваться только конечные графы. Если не сказано другого, под словом «граф» будет подразумеваться простой граф.

В различных математических конструкциях, кроме простых графов и графов с петлями, встречаются и иные виды графов. Например, иногда необходимо, чтобы две вершины соединяло несколько ребер. Определение 3.1.1 в этом случае не годится, так как все такие ребра представляют собой совпадающие пары вершин, а в множестве E все элементы различны. Изменим определение 3.1.1 следующим образом.

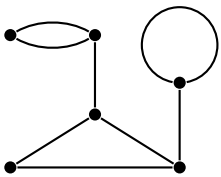


Рис. 3.3

3.1.5. ***Псевдографом*** *называется система, состоящая из непустого множества V, элементы которого называются* ***вершинами****, и совокупности E пар вершин, называемых* ***ребрами****. Совокупность E может содержать одинаковые пары, а также пары с одинаковыми элементами, принадлежащими V.*

Различие между определениями 3.1.1 и 3.1.5 заключается в том, что в первом случае E названо множеством, а во втором — совокупностью. В множестве E все пары вершин разные, а в совокупности E могут встречаться одинаковые пары вершин. Таким образом, *в псевдографе две вершины могут соединяться несколькими ребрами и, кроме того, могут встречаться петли*. По-гречески *pseudos* — ложь, поэтому слово псевдограф означает *мнимый граф*, или *лжеграф*.

Пример. На рис. 3.3 изображен псевдограф, в котором две вершины соединены двумя ребрами и есть петля. □

3.1.6. *Псевдограф без петель называется* ***мультиграфом*** *(или* ***графом с кратными ребрами****).*

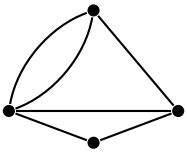


Рис. 3.4

Другими словами, *в мультиграфе нет петель, а некоторые вершины могут быть соединены несколькими ребрами*. По-латински *multum* — много.

Пример. На рис. 3.4 изображен мультиграф, у которого две вершины соединены двумя ребрами. □

3.1.7. *Граф $G(V, E)$ называется* ***ориентированным****, или* ***орграфом****, если принадлежащие множеству E пары вершин являются упорядоченными. Ориентированные ребра называются* ***дугами****. Ориентированность пары вершин, образующих дугу, означает, что одна из них считается* ***началом****, а другая —* ***концом*** *дуги. На рисунках направления от начала к концу дуг орграфа указываются стрелками.*

Пример. Граф, избраженный на рис. 3.5, является ориентированным. □

Иногда рассматриваются и ***смешанные*** графы, имеющие как дуги, так и неориентированные ребра. В таких графах неориентированное ребро (a, b) заменяет две дуги (a, b) и (b, a). Обычно направление дуги указывает, в какую сторону по ней возможно двигаться. Так как ребро не имеет ориентации, двигаться по нему можно в обе стороны.

Пример. На рис. 3.6 приведен пример смешанного графа. □

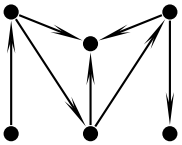


Рис. 3.5

Если порядок графа невелик, то его можно нарисовать. Поскольку изображающие вершины графа точки берутся произвольно, рисунки, изображающие один и тот же граф, могут быть совершенно непохожими. Как узнать, совпадают ли два графа, заданных двумя рисунками? Аналогичная проблема возникает и при других способах задания графа, о которых будет сказано позднее. Решение указанной проблемы стандартное: если можно взаимно однозначно отобразить множество вершин одного графа на множество вершин другого графа так, что две вершины одного графа будут являться концами одного ребра в точности тогда, когда соответствующие им вершины другого графа также соединены ребром, то перед нами две копии одного объекта.

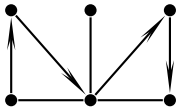
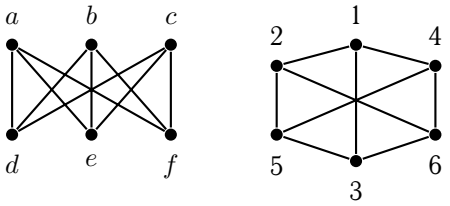


Рис. 3.6

3.1.8. *Графы $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ называются* ***изоморфными****, если существует такая функция φ, взаимно однозначно отображающая множество вершин V_1 на множество вершин V_2, что пара (u, v) является ребром графа G_1 тогда и только тогда, когда пара $(\varphi(u), \varphi(v))$ является ребром графа G_2.*

Слово «изоморфные» образовано из греческих слов *isos* — равный, одинаковый, подобный — и *morphe* — форма.



Граф A. Граф B.

Рис. 3.7

Пример. Графы A и B, изображенные на рис. 3.7, изоморфны. Действительно, пусть

$$\varphi(a)=1,\quad \varphi(b)=5,\quad \varphi(c)=6,\quad \varphi(d)=2,\quad \varphi(e)=3,\quad \varphi(f)=4,$$

тогда тройкам ребер

$$(a,d),\ (a,e),\ (a,f);\quad (b,d),\ (b,e),\ (b,f);\quad (c,d),\ (c,e),\ (c,f)$$

графа A соответствуют тройки

$$(\varphi(a),\varphi(d))=(1,2),\quad (\varphi(a),\varphi(e))=(1,3),\quad (\varphi(a),\varphi(f))=(1,4);$$
$$(\varphi(b),\varphi(d))=(5,2),\quad (\varphi(b),\varphi(e))=(5,3),\quad (\varphi(b),\varphi(f))=(5,4);$$
$$(\varphi(c),\varphi(d))=(6,2),\quad (\varphi(c),\varphi(e))=(6,3),\quad (\varphi(c),\varphi(f))=(6,4)$$

ребер графа B. □

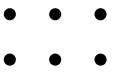


Рис. 3.8

Как известно, два векторных пространства над одним полем изоморфны тогда и только тогда, когда их размерности совпадают. В этом случае имеется очень просто формулируемый и легко проверяемый критерий изоморфности.

Для графов подобный критерий неизвестен, и решить проблему, являются ли два графа изоморфными, часто бывает нелегко.

Среди всех графов наиболее простое строение имеют пустые и полные графы.

3.1.9. *Граф называется* ***полным****, если любые две его вершины соединены ребром. Граф, в котором нет ребер, называется* ***пустым****.*

Пустой граф состоит из одних вершин. На рис. 3.8 изображен пустой граф с шестью вершинами. У полного графа каждая вершина соединена ребрами со всеми остальными вершинами. На рис. 3.9 изображен полный граф четвертого порядка.

3.1.10. ***Двудольным*** *называется граф, множество вершин которого можно так разбить на два подмножества, называемые* ***долями****, что у каждого ребра концы будут принадлежать разным долям.*

Пример. Граф на рис. 3.10*а* двудольный, а на рис. 3.10*б* — полный двудольный. □

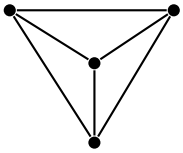


Рис. 3.9

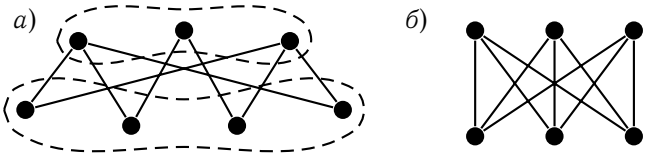


Рис. 3.10

3.2. МАТРИЦЫ СМЕЖНОСТИ И ИНЦИДЕНТНОСТИ

3.2.1. *Две вершины графа называются* ***смежными****, если они являются концами какого-нибудь ребра этого графа. Два ребра называются* ***смежными****, если они имеют общий конец. Вершина* v *и ребро* r *называются* ***инцидентными****, если* v *является концом ребра* r*, в противном случае они называются* ***неинцидентными****.*

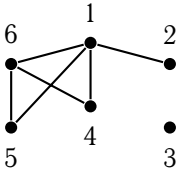


Рис. 3.11

Примеры. Рассмотрим граф, изображенный на рис. 3.11. Его вершины 1 и 6 смежные, так как являются концами ребра $(1,6)$. Вершины 2 и 5 не являются смежными. Вершина 4 и ребро $(4,6)$ инцидентны, вершина 4 и ребро $(5,6)$ неинцидентны. Ребра $(6,5)$ и $(5,1)$ смежны, ребра $(6,4)$ и $(1,2)$ несмежны. □

3.2.2. *Число ребер графа, инцидентных некоторой вершине, называется* ***степенью этой вершины****. Вершина графа, имеющая степень* 0*, называется* ***изолированной****, а вершина, имеющая степень* 1*, —* ***концевой****, или* ***висячей****.*

Пример. Рассмотрим граф, изображенный на рис. 3.11. Его вершины 4 и 5 имеют степень 2, вершина 6 — степень 3, вершина 1 — степень 4. Вершина 2 является висячей, так как имеет степень 1. Вершина 3 изолированная. □

Очевидно, в пустом графе все вершины изолированные, а в полном степень каждой вершины на единицу меньше порядка графа.

3.2.3. *Сумма степеней всех вершин графа равна удвоенному числу ребер.*

Доказательство. Согласно 3.1.2 речь идет о числе ребер. При подсчете указанной суммы каждое ребро считается дважды: при определении степени одного конца и при определении степени другого конца. □

Чтобы задать граф, надо каким-то способом описать множество V его вершин, множество E его ребер, а также указать, какие вершины и ребра являются инцидентными, то есть задать отношение инцидентности. Весьма простым и наглядным является задание графа при помощи рисунка, однако этот метод годится только для простейших случаев. Можно однозначно задать любой конечный граф, если воспользоваться следующими определениями.

3.2.4. *Если задана функция, отображающая множество вершин или множество ребер графа в некоторое множество, элементы которого называются* ***метками****, то граф называется* ***помеченным*** *(или* ***нагруженным****,* ***взвешенным****).*

Иными словами, ребрам или вершинам помеченного графа сопоставлены некоторые метки. Обычно метками являются числа или буквы.

Пример. У графа на рис. 3.12 помечены и вершины, и ребра. □

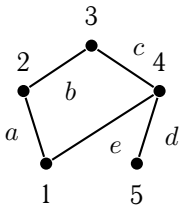


Рис. 3.12

3.2.5. *Матрицей смежности графа (орграфа)* G *порядка* n*, вершинам которого присвоены метки* $1, 2, \ldots, n$*, называется квадратная матрица того же порядка, у которой для любых* $i, j \in \{1, 2, \ldots, n\}$ *элемент, находящийся в* i*-й строке и* j*-м столбце, равен единице, если вершины графа* G *с метками* i *и* j *соединены ребром (дугой с началом в вершине* i*), равен нулю в противном случае.*

Из этого определения следует, что матрица смежности графа обладает следующими свойствами:

а) элементы ее главной диагонали всегда равны нулю;

б) матрица является симметричной;

в) число единиц в i-й строке равно степени вершины с пометкой i;

г) столбец и строка, соответствующие изолированной вершине, не содержат единиц.

Зная эту матрицу, легко составить список ребер графа, причем достаточно знать элементы матрицы по одну сторону от главной диагонали.

Примеры. Матрица смежности графа, изображенного на рис. 3.12, выглядит следующим образом:

$$\begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}. \qquad \square$$

Матрица смежности орграфа, не являющегося мультиграфом, обычно не симметрична, так как при ее составлении вершины орграфа играют различные роли. Элементы ее главной диагонали всегда равны нулю. Столбец и строка, соответствующие изолированной вершине, не содержат единиц.

У орграфа на рис. 3.13*б* матрица смежности выглядит так:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}. \qquad \square$$

3.2.6. *В матрице смежности псевдографа число, находящееся на пересечении* i*-й строки и* j*-го столбца, совпадает с числом ребер, соединяющих вершины* i *и* j*, при этом каждая петля считается двумя ребрами.*

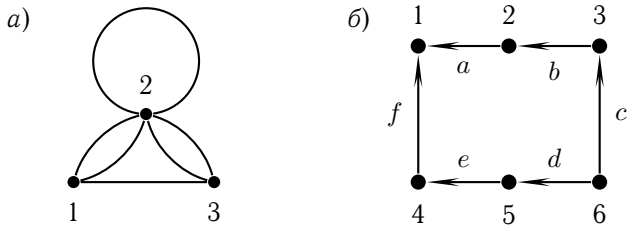


Рис. 3.13

Число на пересечении i-й строки и j-го столбца матрицы смежности псевдоорграфа равно числу дуг, исходящих из вершины i и оканчивающихся в вершине j.

Матрица смежности псевдографа обладает следующими свойствами:

а) матрица является симметричной;

б) элементы ее главной диагонали четные;

б) если на главной диагонали стоит число $2k$ и $k \neq 0$, то у соответствующей вершины имеется k петель.

Пример. Псевдограф, изображенный на рис. 3.13*а*, имеет следующую матрицу смежности:

$$\begin{pmatrix} 0 & 2 & 1 \\ 2 & 2 & 2 \\ 1 & 2 & 0 \end{pmatrix}. \qquad \square$$

Если помеченными являются как вершины, так и ребра графа, то его можно задавать при помощи матрицы инцидентности.

3.2.7. ***Матрица инцидентности помеченного (m,n)-графа*** *имеет m строк и n столбцов. Ее элемент, расположенный в i-й строке и j-м столбце, равен 1, если вершина i инцидентна ребру j, равен 0 в остальных случаях.*

Пример. Матрица инцидентности графа, изображенного на рис. 3.14, выглядит следующим образом:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}. \qquad \square$$

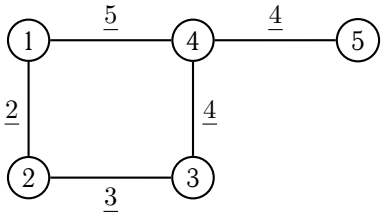


Рис. 3.14

3.2.8. *Матрица инцидентности* (m,n)*-орграфа имеет* m *строк и* n *столбцов. Ее элемент, расположенный в* i*-й строке и* j*-м столбце, равен* 1*, если вершина* i *является началом дуги* j*, равен* -1*, если вершина* i *является концом дуги* j*, равен 0 в остальных случаях.*

Пример. Матрица

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & -1 & -1 & 0 & 0 \end{pmatrix}$$

является матрицей инцидентности орграфа, изображенного на рис. 3.13, если считать, что ребра помечены в алфавитном порядке. □

Иногда матрицу инцидентности задают иначе: ее элемент α_{ij} равен -1, если вершина i является началом дуги j, равен 1, если вершина i является концом дуги j, равен 0 в остальных случаях.

3.3. ОПЕРАЦИИ С ГРАФАМИ

Имея один или несколько графов, можно из них строить новые графы.

3.3.1. *Граф* G_1 *называется* ***подграфом*** *(или* ***частью) графа*** G_2*, если каждая вершина графа* G_1 *является вершиной графа* G_2 *и каждое ребро графа* G_1 *является ребром графа* G_2*. Про подграф* G_1 *можно также сказать, что он* ***содержится в графе*** G_2*.*

Замечание. Некоторые авторы подграфом графа считают лишь такую его часть, в которой две вершины соединены ребром тогда и только тогда, когда они соединены ребром в графе. Иными словами, если

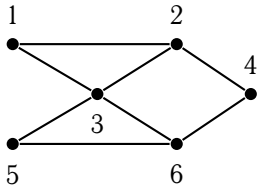


Рис. 3.15
Граф G_1

в таком подграфе отсутствует ребро, имеющееся в графе, то должны отсутствовать и концы этого ребра.

Примеры. Граф G_2 является подграфом графа G_1, а граф G_3 содержится как в G_1, так и в G_2 (рис. 3.16). □

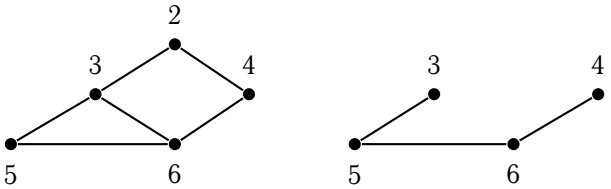


Рис. 3.16
Графы G_2 (слева) и G_3 (справа)

3.3.2. *Пусть v — вершина графа $G(V, E)$. Графом, полученным из G **удалением вершины** v, называется граф $G_1(V_1, E_1)$, у которого $V_1 = V \setminus \{v\}$, а множество E_1 получается из E удалением всех ребер, инцидентных вершине v. Обозначение: $G_1 = G - v$.*

Пример. Граф G_4 (рис. 3.17) получен из графа G_1 (рис. 3.15) удалением вершины 3. □

3.3.3. *Граф $G_1(V_1, E_1)$ называется **подграфом графа** $G(V, E)$, **полученным удалением ребра** $r \in E$, если $V_1 = V$ и $E_1 = E \setminus \{r\}$. Обозначение: $G_1 = G - r$.*

Пример. Граф G_5 (рис. 3.17) получен из графа G_1 (рис. 3.15) удалением ребра $(2, 3)$. □

3.3.4. *Пусть X — множество каких-либо элементов графа G. Через $G - X$ обозначается граф, полученный из G удалением всех ребер и вершин, принадлежащих X.*

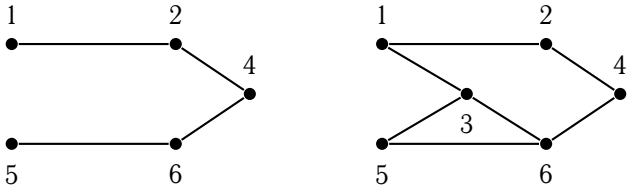


Рис. 3.17
Графы G_4 (слева) и G_5 (справа)

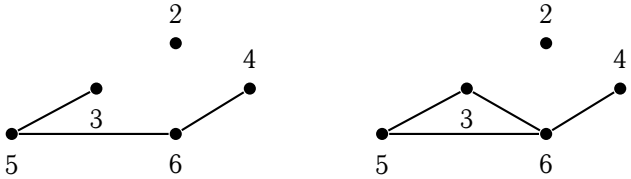


Рис. 3.18
Графы G_6 (слева) и G_7 (справа)

Пример. Граф G_3 (рис. 3.16) получен из графа G_1 (рис. 3.15) удалением вершин 1 и 2 и ребра $(3,6)$, поэтому $G_3 = G_1 - X$, где $X = \{1,2,(3,6)\}$. □

Теперь рассмотрим операции, обратные введенным в 3.3.2–3.3.4.

3.3.5. *При* ***добавлении в граф*** $G(V,E)$ ***вершины*** $v \notin V$ *получается граф* $G_1(V_1,E_1)$*, у которого* $V_1 = V \cup \{v\}$ *и* $E_1 = E$*. Обозначение:* $G_1 = G + v$.

Пример. Граф G_6 (рис. 3.18) получен добавлением вершины 2 из графа G_3 (рис. 3.16). □

3.3.6. *Пусть* r *— ребро, не входящее в множество ребер графа* $G(V,E)$*, однако концы этого ребра принадлежат графу* G*. Говорят, что* ***граф*** $G_1(V_1,E_1)$ ***получен из графа*** G ***добавлением ребра*** r*, если* $V_1 = V$ *и* $E_1 = E \cup \{r\}$*. Обозначение:* $G_1 = G + r$.

Пример. Граф G_7 (рис. 3.18) получен из графа G_6 (там же) добавлением ребра $(3,6)$. □

3.3.7. *Пусть* X *— множество каких-либо элементов графа* G_1 *и* G_2 *— подграф графа* G_1*, не содержащий эти элементы. Через* $G + X$ *обозначается граф, полученный из* G *добавлением всех ребер и вершин, принадлежащих* X.

Пример. Пусть $X = \{2, (3,2), (2,4), (3,6)\}$ — множество, содержащее вершину и 3 ребра графа G_2 (рис. 3.16). Если G_3 — граф, изображенный там же, то $G_2 = G_3 + X$. □

3.3.8. *Множество всех вершин графа, смежных с вершиной v, называется* ***окружением*** *этой вершины и обозначается через $N(v)$.*

Пример. Окружение вершины 3 графа G_1 на рис. 3.15 состоит из вершин 1, 2, 5, 6, поэтому $N(3) = \{1,\ 2,\ 5,\ 6\}$. □

3.3.9. *Пусть v — одна из вершин графа G. Следующее преобразование графа G называется* ***расщеплением вершины*** *v:*

1) *окружение $N(v)$ вершины v произвольным способом разбивается на два подмножества N_1 и N_2;*

2) *из графа G удаляется вершина v;*

3) *к полученному графу добавляются вершины v_1 и v_2 и ребро (v_1, v_2);*

4) *вершины из множества N_1 соединяются ребрами с вершиной v_1, а вершины из N_2 — с v_2.*

Из этого определения следует, что расщеплением одной и той же вершины из одного графа можно, вообще говоря, получить несколько разных новых графов.

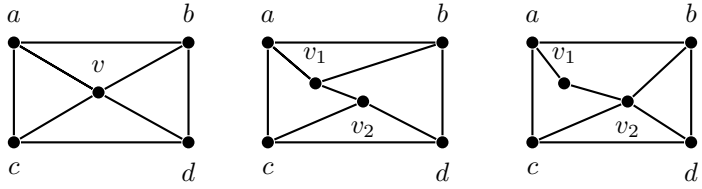


Рис. 3.19
Графы G_8 (слева), G_9 (в центре), G_{10} (справа)

Пример. Графы G_9 и G_{10} получены из графа G_8 расщеплением вершины v (рис. 3.19). □

Следующая операция является обратной к операции, введенной в 3.3.9.

3.3.10. *Пусть u и v — две вершины графа G, N_1 и N_2 — окружения этих вершин. Граф F называется* ***графом, полученным из G отождествлением вершин*** *u и v, если он получается из графа $H = G - \{u, v\}$*

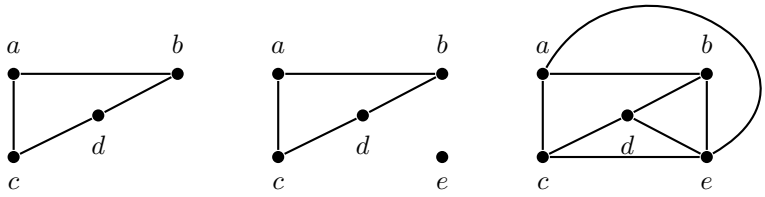


Рис. 3.20
Графы G_{11} (слева), G_{12} (в центре), G_{13} (справа)

добавлением новой вершины и соединением этой вершины ребрами с каждой принадлежащей графу H вершиной из множества $(N_1 \cup N_2) \setminus \{u, v\}$.

Пример. Изучим графы на рис. 3.19 и 3.20. Отождествим вершины v_1 и v_2 графа G_{10}. Окружение вершины v_1 состоит из вершин a и v_2. Окружение вершины v_2 образовано вершинами b, c, d, v_1. После удаления из G_{10} вершин v_1 и v_2 получается граф G_{11}. Добавляя вершину e, получаем граф G_{12}. Соединяем ребрами вершину e с вершинами a, b, c, d, входившими в множество $N(v_1) \cup N(v_3) \setminus \{v_1, v_2\}$, и получаем граф G_{13}. □

3.3.11. *Про граф, полученный из графа G отождествлением смежных вершин u и v, говорят, что он получен из G **стягиванием ребра** (u, v). Граф G называется **стягиваемым к графу** G_1, если G_1 можно получить из G с помощью некоторой последовательности стягиваний ребер.*

Пример. Стягиванием ребра (v_1, v_2) графа G_9 получается граф G_8. При стягивании ребра (v_1, v_2) графа G_{10} также возникает граф G_8 (рис. 3.19). □

3.3.12. *Если множество ребер подграфа $G_1(V_1, E_1)$ графа G совпадает с множеством всех тех ребер графа G, оба конца которых принадлежат V_1, то G_1 называется **подграфом, порожденным множеством вершин** V_1, и обозначается через $G(V_1)$.*

Иначе говоря, две вершины подграфа G_1 соединены ребром тогда и только тогда, когда они соединены ребром в графе G.

Пример. Подграф G_2 (рис. 3.16) графа G_1 (рис. 3.15) порожден множеством $\{2, 3, 4, 5, 6\}$ своих вершин. Граф G_3 (рис. 3.16) также является подграфом графа G_1, но он не порожден множеством своих вершин, так как в нем отсутствует ребро $(3, 6)$. □

3.3.13. *Если множество вершин подграфа $G_1(V_1, E_1)$ графа G совпадает с множеством концов всех его ребер, то G_1 называется* ***подграфом, порожденным множеством ребер*** *E_1.*

Из этого определения следует, что подграф G_1 можно получить так: в множестве ребер графа G выбираем подмножество E_1, а затем из концов этих ребер формируем множество вершин V_1.

Пример. Подграфы G_2 и G_3 (рис. 3.16) графа G_1 (рис. 3.15) порождены множествами своих ребер, а подграф G_6 (рис. 3.18) множеством ребер $\{(5,3),(5,6),(6,4)\}$ не порождается. □

3.3.14. *Граф H называется* ***дополнением графа*** *G, если множества вершин этих графов совпадают и две вершины графа H соединены ребром тогда и только тогда, когда они не являются смежными в графе G. Дополнение графа G обозначается через $\overline{G}$.*

Другими словами, чтобы получить дополнение графа G, следует удалить из G все ребра и соединить ребрами те вершины, которые в G не были смежными.

Пример. Граф G_{14} (рис. 3.21) является дополнением графа G_{12} (рис. 3.20). □

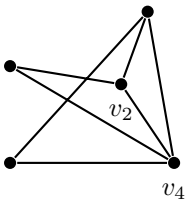


Рис. 3.21
Граф G_{14}

3.3.15. *Граф $G(V, E)$ называется* ***объединением*** *двух* ***графов*** *$G_1(V_1, E_1)$ и $G_2(V_2, E_2)$, если*

$$V = V_1 \cup V_2, \quad E = E_1 \cup E_2.$$

Объединение двух графов называется ***дизъюнктным****, если у этих графов нет общих вершин.*

Иногда под объединением графов подразумевают именно дизъюнктное объединение. Термин «дизъюнктный» означает *раздельный* и происходит от латинского слова *disjunctivus*.

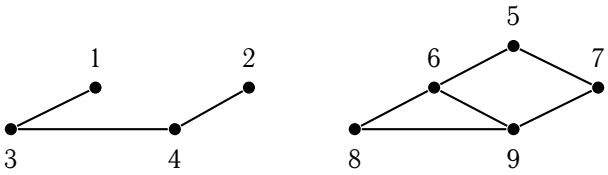


Рис. 3.22
Графы G_{15} (слева) и G_{16} (справа)

Пример. Объединяя графы G_{15} и G_{17}, получим граф G_{18}, а при объединении графов G_{15} и G_{16} возникает граф G_{19}. Это объединение дизъюнктное. □

3.3.16. *Предположим, что графы $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ не имеют общих вершин. Говорят, что граф $G(V, E)$ получен* ***соединением графов*** *G_1 и G_2, если $V = V_1 \cup V_2$, а E состоит из всех ребер, принадлежащих графам G_1 и G_2, и всех ребер, соединяющих каждую вершину графа G_1 с каждой вершиной графа G_2.*

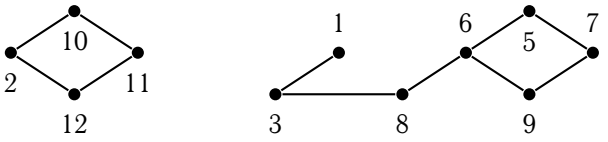


Рис. 3.23
Графы G_{17} (слева) и G_{18} (справа)

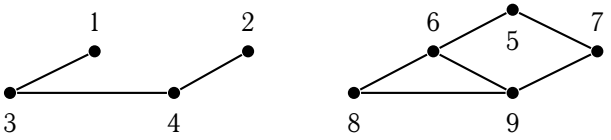


Рис. 3.24
Граф G_{19}

Пример. При соединении графов G_{20} и G_{21} получается граф G_{22} (рис. 3.25). □

3.3.17. *Граф $G(V, E)$ называется* ***декартовым произведением графов*** *$G_1(V_1, E_1)$ и $G_2(V_2, E_2)$, если $V = V_1 \times V_2$ и вершины (u_1, u_2) и (v_1, v_2) смежны в G тогда и только тогда, когда либо $u_1 = v_1$ и ребро с концами u_2, v_2 принадлежит E_2, либо $u_2 = v_2$ и ребро с концами u_1, v_1 принадлежит E_1. Обозначение: $G_1 \square G_2$.*

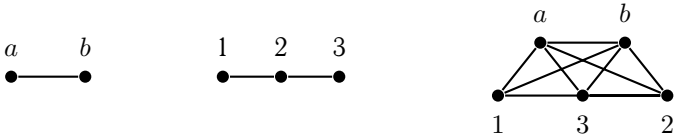


Рис. 3.25
Графы G_{20} (слева), G_{21} (в центре), G_{22} (справа)

Замечание. Существуют и другие определения и обозначения декартова произведения графов.

Пример. Нарисуем граф H, представимый в виде декартова произведения графов $G_{20}(V_1, E_1)$ и $G_{21}(V_2, E_2)$ (рис. 3.25). Декартовым произведением множеств $V_1 = \{a, b\}$ и $V_2 = \{1, 2, 3\}$ вершин этих графов является множество пар

$$V = \{(a,1), (a,2), (a,3), (b,1), (b,2), (b,3)\}\,.$$

Элементы этого множества — вершины графа H. Ребрами графа H являются пары элементов множества V, т. е. пары пар. Чтобы несколько упростить записи, пометим вершины графа H следующим образом:

$$(a,1) = a_1\,, \quad (a,2) = a_2\,, \quad (a,3) = a_3\,,$$
$$(b,1) = b_1\,, \quad (b,2) = b_2\,, \quad (b,3) = b_3\,.$$

В парах $(a, 1)$ и $(b, 1)$ вторые элементы совпадают, а первые являются вершинами графа G_{20} и соединены в нем ребром, поэтому вершины графа H с метками a_1 и b_1 тоже соединены ребром. По аналогичным причинам пары (a_2, b_2) и (a_3, b_3) также являются ребрами графа H.

В парах $(a, 1)$ и $(a, 2)$ совпадают первые элементы, а вторые являются вершинами графа G_{20} и соединены в нем ребром. Это означает, что (a_1, a_2) — ребро графа H. Ребрами являются также пары

$$(a_2, a_3)\,, \quad (b_1, b_2)\,, \quad (b_2, b_3)\,.$$

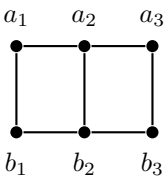


Рис. 3.26
Граф H

Нетрудно убедиться в том, что, за исключением упомянутых, ни одна пара вершин графа H не соединена ребром, так как не удовлетворяет условиям определения 3.3.17. Получаем граф, изображенный на рис. 3.25 справа. □

3.4. МАРШРУТЫ

3.4.1. *При $n \geqslant 2$ **маршрутом** в графе, **соединяющим вершины** v_1 и v_n, называется последовательность*

$$v_1, r_1, v_2, r_2, v_3, \ldots, v_{n-1}, r_{n-1}, v_n,$$

*в которой вершины и ребра чередуются и для каждого $i = \overline{1, n-1}$ ребро r_i инцидентно вершинам v_i и v_{i+1}. Последовательность, состоящая из одной вершины, также называется **маршрутом**.*

3.4.2. *Число ребер в маршруте называется **длиной маршрута**.*

3.4.3. *Маршрут, в котором все ребра попарно различны, называется **цепью**.*

3.4.4. *Цепь, в которой попарно различны все вершины, называется **простой**.*

Пример. Последовательность

$$a, 3, d, 4, b, 5, e, 6, c, 7, f, 9, e, 5, b$$

является маршрутом длины 7, соединяющим в графе Q вершины a и b (рис. 3.27). Этот маршрут не является цепью. В графе Q на рис. 3.27 цепями являются, например, маршрут $a, 1, b, 5, e, 6, c, 7, f, 9, e, 8, d$ и маршрут $a, 3, d, 4, b, 5, e, 6, c, 7, f$. Последняя цепь простая. □

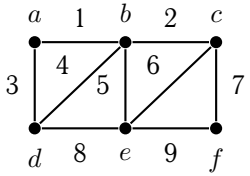


Рис. 3.27
Граф Q

Очевидно, любой маршрут, в котором попарно различны все вершины, является простой цепью.

3.4.5. *Маршрут, в котором первая и последняя вершины совпадают, называется **замкнутым**. Замкнутая цепь называется **циклом**.*

Термин «цикл» образован от греческого слова *kyklos* — круг.

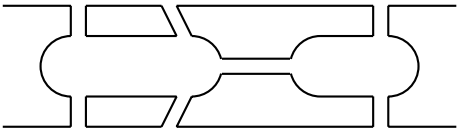


Рис. 3.28

Пример. В графе Q на рис. 3.27 цепь

$$d, 4, b, 2, c, 7, f, 9, e, 8, d$$

является циклом. □

Замечание. Само слово «маршрут» указывает на то, что мы как бы проходим по вершинам и ребрам графа. Иногда вместо слова «маршрут» используется слово «путь», но мы так будем называть маршрут в орграфе. Для многих графов (но не для всех) маршрут однозначно определяется указанием последовательности входящих в него ребер или вершин. Мы будем часто этим пользоваться.

Рассмотрим следующую задачу. Через старинный город Кенигсберг, нынешний Калининград, протекает река Прегель. Триста лет назад берега этой реки и два острова были связаны семью мостами так, как показано на рис. 3.28. (Разумеется, это лишь схема расположения.) Можно ли так проложить свой путь, чтобы пройти по каждому мосту ровно один раз и оказаться на том же берегу?

Поскольку для решения задачи неважно, на каком расстоянии друг от друга находятся мосты, рис. 3.28 можно заменить мультиграфом, в котором вершины соответствуют берегам реки и островам, а ребра — мостам (рис. 3.29). Интересующую нас задачу можно сформулировать так: содержит ли изображенный на рис. 3.29 мультиграф цикл, включающий все его ребра? Поскольку число вершин графа невелико, ответ на задачу можно получить, просмотрев все циклы. В 1736 г. Л. Эйлер нашел признак существования такого цикла, пригодный для любого графа. Считается, что с этого момента и началось развитие теории графов.

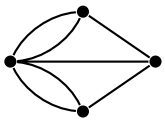


Рис. 3.29

Прежде чем сформулировать найденный Эйлером признак, приведем несколько определений.

3.4.6. *Граф называется* ***связным****, если для любых двух его вершин существует цепь, соединяющая эти вершины.*

Пример. На рис. 3.30 изображен связный граф, а на рис. 3.31 — несвязный. □

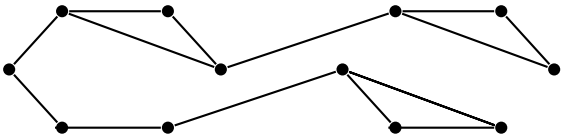


Рис. 3.30

3.4.7. *Цикл, содержащий все ребра графа, называется* ***эйлеровым****. Граф, в котором есть эйлеров цикл, называется* ***эйлеровым****.*

Пример. Граф на рис. 3.32 эйлеров. Эйлеров цикл получится, если пройти по вершинам в следующем порядке: $1, 2, 3, 4, 2, 5, 4, 6, 7, 5, 8, 1$.

Среди графов на рис. 3.22–3.24 эйлеровым является лишь граф G_{17}, в чем легко убедиться непосредственной проверкой.

Граф, изображенный на рис. 3.27, также не содержит эйлерова цикла. Это совсем не очевидно, и для проверки требуется перебрать несколько маршрутов. Следующая теорема делает эту работу излишней. □

3.4.8. *Связный граф является эйлеровым тогда и только тогда, когда степени всех его вершин четны.*

Доказательство. Предположим, что в графе есть эйлеров цикл. Поскольку граф связный, в этот цикл включены все вершины графа. Выберем одну из них за начальную и будем двигаться по ребрам так, чтобы пройти весь цикл. Войдя в очередную вершину по одному ребру, мы выйдем из нее по другому, так как в цикле одно и то же ребро не может встречаться дважды. Если на каком-то шаге мы снова попадем в эту вершину, то войдем в нее по третьему ребру, а выйдем по четвертому, и т. д.

Эйлеров цикл содержит все ребра графа, поэтому мы обязательно пройдем по всем ребрам, инцидентным данной вершине. Видим, что если вершина не является исходной точкой маршрута, то ее степень четна. Степень начальной вершины также четна. Действительно,

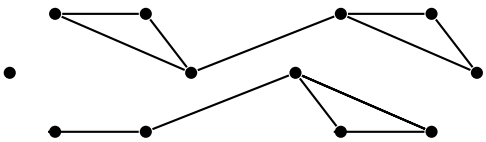


Рис. 3.31

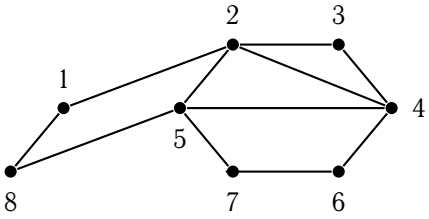


Рис. 3.32

начнем маршрут из другой вершины. Теперь вершина не начальная и проведенные выше рассуждения показывают, что ее степень четна.

Докажем, что если степени всех вершин графа G четны, то этот граф является эйлеровым. Для этого поэтапно будем строить эйлеров цикл. Примем одну из вершин графа G за исходную и обозначим ее через v_1. Будем прокладывать маршрут, двигаясь по ребрам графа так, чтобы не проходить дважды по одному ребру.

Предположим, что через какое-то число шагов мы попали в вершину u, отличную от начальной. Пусть до этого мы уже побывали в этой вершине $k \geq 0$ раз. Каждый раз мы входили по одному ребру и выходили по другому, так как каждое ребро входит в цикл только один раз. Вследствие этого в цикл включено k ребер, инцидентных вершине u, и еще одно ребро, по которому мы вошли в вершину последний раз. Так как степень вершины u четна, то найдется по крайней мере одно ребро, еще не пройденное и потому не включенное в цикл. По этому ребру мы вершину покинем и перейдем в смежную вершину. Если она отлична от v_1, то мы ее также покинем, и т. д. Таким образом, прокладка маршрута будет продолжаться по крайней мере до тех пор, пока мы не вернемся в вершину v_1.

Мы прокладываем маршрут в конечном графе, содержащем конечное число ребер. По каждому ребру можно пройти только один раз,

поэтому с каждым шагом число ребер, которые могут быть включены в маршрут, уменьшается и прокладка маршрута через конечное число шагов оборвется. В построенном маршруте начальная и конечная вершины совпадают, поэтому он является циклом. Обозначим его через C_1. Предположим, что в цикл C_1 вошли не все ребра графа G. Удалим из G все ребра, принадлежащие C_1, и полученный граф обозначим через G_1. Степень каждой вершины графа G четна, и удалено четное число инцидентных ей ребер, поэтому степени всех вершин графа G_1 также четны. Все вершины цикла C_1 принадлежат графу G_1. Возможны следующие случаи.

а) Множество вершин графа G_1 совпадает с множеством вершин цикла C_1. Согласно сделанному ранее предположению G_1 содержит хотя бы одно ребро w. Пусть v_2 — конец этого ребра. Примем вершину v_2 за начальную и будем двигаться по ребрам графа G_1, строя цепь. Приведенные выше рассуждения показывают, что через конечное число шагов мы непременно снова попадем в вершину v_2. Построенный цикл обозначим через C_2.

Цикл C_1 является маршрутом, т. е. последовательностью чередующихся вершин и ребер (см. 3.4.1). Поскольку C_1 содержит все вершины графа G_1, вершина v_2 также принадлежит этой последовательности. Обозначим через C_3 маршрут, включающий все элементы цикла C_1 от v_1 до v_2, затем все элементы цикла C_2 и далее все элементы цикла C_1 от v_2 до конца, т. е. до v_1 (рис. 3.33). Построенный маршрут является циклом. Если этот цикл не эйлеров, удаляем из G все принадлежащие циклу C_3 ребра и строим таким же способом еще больший цикл, и т. д.

б) Некоторые вершины графа G_1 не принадлежат циклу C_1. Пусть w — одна из таких вершин. Граф G связный, поэтому вершины v_1 и w соединены в нем каким-то маршрутом. Обозначим через z первое ребро, принадлежащее этому маршруту, но не входящее в цикл C_1, а

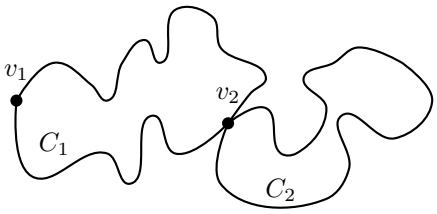


Рис. 3.33
Прокладывание маршрута C_3

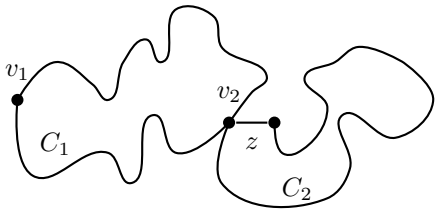


Рис. 3.34
Прокладывание маршрута, второй случай

через v_2 — конец этого ребра, принадлежащий C_1. Построим в графе G_1 цикл C_2 с началом в v_2, проходящий через z. Далее соединим C_1 и C_2 в один цикл так, как описано выше в п. а) (рис. 3.34). Если этот цикл не эйлеров, продолжим построения так, как описано выше. Так как граф G конечный, процесс получения все больших и больших циклов оборвется и последний из них будет эйлеровым. □

Хотя в доказательстве утверждения 3.4.8 и указан некоторый способ построения эйлерова цикла, для практических целей он неудобен. Опишем алгоритм Флери, позволяющий во многих случаях найти в эйлеровом графе эйлеров цикл гораздо быстрее. Для этого нам потребуются некоторые понятия и утверждения.

3.4.9. *Максимальный (по включению вершин и ребер) связный подграф графа G называется* ***связной компонентой*** *графа G.*

3.4.10. *Мостом (или* ***перешейком, разрезающим ребром****) называется такое ребро графа, удаление которого увеличивает число связных компонент.*

Пример. Граф, изображенный на рис. 3.35, имеет мосты r_1 и r_2 и состоит из четырех связных компонент. □

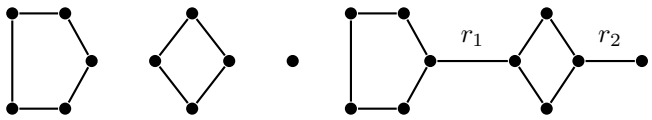


Рис. 3.35

3.4.11. *Если ребро графа принадлежит некоторому циклу, то оно не может быть мостом.*

Доказательство. Пусть в графе G есть цикл C, включающий ребро r с концами a и b (рис. 3.36). Докажем, что удаление ребра r не может увеличить числа связных компонент графа. Для этого надо доказать, что для любых двух вершин v и w графа G справедливо следующее утверждение: если вершины v и w были связаны некоторым маршрутом M, то и после удаления ребра r найдется маршрут, связывающий эти вершины.

Очевидно, если маршрут M не содержал ребра r, то после удаления указанного ребра из графа эти вершины по-прежнему окажутся связаны маршрутом M.

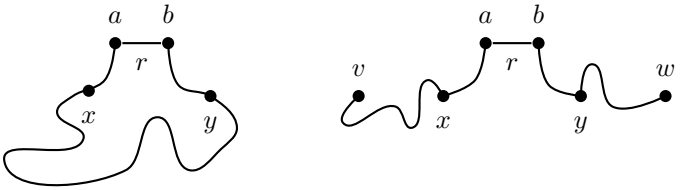


Рис. 3.36
Цикл C и маршрут M

Будем поэтому рассматривать случай, когда маршрут M содержит ребро r (рис. 3.36). В этом случае маршруту M принадлежат оба конца ребра r. Вершины a и b принадлежат циклу C и потому соединены не только ребром r, но и маршрутом M_1, получающимся из цикла C после удаления ребра r. Вставив в маршрут M вместо ребра r маршрут M_1, получим маршрут M_2, связывающий вершины v и w и не содержащий ребра r (рис. 3.37). После удаления из графа ребра r вершины v и w останутся связанными этим маршрутом. □

3.4.12. *Следующие действия приводят к построению в эйлеровом графе G эйлерова цикла. Предполагается, что ребра графа помечены так, что разные ребра имеют разные метки.*

1. *Выбираем произвольным образом вершину v графа G и ребро, инцидентное этой вершине. Выписываем метку ребра. Переходим по ребру к следующей вершине. Удаляем ребро из графа.*

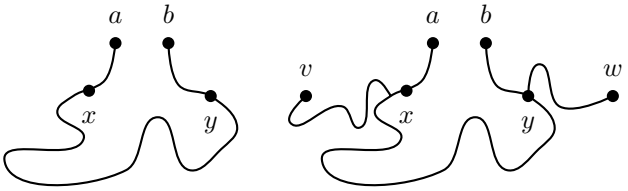


Рис. 3.37
Маршруты M_1 и M_2

2. *Пусть уже удалено $k \geqslant 1$ ребер. Выбираем какое-нибудь инцидентное очередной вершине ребро, придерживаясь следующих правил:*

а) *ребро, связывающее данную вершину с вершиной v, выбираем только в том случае, если нет других возможностей;*

б) *мост также выбираем только в том случае, когда нет других возможностей. Выписываем метку ребра. Переходим по нему к следующей вершине. Удаляем пройденное ребро.*

3. *Когда пройдены все ребра, построение цикла заканчивается.*

Доказательство. На каждом шаге мы имеем дело с графом, число ребер которого на единицу меньше числа ребер графа, рассматривавшегося на предыдущем шаге. Граф G эйлеров, в нем есть эйлеров цикл, поэтому он мостов не содержит (см. 3.4.11). Однако в процессе работы ребра удаляются и можно получить граф, имеющий мосты. Докажем следующее утверждение: *если число непомеченных ребер, инцидентных вершине, в которой мы находимся на очередном шаге, больше единицы, то некоторые из этих ребер мостами не являются.*

Обозначим вершину, в которой мы находимся, через w. Предположим, что утверждение неверно и все ребра, инцидентные вершине w, являются мостами. Возможны два случая.

а) Имеется цикл, содержащий некоторые из этих ребер. Ребра, вошедшие в цикл, не являются мостами (см. 3.4.11). Это противоречит предположению.

б) В графе нет цикла, содержащего ребро, инцидентное вершине w. Имеются две возможности.

1) При удалении любого ребра, инцидентного вершине w, начальная вершина v и вершина w остаются связанными маршрутом. Это означает, что имеется по крайней мере два разных маршрута T_1 и T_2 из w в v. Пусть один из них проходит через ребро (w, a_1), а второй — через

ребро (w, b_1). Очевидно, если две вершины графа связаны некоторым маршрутом, то существует цепь, соединяющая эти же вершины. Будем поэтому предполагать, что T_1 и T_2 — цепи.

Обозначим через u первую общую вершину этих маршрутов при движении от вершины w к вершине v по маршруту T_1. Маршрут T_1 выглядит следующим образом:

$$w, a_1, \ldots, a_i, u, a_{i+1}, \ldots, a_m, v\,.$$

Цепь T_2 можно изобразить так:

$$w, b_1, \ldots, b_j, u, b_{j+1}, \ldots, b_n, v\,.$$

Из кусков этих маршрутов строим маршрут T_3 (рис. 3.38):

$$w, a_1, \ldots, a_i, u, b_j, b_{j-1}, \ldots, b_1, w\,.$$

Докажем, что T_3 является циклом. Действительно, каждый из маршрутов

$$w, a_1, \ldots, a_i, u, \qquad u, b_j, b_{j-1}, \ldots, b_1, w$$

является цепью и потому ни один из них повторяющихся ребер не содержит. Если же у этих маршрутов есть общее ребро (a_k, b_s), то вершина a_k является общей для маршрутов T_1 и T_2 и встречается ранее вершины u, что невозможно.

Видим, что ребра (w, a_1) и (w, b_1) принадлежат циклу и потому не могут быть мостами. Получаем противоречие с предположением.

2) После удаления одного из мостов вершины v и w оказываются принадлежащими разным связным компонентам. Пусть G_1 — та компонента, которой принадлежит вершина w. Степень вершины w в графе G четна согласно условию. Если мы ее уже проходили ранее, то при каждом проходе помечали и удаляли два ребра, так что ее степень оставалась четной. Теперь мы вошли в вершину w и при этом пометили еще одно ребро, поэтому на данный момент имеется нечетное число непомеченных ребер, инцидентных вершине w. После удаления моста, инцидентного вершине w, ее степень стала четной. Степени остальных вершин графа G_1 не изменились и остались четными. Это означает, что граф G_1 эйлеров (см. 3.4.8). В нем имеется эйлеров цикл, включающий все ребра, инцидентные в G_1 вершине w, и потому эти ребра не могут быть мостами (см. 3.4.11). Тем более они не могут быть мостами в графе, который рассматривался до удаления моста. Опять получили противоречие с предположением.

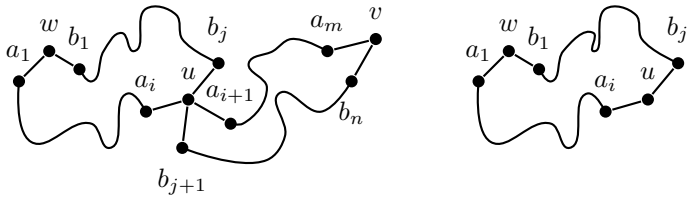


Рис. 3.38
Маршруты T_1, T_2, T_3

Мы убедились в том, что если число непомеченных ребер, инцидентных вершине w, больше единицы, то, войдя в эту вершину в процессе построения эйлерова цикла, мы всегда можем покинуть ее по ребру, не являющемуся мостом. Если же у вершины w имеется всего два инцидентных ей непомеченных ребра, то по одному мы в нее войдем, а по другому покинем и больше в нее не попадем.

Поскольку на каждом шаге у любой вершины графа, отличной от начальной вершины v и от той, в которой мы находимся, имеется четное число инцидентных ей непомеченных ребер, процесс может оборваться только тогда, когда на очередном шаге мы попадем в вершину v и при этом окажется, что нет непомеченного ребра, по которому из этой вершины можно выйти. Очевидно, на этом шаге пройденный маршрут является циклом. Осталось доказать, что этот цикл содержит все ребра графа.

Предположим, что это не так и некоторые ребра остались непомеченными. Обозначим построенный цикл через C. Если все ребра, инцидентные вершинам, вошедшим в цикл C, содержатся в этом цикле, то C является связной компонентой графа G. Однако у G имеются ребра, не вошедшие в C. Это означает, что граф G имеет по крайней мере две связные компоненты, т. е. является несвязным. Это противоречит условию.

Таким образом, среди ребер, инцидентных одной из вершин, принадлежащих циклу C, имеется непомеченное ребро r. Удалим из G все ребра, принадлежащие циклу C. Полученный граф может состоять из нескольких связных компонент. Обозначим через H ту из них, которая содержит ребро r.

У графов C и H имеется по крайней мере одна общая вершина. Обозначим через M граф, порождаемый множеством всех тех ребер, которые принадлежат циклу C и являются инцидентными всем общим вершинам графов C и H. Все принадлежащие графу M ребра

помечены. Выберем ребро с наибольшим номером и обозначим его через p. Очевидно, среди ребер из M оно было помечено последним.

Предположим, что процесс построения цикла C после присвоения метки ребру p не закончился. Поскольку в этот момент цикл C еще не построен, имеются ребра, принадлежащие этому циклу, но еще не помеченные. Обозначим граф, порожденный этими ребрами, через Q. Очевидно, Q является цепью, соединяющей вершину v с одним из концов ребра p.

У графов H и Q нет общих вершин. Действительно, пусть u — общая вершина этих графов. Она принадлежит графу M. Ей инцидентны два ребра, принадлежащие циклу C и потому также содержащиеся в M. Они помечены и, следовательно, не принадлежат графу Q. Но тогда и u не принадлежит цепи Q.

Если некоторое непомеченное ребро t соединяет какие-либо вершины графов Q и H, то его конец, принадлежащий Q, содержится и в H, т. е. является общей вершиной этих графов, что, как мы видели, невозможно. Таким образом, нет непомеченных ребер, соединяющих вершины графов Q и H.

Представим себе, что в процессе присвоения меток ребрам цикла C мы попали в конец f ребра p, не принадлежащий графу Q, и на следующем шаге собираемся присвоить номер этому ребру. Все ребра, получившие метки, удалены, и ввиду сказанного выше p является единственным ребром, соединяющим вершины графов Q и H. Очевидно, на данный момент это ребро является мостом. Присваивая ему номер, мы выбираем мост. Однако вершина f соединена непомеченным ребром с какой-то другой вершиной графа H, так как этот граф связный. Видим, что при построении цикла C был выбран исходящий из f мост, хотя имелась другая возможность. Это противоречит предписаниям алгоритма.

Если же после присвоения ребру p метки работа была окончена, то концом этого ребра является вершина v. Обозначим через z другой конец ребра p.

По предположению ребро p инцидентно одной из вершин, принадлежащих как циклу C, так и графу H. Рассмотрим два случая.

1) Общей является вершина z. Граф H связный, поэтому из вершины z идет ребро в другую вершину этого графа. Оно не имеет метки. Видим, что было выбрано ребро p, ведущее в вершину v, хотя имелась другая возможность.

2) Общей является вершина v. Приведенные в п. 1) рассуждения показывают, что имеется ребро, выходящее из вершины v, принадле-

жащее графу H и потому непомеченное. По этому ребру вершину v можно было покинуть и продолжить работу.

Таким образом, цикл C не содержит все ребра графа G только тогда, когда совершались действия, противоречащие алгоритму, описанному в 3.4.12. □

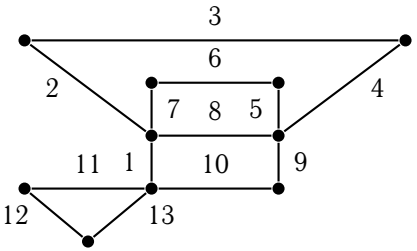


Рис. 3.39

Пример. Ребра графа на рис. 3.39 занумерованы в соответствии с алгоритмом, указанным в 3.4.12. □

3.4.13. *Цепь в графе называется* ***эйлеровой****, если она содержит все ребра графа.*

3.4.14. *Эйлерова цепь в графе существует тогда и только тогда, когда этот граф связный и ровно две его вершины имеют нечетную степень.*

Доказательство. «⇒» Пусть v_1 и v_2 — концы эйлеровой цепи. Соединив их ребром r, получим замкнутую цепь, т. е. цикл. Он содержит все ребра нового графа и потому является эйлеровым. Согласно 3.4.8 в полученном графе степени всех вершин, в том числе вершин v_1 и v_2, четны. Удаляя ребро r, вернемся к исходному графу. У вершин v_1 и v_2 степени уменьшились на единицу и стали нечетными. Степени остальных вершин остались четными.

«⇐» Предположим, что нечетные степени имеют вершины u и v. Соединив их ребром r, получим граф (или мультиграф), у которого степени всех вершин четны. Такой граф содержит эйлеров цикл. Удаляя из этого цикла ребро r, получим эйлерову цепь. □

3.4.15. *Метод доказательства утверждения* 3.4.14 *подсказывает следующий способ построения эйлеровой цепи в графе, содержащем*

ровно две вершины нечетной степени. Добавляем в граф ребро, соединяющее эти вершины. Описанным в 3.4.12 *методом строим эйлеров цикл. Удаляем добавленное ребро и получаем эйлерову цепь.* □

В 1859 г. ирландский математик У. Гамильтон придумал игру «Кругосветное путешествие», в которой требовалось обойти по ребрам некоторый граф так, чтобы побывать в каждой вершине в точности один раз.

3.4.16. *Цикл называется* ***простым****, если все его вершины, кроме первой и последней, попарно различны.*

3.4.17. *Простой цикл, содержащий все вершины графа, называется* ***гамильтоновым****. Граф называется* ***гамильтоновым****, если он содержит гамильтонов цикл.*

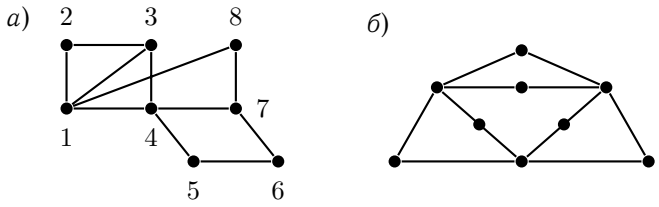


Рис. 3.40

Пример. На рис. 3.40*а* изображен гамильтонов граф, не являющийся эйлеровым, а на том же рис. 3.40*б* граф эйлеров, но не гамильтонов. Для гамильтонова графа цифрами указан порядок обхода его вершин. □

Стоит отметить, что каждый граф G можно превратить в гамильтонов путем добавления нескольких ребер. Действительно, выберем в G какую-нибудь вершину в качестве начальной и будем двигаться по ребрам, посещая другие вершины. Предположим, что мы попали в такую вершину v, из которой по инцидентным ей ребрам можно пройти только в те вершины, в которых мы уже побывали, и в то же время в графе еще есть вершины, в которых мы не были. Добавим к графу G ребро, соединяющее v с одной из еще не посещенных вершин. Теперь можно двигаться дальше. Повторив эти действия несколько раз, мы пройдем по всем вершинам графа.

Похожие, на первый взгляд, задачи прохождения по всем ребрам графа и по всем его вершинам на самом деле являются совершенно

разными. Как мы уже выяснили, существует очень простое условие, выполнение которого необходимо и достаточно для того, чтобы граф был эйлеровым. Для гамильтоновых графов таких условий не найдено. Для эйлеровых графов придуман алгоритм, быстро приводящий к построению эйлерова цикла. Для гамильтоновых графов такой алгоритм неизвестен. В то же время многие прикладные задачи сводятся к проблеме построения гамильтонова цикла. В связи с этим найдено много различных критериев гамильтоновости графа. Некоторые из них основываются на следующих простых соображениях. В полном графе любая пара вершин соединена ребром (см. 3.1.9) и из каждой вершины можно попасть в любую другую вершину. Очевидно, такой граф является гамильтоновым. Кажется правдоподобным, что если удалить небольшую часть ребер, то полученный граф все еще будет гамильтоновым. При этом следует оставить у каждой вершины достаточное количество ребер для того, чтобы при организации гамильтонова цикла можно было иметь необходимую степень свободы передвижения. Приведем один из таких критериев.

3.4.18 (условие Дирака)**.** *Если граф G имеет $n \geqslant 3$ вершин и степень каждой вершины не меньше числа $n/2$, то этот граф гамильтонов.*

Доказательство. Предположим, что граф G гамильтонова цикла не содержит. Как уже говорилось, его можно превратить в гамильтонов путем добавления каких-то ребер $r_1, \ldots, r_k$. Будем предполагать, что гамильтонов граф возник лишь после добавления ребра r_k. Обозначим этот граф через G_1, а через G_2 — граф $G_1 - r_k$. Согласно сказанному граф G_1 гамильтонов, а G_2 не гамильтонов. Отметим, что множества вершин графов G, G_1 и G_2 совпадают и степень каждой вершины в графах G_1 и G_2 не может быть меньше степени той же вершины в графе G, т. е. числа $n/2$.

Докажем, что на самом деле при сделанных предположениях степень одной из вершин графа G_2 меньше $n/2$. Обозначим через a и b концы ребра r_k. Необходимость добавления ребра r_k показывает, что гамильтонов цикл в графе G_1 непременно включает в себя это ребро. Поскольку цикл представляет собой замкнутый маршрут, то можно принять его конец a за исходную вершину, следуя по гамильтонову циклу, пройти по всем вершинам графа G_1 и вернуться в a. Очевидно, при этом направление обхода можно выбрать так, что r_k будет последним пройденным ребром. Перечислим пройденные вершины:

$$v_1 = a, v_2, v_3, \ldots, v_{n-1}, v_n = b\,. \tag{3.4.1}$$

Так как цикл гамильтонов, эта последовательность содержит все вершины графа G_1 и соседние вершины связаны ребрами. Найденный цикл изображен на рис. 3.41.

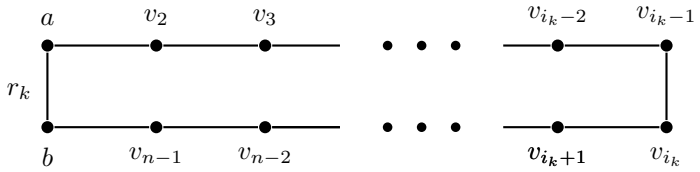


Рис. 3.41
Гамильтонов цикл в графе G_1

Докажем, что вопреки предположению граф G_2 также содержит гамильтонов цикл. Предположим, что окружение вершины a в графе G_1 состоит из вершин

$$v_{i_1}, v_{i_2}, \ldots, v_{i_s}\,, \quad i_1 < i_2 < \ldots < i_s\,, \tag{3.4.2}$$

и что одна из вершин

$$v_{i_2-1}, v_{i_3-1}, \ldots, v_{i_s-1}\,, \tag{3.4.3}$$

например, v_{i_k-1}, в графе G_2 соединена ребром с b. (Из (3.4.1) следует, что вершина v_{i_1} совпадает с v_2 и потому v_{i_1-1} совпадает с v_1, т. е. с a. Ребро (a, b) не принадлежит графу G_2. Ввиду этого вершину $a = v_{i_1-1}$ пока исключаем из рассуждений.) Сделанное предположение иллюстрирует рис. 3.42.

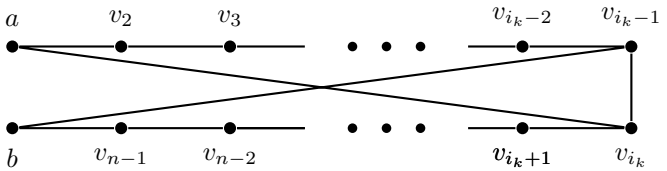


Рис. 3.42
Подграф графа G_2

Рассмотрим такую последовательность вершин:

$$a, v_2, v_3, \ldots, v_{i_k-1}, b, v_{n-1}, v_{n-2}, \ldots, v_{i_k}, a\,.$$

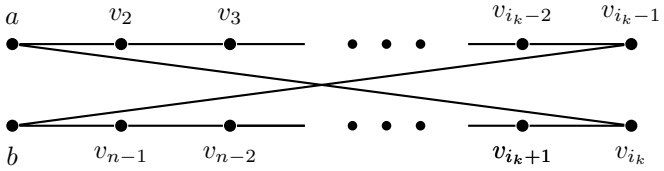


Рис. 3.43
Гамильтонов цикл в графе G_2

В этой последовательности все вершины графа G_2, за исключением вершины a, перечислены без повторений, любые две соседние вершины соединены ребром и все ребра принадлежат графу G_2. Это означает, что G_2 вопреки предположению содержит гамильтонов цикл (рис. 3.43).

Приходим к выводу, что ни одна из вершин в последовательности (3.4.3) не может принадлежать окружению вершины b в графе G_2. Число вершин в этой последовательности на единицу меньше числа вершин в последовательности (3.4.2), содержащей все вершины из окружения вершины a. Согласно условию в окружении вершины a имеется не менее $n/2$ вершин. Таким образом, последовательность (3.4.3) содержит не менее $n/2-1$ вершин. Граф G_2, кроме вершины b, содержит еще $n-1$ вершин, поэтому окружение вершины b содержит не более $(n-1)-(n/2-1)=n/2$ вершин. Однако вершина a также не принадлежит этому окружению и не содержится в последовательности (3.4.3). Отсюда следует, что число m вершин в окружении вершины b удовлетворяет следующему неравенству:

$$m \leqslant \frac{n}{2} - 1 < \frac{n}{2}\,.$$

Видим, что степень вершины b графа G_2 меньше $n/2$. Степень вершины b графа G не может быть больше степени вершины b графа G_2, следовательно, она также меньше $n/2$, а это противоречит условию. □

Приведем также одно из необходимых условий существования в графе гамильтонова цикла.

3.4.19. *Точкой сочленения (или* ***разделяющей вершиной***, ***разрезающей вершиной****) называется такая вершина графа, удаление которой увеличивает число связных компонент этого графа. Связный граф, не имеющий точек сочленения, называется* ***блоком****.*

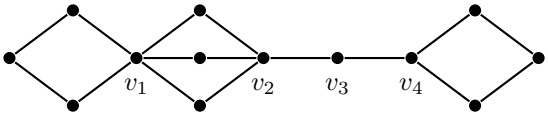


Рис. 3.44
Граф G_{17}, имеющий точки сочленения

Пример. У изображенного на рис. 3.44 графа точками сочленения являются вершины v_1, v_2, v_3, v_4. □

3.4.20. *В гамильтоновом графе нет точек сочленения.*

Доказательство. Предположим, что в гамильтоновом графе G с $n+1$ вершинами есть точка сочленения v. Пусть $v, v_1, v_2, \ldots, v_n, v$ — гамильтонов цикл. После удаления вершины v получим граф, имеющий несколько связных компонент. Обозначим через G_1 ту компоненту, которой принадлежит вершина v_1. Удаляя вершину v из G, мы удалим из графа G все ребра, инцидентные v, и только их. Так как $v_i \neq v$ для $i = \overline{1,n}$, то ребра $(v_1, v_2), \ldots, (v_{n-1}, v_n)$ неинцидентны v и потому не будут удалены из графа. Эти ребра образуют цепь, проходящую через вершины $v_1, v_2, \ldots, v_n$, следовательно, указанные вершины принадлежат одной связной компоненте, т. е. G_1. Видим, что граф, полученный из G удалением вершины v, имеет только одну связную компоненту и потому вершина v точкой сочленения не является. □

Найти гамильтонов цикл в графе G, а при необходимости и выделить все такие циклы можно полным перебором возможных вариантов. Составляем всевозможные последовательности вершин графа G так, чтобы каждая вершина, не являющаяся первым членом последовательности, встречалась в каждой последовательности ровно один раз, а первая вершина в последовательности входила в нее дважды, причем второе вхождение должно быть последним членом последовательности. Для каждой такой последовательности проверяем существование маршрута, проходящего через вершины графа в том порядке, в котором они перечислены в этой последовательности. Если такой маршрут существует, то он является гамильтоновым циклом. Очевидно, после перебора всех последовательностей будут выделены все гамильтоновы циклы в графе G.

3.5. ПЛАНАРНЫЕ ГРАФЫ

3.5.1. *Отображение вершин и ребер графа в точки и непрерывные кривые некоторого пространства, при котором вершины, инцидентные ребру, отображаются в концы кривой, сопоставленной этому ребру, называется его* ***укладкой****. Укладка называется* ***правильной****, если разным вершинам соответствуют разные точки, а кривые, соответствующие ребрам, не пересекаются и не проходят через точки, соответствующие вершинам графа (исключение составляют концы кривых).*

3.5.2. *Плоским, или* ***планарным****, называется граф, допускающий правильную укладку на плоскость.*

Часто плоским называют такой граф, который уже правильно уложен на плоскости.

Пример. На рис. 3.45 слева изображен планарный граф, справа — его правильная укладка на плоскость. □

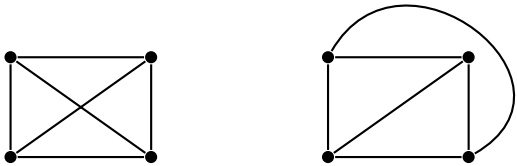


Рис. 3.45

3.5.3. *Часть плоскости, ограниченная ребрами графа, правильно уложенного на этой плоскости, и не содержащая его вершин и ребер, называется* ***гранью****. В число граней входит и внешняя по отношению к графу часть плоскости.*

3.5.4 (формула Эйлера)**.** *Число n_v вершин, число n_r ребер и число n_g граней связного плоского графа удовлетворяют соотношению*

$$n_v - n_r + n_g = 2\,. \tag{3.5.1}$$

Доказательство проведем индукцией по числу ребер. При $n_r = 0$ связный граф имеет одну вершину и, следовательно, одну грань. Формула (3.5.1) в этом случае верна. Пусть она справедлива для всех связных планарных графов с m ребром. Рассмотрим граф с $m+1$ ребрами. Он может быть получен из подходящего связного планарного графа

с m ребрами либо путем соединения ребром двух вершин, либо путем добавления еще одной вершины и соединения ее ребром с некоторой вершиной исходного графа. В первом случае число вершин не изменится, число ребер увеличится на единицу и на единицу вырастет число граней (одна из граней исходного графа разделится на две грани). Во втором случае добавляемая вершина является висячей, число граней не возрастает, на единицу возрастает число ребер и число вершин. В каждом из этих случаев формула (3.5.1) верна. □

3.5.5. *Если связный планарный граф имеет более трех вершин, то число n_r его ребер и число n_v вершин удовлетворяют неравенству*

$$n_r \leqslant 3n_v - 6\,. \tag{3.5.2}$$

Доказательство. Обозначим через n_g число граней графа G. У каждой грани граница состоит не менее чем из трех ребер. Умножим 3 на число граней, получим число $3n_g$ — оценку числа ребер графа. Некоторые ребра разделяют две грани, и их мы подсчитали дважды. Граница грани может содержать более трех ребер, значит, какие-то ребра в наш подсчет не попали. Однако никакое ребро не попало в наш счет трижды, так как никакое ребро не разделяет в плоском графе три грани. Введем следующие обозначения:

a — число ребер, не попавших в подсчет;

b — число ребер, учтенных один раз;

c — число ребер, учтенных дважды.

Очевидно,

$$2(a+b+c) = 2n_r\,, \quad b+2c = 3n_g\,,$$

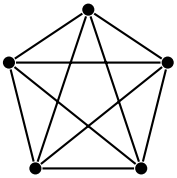
поэтому $2n_r = 2a+2b+2c \geqslant b+2c = 3n_g$, т. е. число $3n_g$ не превосходит удвоенного числа всех ребер графа: $3n_g \leqslant 2n_r$. Делим обе части на три и получаем неравенство $n_g \leqslant \frac{2}{3}n_r$. Отсюда и из (3.5.1) следует, что

$$2 = n_v - n_r + n_g \leqslant n_v - n_r + \frac{2}{3}n_r\,,$$
$$6 \leqslant 3n_v - 3n_r + 2n_r, \quad n_r \leqslant 3n_v - 6\,.$$

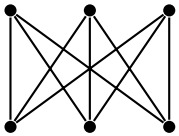
□

3.5.6. *В каждом связном планарном графе G имеется вершина, степень которой не превышает пяти.*

Доказательство. Предположим, что степень каждой вершины графа больше либо равна шести, тогда сумма степеней всех вершин графа G не может быть меньше числа $6n_v$. С другой стороны, сумма степеней всех вершин графа G равна удвоенному числу ребер (см. 3.2.3), т. е.



Граф K_5



Граф $K_{3,3}$

Рис. 3.46

$2n_r$. Получаем неравенство $6n_v \leqslant 2n_r$, а из него, сокращая обе части на 2 — соотношение

$$3n_v \leqslant n_r\,.$$

Однако это неравенство неверно, так как $3n_v - 6 \geqslant n_r$ согласно 3.5.5. □

3.5.7. *Изображенные на рис.* 3.46 *графы* K_5 *и* $K_{3,3}$ *не являются планарными.*

Доказательство. а) Граф K_5 имеет 5 вершин и 10 ребер. Эти числа не удовлетворяют соотношению (3.5.2), так как $5 \cdot 3 - 6 = 9 < 10$. б) Граф $K_{3,3}$ имеет 6 вершин и 9 ребер. Предполагая, что граф планарный, из (3.5.1) найдем число n_g его граней:

$$6 - 9 + n_g = 2, \quad n_g = 5\,.$$

Любой цикл в этом графе содержит не менее четырех ребер. Это означает, что в его плоской укладке каждая грань должна быть ограничена не менее чем четырьмя ребрами. Далее рассуждаем так же, как и при доказательстве утверждения 2.5.5. Умножая число граней на 4, мы некоторые ребра учитываем дважды, но ни одно ребро не считаем трижды, поэтому $4n_g \leqslant 2n_r$, откуда $2n_g \leqslant n_r$. Число ребер и граней графа $K_{3,3}$ этому неравенству не удовлетворяет, так как $2 \cdot 5 > 9$. □

3.5.8. ***Теорема Понтрягина — Куратовского.*** *Граф является планарным тогда и только тогда, когда он не содержит подграфов, стягиваемых к графам* K_5 *и* $K_{3,3}$.

Доказательство ввиду значительного объема опустим. □

Пример. Последовательно стягивая отмеченные штрихами ребра $(1,4)$, $(2,8)$, $(3,6)$, $(5,10)$, $(7,9)$, из левого графа на рис. 3.47 получаем граф K_5 (рис. 3.48 справа). Видим, что исходный граф не планарный. □

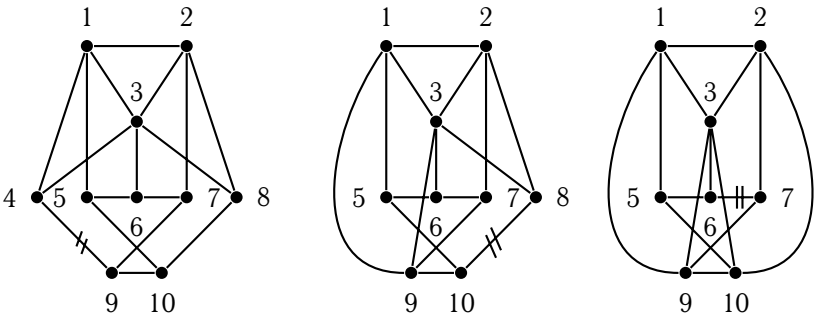


Рис. 3.47

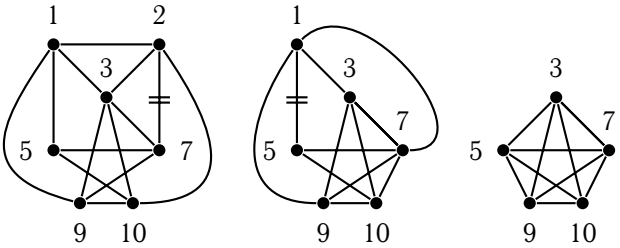


Рис. 3.48

3.6. РАСКРАСКИ ГРАФОВ

3.6.1. *Функция f, отображающая множество вершин графа G в множество $\{1, 2, \ldots, k\}$, называется k-**раскраской** графа G. Раскраска называется **правильной**, если $f(u) \neq f(v)$ для любых смежных вершин u и v графа G.*

3.6.2. *Множество всех вершин графа, отображаемых при раскраске на одно и то же число, называется **цветным** (или **одноцветным**) **классом**.*

3.6.3. *Граф, для которого существует правильная k-раскраска, называется k-**раскрашиваемым**.*

Числа $1, 2, \ldots, k$ можно истолковать как k различных цветов, в связи с чем и возникли указанные выше термины. Раскраска графа является правильной тогда и только тогда, когда смежные вершины

окрашиваются в разные цвета. Отметим, что при задании k-раскраски графа число реально использованных цветов может быть меньше k.

Пример. Граф (рис. 3.49) правильно раскрашен тремя красками. □

3.6.4. *Наименьшее число k, для которого граф G является k-раскрашиваемым, называется* ***хроматическим числом*** *этого графа и обозначается через $\chi(G)$. Если $\chi(G) = k$, то граф G называется* ***k-хроматическим****, а k-раскраска —* ***минимальной****.*

Греческое слово $chr\bar{o}ma$ переводится как *цвет, краска*.

Пример. На рис. 3.50*а* изображен 4-хроматический граф, а на рис. 3.50*б* — 3-хроматический.

Следующая задача показывает, что экзотическая, на первый взгляд, проблема раскраски графа имеет прикладное значение.

3.6.5. ***Задача составления расписания.*** *На конференции предполагается заслушать определенное количество докладов. Все участники распределены по секциям. На каждом заседании секции заслушивается один доклад. Некоторые докладчики могут делать более одного сообщения и могут выступать в разных секциях. Продолжительность всех докладов одинакова. Надо так составить расписание, чтобы на заседании каждой секции два доклада не были поставлены на одно и то же время и чтобы сообщения одного участника были поставлены в разное время.*

Решение. Сопоставим взаимно однозначно множеству всех докладов множество вершин такого графа G, у которого две вершины являются смежными тогда и только тогда, когда сопоставленные им доклады нельзя делать одновременно. Рассмотрим некоторую правильную раскраску графа G. Очевидно, доклады, соответствующие вершинам

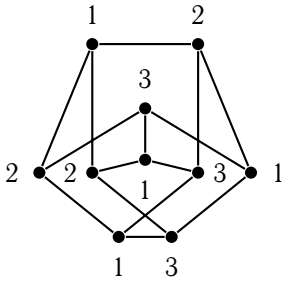


Рис. 3.49

одного цвета, можно читать одновременно. Можно сказать, что правильная раскраска определяет допустимое расписание. Верно и обратное: каждое допустимое расписание соответствует некоторой правильной раскраске графа G.

Хроматическое число графа G позволяет оценить время, необходимое для проведения конференции. Например, если продолжительность каждого доклада составляет один час и доклады, соответствующие вершинам одного цвета, читаются одновременно, то для того, чтобы сделать все доклады, надо минимум $\chi(G)$ часов. □

3.6.6. *Любой планарный граф G можно правильно раскрасить пятью красками.*

Доказательство. Рассмотрим некоторую правильную укладку графа G на плоскости. Если граф несвязный, то каждую его связную компоненту можно раскрасить независимо от остальных компонент и потому хроматическое число графа совпадает с наибольшим из хроматических чисел связных компонент. Будем поэтому предполагать, что граф G связный.

Воспользуемся индукцией по числу вершин графа. Если это число не превосходит пяти, то и красок требуется не более пяти. Предположим, что утверждение справедливо для всех планарных графов с n вершинами и что граф G имеет $n+1$ вершину. Согласно 3.5.6 в G есть такая вершина v, степень которой не превосходит пяти. Обозначим вершины, входящие в ее окружение $N(v)$, через $v_1, \ldots, v_k$. Ввиду выбора вершины v получаем оценку $k \leq 5$.

Граф $G_1 = G - v$ имеет n вершин. Согласно индуктивному предположению для его раскраски требуется не более пяти цветов. Пусть он уже раскрашен. Если одна из пяти красок не была использована при окраске вершин v_i, $i = \overline{1,k}$, то вершину v можно окрасить этой краской, и утверждение окажется справедливым.

Предположим поэтому, что при раскраске графа G_1 для окраски вершин, входящих в $N(v)$, были использованы все пять цветов. Это означает, что множество $N(v)$ состоит из пяти вершин. Будем считать, что для каждого $i = \overline{1,5}$ вершина v_i окрашена цветом i. Обозначим через H_{ij} подграф графа G_1, порожденный всеми вершинами, окрашенными в цвета i и j. Через H_{ij}^k, где $k \in \{i,j\}$, обозначим ту связную компоненту подграфа H_{ij}, которой принадлежит вершина v_k. Рассмотрим два случая.

а) Имеется два таких цвета $i,j \in \{1,\ldots,5\}$, что $H_{ij}^i \neq H_{ij}^j$. Это означает, что вершины v_i и v_j принадлежат разным связным компонентам графа H_{ij}. Графу H_{ij} принадлежат все вершины графа G_1, окра-

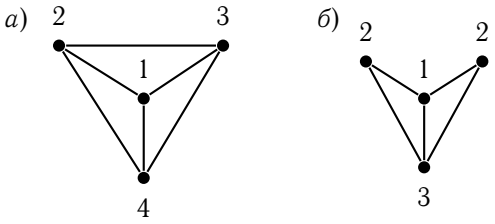


Рис. 3.50

шенные в цвета i и j. Если две вершины принадлежат разным связным компонентам графа H_{ij}, то они не могут быть смежными ни в этом графе, ни в графе G_1. Перекрасим имеющие цвет i вершины графа H_{ij}^j в цвет j, а вершины цвета j — в цвет i. Сказанное выше означает, что новая раскраска графа G_1 также является правильной. Вершина v_j первоначально имела цвет j, а теперь она имеет цвет i. Остальные вершины из окружения вершины v цвет не изменили, следовательно, теперь при раскраске вершин из множества $N(v)$ использованы четыре цвета. Такой вариант рассмотрен выше.

б) Для любой пары цветов $i, j \in \{1, \ldots, 5\}$ вершины v_i и v_j принадлежат одной и той же связной компоненте графа H_{ij}. Пусть принадлежащие множеству $N(v)$ вершины расположены так, как указано на рис. 3.51.

Так как вершины v_1 и v_3 принадлежат одной и той же связной компоненте H_{13}^1 графа H_{13}, имеется простая цепь C_1, соединяющая v_1 и v_3. Все вершины цепи G_1 окрашены в цвета 1 и 3. Принадлежность вершин v_2 и v_4 связной компоненте H_{24}^2 означает существование простой цепи C_2, соединяющей вершины v_2 и v_4, у которой все вершины окрашены в цвета 2 и 4. Цепи C_1 и C_2 не могут иметь общих вершин, так как цвета вершин первой цепи отличны от цветов вершин второй цепи. Ввиду взаимного расположения вершин v_1–v_4 какие-то ребра цепей C_1 и C_2 должны пересечься. Однако это невозможно, так как граф G_1 правильно уложен. Приходим к выводу, что случай б) невозможен. □

Существует ряд алгоритмов раскраски графов (не только планарных). Их можно разделить на два класса. К одному относятся алгоритмы, приводящие к минимальной раскраске. Эти алгоритмы требуют большого объема работы. Другой класс образуют более простые алгоритмы, при использовании которых можно надеяться получить

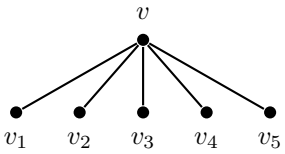


Рис. 3.51

раскраску, близкую к минимальной. Ниже описываются некоторые алгоритмы, относящиеся ко второму классу.

3.6.7. ***Алгоритм последовательной раскраски.***

1. *Выбираем произвольно одну из вершин графа и окрашиваем ее в цвет* 1.

2. *На очередном шаге среди еще не окрашенных вершин также произвольно выбираем вершину* v *и окрашиваем ее в цвет, удовлетворяющий двум условиям:*

а) *этот цвет не использовался при окраске никаких вершин, входящих в окружение вершины* v*;*

б) *цвет имеет наименьший номер среди всех цветов, удовлетворяющих условию* а).

3. *После окраски последней вершины работа заканчивается.*

Обоснование алгоритма. Условие 2а) гарантирует, что раскраска получится правильной. Условие 2б) позволяет надеяться, что полученная раскраска не слишком сильно отличается от минимальной. □

3.6.8. *Множество вершин графа называется* ***независимым****, если никакие две вершины из этого множества не являются смежными. Независимое множество вершин называется* ***максимальным****, если оно не является собственным подмножеством другого независимого множества вершин того же графа.*

Нетрудно заметить, что при правильной раскраске графа каждый цветной класс является независимым множеством.

3.6.9. ***Оптимизирующий алгоритм раскраски.*** *Предположим, что граф* G *каким-то способом уже раскрашен* k *цветами.*

1. *Пусть* V_1 *— множество всех вершин этого графа, имеющих цвет* 1*. Поскольку раскраска правильная, множество* V_1 *независимое. Оно является подмножеством некоторого максимального независимого множества* V_1^m*. Перекрасим в цвет* 1 *все те вершины из* V_1^m*, цвет которых отличен от* 1.

2. *На s-м шаге рассматриваем подграф графа G, порожденный множеством всех вершин, окрашенных в цвета $s, s+1, \ldots, k$. Пусть V_s — множество всех вершин этого подграфа, окрашенных в цвет s. Перекрашиваем в цвет s все те вершины из максимального независимого множества $V_s^m \supseteq V_s$, цвет которых отличен от s.*

3. *Работа заканчивается после просмотра всех цветных классов графа G.*

Обоснование алгоритма. Раскраска графа G, получаемая на каждом шаге, является правильной, так как никакие две вершины из множества V_s^m не являются смежными (см. 3.6.8). Число первоначально использованных цветов в процессе работы по алгоритму 3.6.9 не возрастает и даже может уменьшиться. В частности, если начальная раскраска была минимальной, то из нее получится также минимальная раскраска. □

Алгоритм 3.6.9 нетрудно приспособить и для случая, когда граф еще не раскрашен.

3.6.10. ***Алгоритм раскраски.***

1. *Выбираем в графе G некоторое максимальное независимое множество вершин и окрашиваем их цветом* 1. *Удалив окрашенные вершины, получим граф G_1.*

2. *Пусть на предыдущем шаге построен граф G_k. Выбираем в нем некоторое максимальное независимое множество вершин и окрашиваем их в цвет $k+1$. Удалив окрашенные вершины, получим граф G_{k+1}.*

3. *После окраски последней вершины работа оканчивается.* □

Окрасив вершины графа по алгоритму 3.6.10, можно попытаться улучшить результат. Для этого надо изменить нумерацию цветов и воспользоваться алгоритмом 3.6.9.

3.7. ДЕРЕВЬЯ

3.7.1. *Связный граф, не имеющий циклов, называется* ***деревом****. Связный подграф дерева называется* ***поддеревом****. Граф, у которого все связные компоненты являются деревьями, называется* ***лесом****.*

Пример. Каждый из изображенных на рис. 3.52 связных графов является деревом, а их объединение — лесом. □

3.7.2. *В дереве любая цепь является простой.*

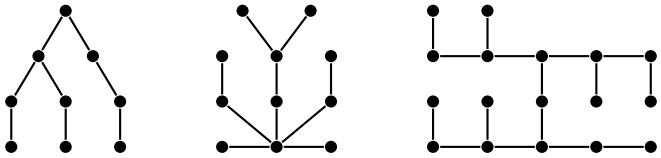


Рис. 3.52
Лес

Доказательство. Предположим, что в дереве G нашлась цепь, проходящая через вершины $v_1, v_2, \ldots, v_k$ и не являющаяся простой. Это означает, что какая-то вершина встречается в ней дважды, т. е. что существуют такие $i, j \in \{1, 2, \ldots, k\}$, что $i < j$ и $v_i = v_j$. Очевидно, $j \neq i + 1$ и $j \neq i + 2$, так как мы рассматриваем простые графы. Но тогда цепь $v_i, v_{i+1}, \ldots, v_j$ является циклом. Это невозможно, поскольку G — дерево. □

3.7.3. *Граф G_1, полученный добавлением к дереву G новой вершины и ребра, связывающего эту вершину с одной из вершин графа G, является деревом.*

Доказательство. Обозначим добавленную вершину через v, и пусть она соединена ребром с вершиной w. Очевидно, в графе G_1 вершина v является висячей. Граф G связный, поэтому для любой его вершины u существует маршрут, связывающий u с w. Добавив к нему ребро (w, v) и вершину v, получим маршрут, соединяющий вершины u и v. Это доказывает, что граф G_1 также связный.

Осталось показать, что в G_1 нет циклов. Предположим, что это не так, и что в G_1 имеется цикл C. Ребро (v, w) графа G_1 является мостом (см. 3.4.10) и потому не может принадлежать циклу C (см. 3.4.11). Видим, что все принадлежащие циклу C ребра содержатся в графе G. Ввиду этого C — цикл в дереве G, что невозможно. □

3.7.4. *Граф G является деревом тогда и только тогда, когда он не содержит циклов, но при добавлении любого ребра из него получается граф, содержащий ровно один простой цикл.*

Доказательство. «⇒» Если граф G является деревом, то он не имеет циклов (см. 3.7.1). Пусть u и v — две несмежные вершины графа. Поскольку граф связный, вершина v соединена некоторой цепью C с u. Из 3.7.2 следует, что эта цепь простая. Добавим к G ребро (u, v) и полученный граф обозначим через G_1. Присоединив к C в конце

указанное ребро и вершину v, получим простой цикл, содержащийся в G_1.

Предположим, что в графе G_1 есть второй цикл F. Если он не содержит ребра (u, v), то все его элементы принадлежат дереву G и F является циклом в G, что невозможно. Вследствие сказанного F можно представить в виде $v, v_1, \ldots, v_k, u, v$. Пусть C имеет вид $v, z_1, \ldots, z_s, u, v$. Тогда

$$v, v_1, \ldots, v_k, u, z_s, z_{s-1}, \ldots, z_1, v$$

является циклом в G, что также невозможно.

«$\Leftarrow$» Дано: граф G не содержит циклов, но при добавлении к нему любого ребра получается граф, содержащий один простой цикл. Предположим, что G не является деревом. Из 3.7.1 следует, что G не может быть связным. Это означает, что какие-то вершины u и v графа G нельзя соединить маршрутом. Добавляя к G ребро (u, v), получим граф G_1, содержащий цикл. Проведенные выше рассуждения показывают, что этот цикл непременно содержит ребро (u, v), вследствие чего его можно представить в виде $u, z_1, z_2, \ldots, z_k, v, u$. Но тогда $u, z_1, z_2, \ldots, z_k, v$ — цепь в G, соединяющая вершины u и v, что противоречит выбору этих вершин. □

3.7.5. *Если у дерева есть ребро, то у него есть висячая вершина.*

Доказательство. Предположим, что у дерева D висячих вершин нет. Так как граф D связный, то отсюда следует, что степень каждой его вершины не меньше двух. Построим цепь следующим образом. Выберем некоторую вершину v_1, из нее перейдем в вершину v_2, затем — в $v_3 \notin \{v_1, v_2\}$ и т. д. Поскольку у каждой вершины имеется по крайней мере два инцидентных ей ребра, то, придя впервые в какую-нибудь вершину, мы всегда можем ее покинуть. Движение будет продолжаться до тех пор, пока мы не попадем в вершину, из которой нельзя выйти, так как все инцидентные ей ребра уже включены в цепь. Это, однако, означает, что данную вершину мы уже проходили ранее, и построенная цепь содержит цикл, что невозможно, так как D — дерево. □

3.7.6. *Граф G является деревом тогда и только тогда, когда он связный и число его ребер на единицу меньше числа вершин.*

Доказательство. Пусть G имеет m вершин.

«$\Rightarrow$» Докажем утверждение индукцией по m. Если $m = 1$, то граф не имеет ребер. Если $m = 2$, то G либо не имеет ребер, состоит из двух

связных компонент и не является деревом, либо имеет одно ребро, соединяющее обе вершины. В обоих случаях утверждение справедливо.

Предположим, что $m > 2$ и что каждое дерево с $m-1$ вершинами имеет $m-2$ ребер. Граф G связный, поэтому в нем есть ребра. Согласно 3.7.5 одна из его вершин висячая. Обозначим ее через w. Удалим w вместе с инцидентным ей ребром r. Докажем, что полученный граф G_1 связный. Действительно, если это не так, то в G_1 есть вершины u и v, которые нельзя соединить маршрутом. В связном графе G имеется маршрут M, соединяющий u и v. Очевидно, ребро r и вершина w принадлежат M.

Пусть z — вершина, связанная с w ребром r. Так как в вершину w можно попасть только по ребру r, маршрут M содержит подпоследовательность z, r, w, r, z. Заменив ее вершиной z, получим маршрут в графе G_1, соединяющий u и v (рис. 3.53).

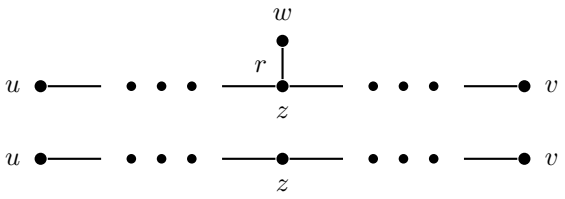


Рис. 3.53
Маршруты в графах G и G_1

Итак, граф G_1 связный. Он не содержит циклов, так как иначе циклы содержал бы и граф G. Видим, что G_1 является деревом, имеющим $m-1$ вершину. Согласно индуктивному предположению G_1 имеет $m-2$ ребра. Но в таком случае граф G имеет $m-1$ ребро.

«$\Leftarrow$» Это утверждение также докажем индукцией по числу вершин графа. Граф, состоящий из одной вершины, является деревом. Пусть $m \geqslant 2$ и каждый граф, у которого число ребер на единицу меньше числа вершин, является деревом. Возможны два случая.

а) У графа G имеется висячая вершина. После ее удаления получим граф G_1 с $m-1$ вершиной и $m-2$ ребрами. Проведенные выше рассуждения показывают, что граф G_1 связный. Согласно индуктивному предположению G_1 является деревом. Вследствие 3.7.3 G также является деревом.

б) Предположим, что у G нет висячей вершины. В этом случае степень каждой его вершины не может быть меньше двух и сумма s степеней всех вершин не меньше числа $2m$. Однако согласно 3.2.3

s равно удвоенному числу ребер, т. е. числу $2(m-1)$. Полученное противоречие доказывает, что у G имеются висячие вершины. □

3.7.7. *Подграф графа называется* ***о́стовным*** *(или* ***суграфом****), если он содержит все вершины графа.*

3.7.8. *Остовный подграф графа G, пересечение которого с каждой связной компонентой графа G является деревом, называется* ***о́стовом*** *графа G.*

3.7.9. *Остовный подграф связного графа, являющийся деревом, называется* ***о́стовным деревом****.*

Остовное дерево называют также **каркасом, остовом, скелетом, стягивающим деревом**.

3.7.10. *Каждый связный псевдограф содержит остовное дерево.*

Доказательство. Следующий алгоритм всегда приводит к выделению остовного дерева связного графа G.

1. Выбираем некоторую вершину графа G. Обозначим ее через v_1. Подграф G_1 графа G, состоящий из этой вершины, является деревом.

2. Пусть уже построено дерево G_k с вершинами $v_1, \ldots, v_k$. Если оно содержит не все вершины графа G, ищем в G вершину, смежную какой-либо вершине v_j графа G_k, но не принадлежащую этому графу. Такая вершина всегда существует, так как граф G связный. Обозначим найденную вершину через v_{k+1}. Добавим к G_k вершину v_{k+1} и ребро (v_j, v_{k+1}). Обозначим построенный граф через G_{k+1}. Согласно 3.7.3 этот граф является деревом.

3. Когда на очередном шаге к дереву будет добавлена последняя вершина графа G и дерево превратится в остовное, работа заканчивается. □

3.7.11. *Лес, состоящий из остовных деревьев всех связных компонент графа, назовем* ***о́стовным****.*

В приведенном в 3.7.10 алгоритме выделения остовного дерева графа используется метод последовательного добавления ребер. Можно с той же целью пойти другим путем и последовательно удалять ребра графа так, чтобы разрывать циклы в графе. Как только получим граф без циклов, работа закончится. Для построения остовного леса в несвязном графе приме́ним этот метод к каждой связной компоненте графа.

3.7.12. *Наименьшее число удалений ребер графа, приводящих к построению остовного леса, называется* ***цикломатическим числом*** *этого графа.*

3.7.13. *Цикломатическое число связного графа $G(V, E)$ равно $|E| - |V| + k(G)$, где $k(G)$ — число связных компонент графа G.*

Доказательство. Положим $|E| = m$ и $|V| = n$. Пусть граф G связный. Остовное дерево графа G содержит все его вершины и, следовательно, имеет $n-1$ ребер (см. 3.7.6). Число ребер, которые необходимо удалить из G для построения этого дерева, равно

$$m - (n-1) = m - n + 1\,.$$

Предположим, что граф G несвязный и имеет $k(G)$ связных компонент. Обозначим через m_i и n_i число ребер и вершин i-й связной компоненты. Для построения остовного дерева этой компоненты надо удалить $m_i - n_i + 1$ ребер. Суммируя по всем компонентам и учитывая, что

$$m_1 + \ldots + m_{k(G)} = m\,, \quad n_1 + \ldots + n_{k(G)} = n\,,$$

получаем нужное число $|E| - |V| + k(G)$. □

3.8. ОРИЕНТИРОВАННЫЕ ГРАФЫ

Некоторые ранее приведенные определения в случае ориентированных графов приходится переделывать уже потому, что в таких графах нет ребер, а есть дуги.

3.8.1. ***Цепью в орграфе*** *называется такая последовательность вершин $v_1, \ldots, v_n$, в которой для любого i либо (v_i, v_{i+1}), либо (v_{i+1}, v_i) является дугой. Цепь, у которой первая и последняя вершины совпадают, называется* ***циклом****.*

3.8.2. *Орграф, у которого любые две вершины соединены цепью, называется* ***слабо связным****.*

3.8.3. *Цепь в орграфе, в которой могут совпадать только концы, называется* ***простой****.*

Обычно двигаясь по дугам орграфа, учитывают их ориентацию. Это означает, что движение возможно только из начала дуги в ее конец.

3.8.4. *Последовательность $v_1, r_1, v_2, r_2, \ldots, v_{n-1}, r_{n-1}, v_n$ вершин и дуг орграфа, в которой $r_i = (v_i, v_{i+1})$, $i = \overline{1, n-1}$, называется* ***путем (или ориентированным маршрутом) длины*** *$n-1$ из v_1 в v_n.*

3.8.5. *Если в орграфе существует путь из вершины u в вершину v, то говорят, что вершина v* ***достижима из*** *u.*

3.8.6. ***Матрицей достижимости*** *орграфа порядка n называется квадратная матрица того же порядка, у которой на пересечении i-й строки и j-го столбца стоит 1, если вершина v_j достижима из v_i, и 0 в противном случае. На главной диагонали стоят единицы.*

Пример. Матрица достижимости орграфа, изображенного на рис. 3.55, выглядит так:

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \qquad \square$$

3.8.7. *Орграф называется* ***односторонне связным****, если для любых двух его вершин хотя бы одна достижима из другой.*

3.8.8. *Вершины a и b орграфа G называются* ***сильно связными*** *(или* ***бисвязными, взаимно связными****), если в G существуют пути из a в b и из b в a.*

3.8.9. *Орграф, у которого любые две вершины сильно связны, называется* ***сильно связным****.*

3.8.10. *Максимальный по включению вершин сильно связный подграф орграфа называется* ***компонентой сильной связности*** *(или* ***сильной компонентой, бикомпонентой****).*

3.8.11. ***Конденсат графа*** *(или* ***граф конденсации, граф Герца****) получается из орграфа стягиванием каждой компоненты сильной связности в отдельную вершину.*

Пример. На рис. 3.54 показаны компоненты сильной связности орграфа и конденсат. $\square$

3.8.12. *Замкнутый путь в орграфе называется* ***контуром****. Контур, в котором ни одна вершина не повторяется, называется* ***простым****.*

3.8.13. *Дуга* ***выходит из вершины*** *v* ***(заходит в вершину*** *v****)****, если v является ее началом (концом).*

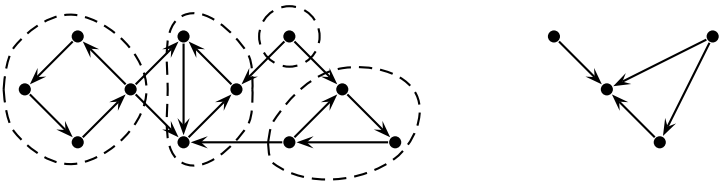


Рис. 3.54

Число дуг, заходящих в вершину v орграфа, называется ***полустепенью захода*** *(или* ***степенью входа****) вершины v. Обозначение:* $\deg_- v$ *(или* $\operatorname{indeg}(v)$*).*

Число дуг, выходящих из вершины v, называется ***полустепенью исхода*** *(или* ***степенью выхода****) вершины v. Обозначение:* $\deg_+ v$ *(или* $\operatorname{outdeg}(v)$*).*

В матрице смежности орграфа сумма элементов строки равна полустепени исхода, а сумма элементов столбца — полустепени захода вершины, соответствующей этой строке.

3.8.14. *Вершина орграфа, у которой полустепень захода равна нулю, называется* ***источником****. Вершина называется* ***стоком****, если равна нулю ее полустепень исхода.*

Пример. В графе на рис. 3.55 имеется два простых цикла:

$$v_1, v_4, v_5, v_3, v_1, \quad v_1, v_4, v_5, v_6, v_2, v_1\,.$$

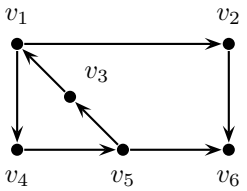


Рис. 3.55

Первый из них является контуром, а второй нет. У вершин v_1 и v_5 полустепень захода равна единице, а полустепень исхода — двум. Вершина v_6 является стоком. □

В матрице смежности орграфа столбец, соответствующий источнику, и строка, соответствующая стоку, состоят из нулей.

3.8.15. *Бесконтурный орграф называется* ***ориентированным деревом****, если он удовлетворяет следующим трем условиям:*

1) *существует вершина v, в которую не заходит ни одна дуга;*
2) *в каждую из остальных вершин заходит ровно одна дуга;*
3) *существует путь из v в любую другую вершину.*

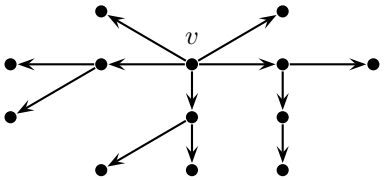


Рис. 3.56

Из этого определения следует, что в ориентированном дереве есть только одна вершина, в которую не заходит ни одна дуга, и что путь из вершины v в любую другую вершину может быть только один.

3.8.16. *Вершина ориентированного дерева, в которую не заходит ни одной дуги, называется* ***корнем*** *этого дерева.*

Пример. На рис. 3.56 изображено ориентированное дерево с корнем v. □

3.8.17. *При замене в орграфе всех дуг ребрами, т. е. после снятия с дуг ориентации, получается мультиграф, называемый* ***основанием*** *орграфа (****ассоциированным графом, соотнесенным графом****).*

Если в орграфе имелись петли, то при построении соотнесенного графа их удаляют.

Пример. Основание орграфа, изображенного на рис. 3.55, показано на рис. 3.57. □

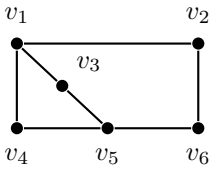


Рис. 3.57

3.9. РАССТОЯНИЕ В ГРАФАХ

Согласно 3.4.2 длиной маршрута называется число входящих в него ребер. Любая цепь является маршрутом, поэтому длиной цепи также является число ее ребер.

3.9.1. *Длина кратчайшей цепи, соединяющей две вершины графа, называется* ***расстоянием между*** *этими* ***вершинами****. Расстояние от вершины v до v считается равным нулю. Расстояние между вершинами,*

принадлежащими разным связным компонентам графа, считается равным ∞.

3.9.2. ***Матрицей расстояний*** *графа* G *с вершинами* $v_1, \ldots, v_n$ *называется квадратная матрица порядка* n, *в которой для* $i, j = \overline{1,n}$ *элемент* i*-й строки и* j*-го столбца равен расстоянию между вершинами* v_i *и* v_j.

В прикладных задачах обычно рассматриваются помеченные графы, в которых ребрам приписаны некоторые действительные числа. Длины маршрутов в таких графах определяются не числом пройденных ребер, а суммой весов этих ребер. Всю информацию о взвешенном графе можно получить из матрицы весов.

3.9.3. *Пусть вершины взвешенного графа (орграфа)* G *каким-то образом перенумерованы и через* v_k *обозначена вершина с номером* k. ***Матрица весов*** *графа (****длин дуг*** *орграфа)* G *квадратная, ее порядок совпадает с порядком графа (орграфа). Элемент, стоящий на пересечении* i*-й строки и* j*-го столбца, равен весу ребра (дуги)* (v_i, v_j), *если такое ребро (такая дуга) существует, в противном случае он равен* ∞.

При решении некоторых задач выгоднее вместо ∞ в матрице весов ставить нуль.

3.9.4. ***Весом*** *(или* ***длиной) маршрута*** *во взвешенном графе называется сумма весов ребер, входящих в этот маршрут.* ***Взвешенным расстоянием*** *между вершинами* u *и* v *называется наименьший из весов маршрутов, связывающих* u *и* v. *Маршрут от* u *до* v, *вес которого совпадает со взвешенным расстоянием между* u *и* v, *называется* ***кратчайшим***.

Ребрам графа могут быть приписаны отрицательные длины. Взвешенное расстояние между двумя вершинами в этом случае может быть отрицательным. Если граф содержит циклы отрицательной длины, то взвешенное расстояние между некоторыми вершинами графа может оказаться неопределенным, даже если эти вершины связаны маршрутом. Действительно, предположим, что некоторый маршрут от вершины u до вершины v включает в себя цикл, вес которого γ меньше нуля. Пусть вес маршрута равен δ.

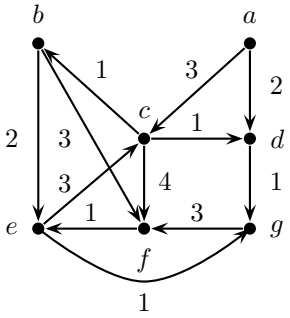


Рис. 3.58

Двигаясь из u в v по тем же ребрам, мы можем пройти по циклу k раз, а затем выйти из цикла и пройти в вершину v. Вес нового маршрута равен $k\gamma + \delta$. Поскольку $\gamma < 0$, этот вес меньше веса исходного маршрута. При увеличении числа k веса полученных маршрутов уменьшаются, поэтому взвешенное расстояние между вершинами u и v не определено.

Алгоритм Дейкстры позволяет найти взвешенные расстояния от какой-либо вершины v графа до остальных вершин в случае, когда веса всех ребер неотрицательны. Как обычно, работа происходит по шагам. На каждом шаге каждой вершине w приписываются метки, указывающие минимальный из весов изученных маршрутов, связывающих данную вершину с вершиной v и соседнюю с w вершину, через которую проходит наилучший маршрут. При поиске взвешенных расстояний в орграфе учитывается ориентация дуг.

3.9.5 (алгоритм Дейкстры).

0. *У каждой вершины ставим временную метку* $(0, \infty)$.

1. *У вершины* v*, от которой отсчитываются расстояния, метку* $(0, \infty)$ *заменяем постоянной меткой* $(v, 0)$.

2. *Ищем вершину, которая приобрела постоянную метку последней. Пусть* u *— такая вершина, и ее постоянная метка равна* (x_u, y_u)*. Предположим, что вершина* p *является концом дуги, выходящей из* u*, имеет временную метку* (x_p, y_p) *и дуга* (u, p) *имеет вес* h*. Если метка вершины* p *имеет вид* $(0, \infty)$ *или отлична от* $(0, \infty)$*, но сумма* $y_u + h$ *меньше* y_p*, то меняем метку* (x_p, y_p) *на* $(u, y_u + h)$*. Так поступаем со всеми вершинами, имеющими временные метки и являющимися концом дуги с началом в* u*.*

3. *Среди вершин, имеющих временные метки, ищем ту, у которой число, стоящее в метке вторым, наименьшее. Делаем эту метку постоянной.*

4. *Если есть вершины с временными метками, переходим к п.* 2.

5. *Работа заканчивается, когда все вершины получат постоянные метки. Расстояние, указанное в метке, является взвешенным расстоянием от данной вершины до вершины* v*.*

6. *Постоянные метки позволяют найти кратчайший маршрут от вершины* v *до любой вершины* u*. От вершины* u *переходим к вершине* u_1*, указанной в метке вершины* u*. От вершины* u_1 *переходим к вершине* u_2*, указанной в метке вершины* u_1*, и т. д. В некоторый момент мы попадем в вершину* v*. Теперь надо пройти все вершины в обратном порядке и получить требуемый маршрут.*

Примеры 1. Пользуясь алгоритмом Дейкстры, найдем взвешенные расстояния от вершины a до прочих вершин графа, изображенного на рис. 3.58. Процесс смены меток показан на рис. 3.59–3.62. □

Удобнее метки ставить не при вершинах на рисунке, а свести их в таблицу. Это к тому же позволяет обойтись без рисунка, если граф задан матрицей смежности.

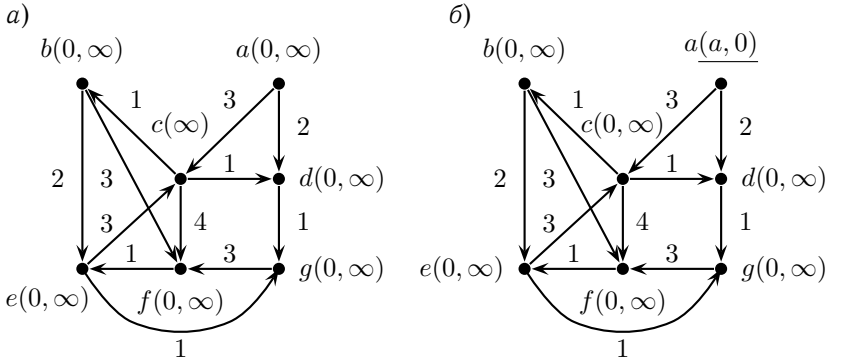


Рис. 3.59
а) все вершины получают метки; б) вершина a получает постоянную метку

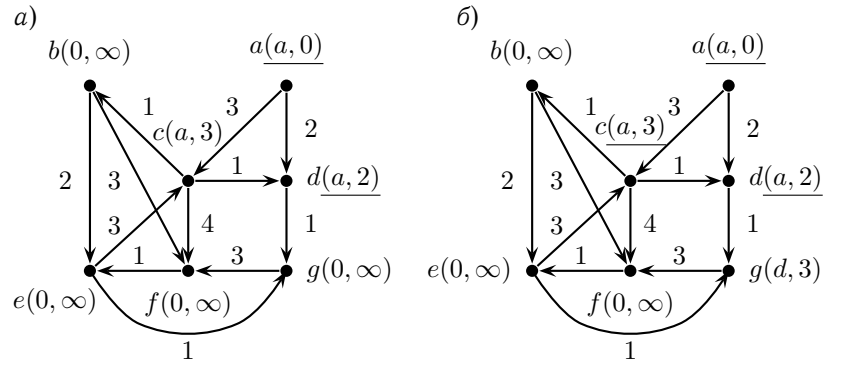


Рис. 3.60
а) меняем метки у вершин c и d, метку у d делаем постоянной; б) меняем метку у вершины g, фиксируем метку у c

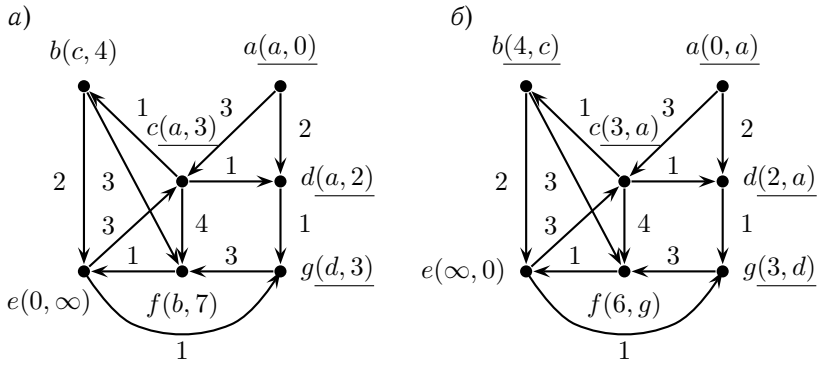


Рис. 3.61
а) меняем метки у вершин b и f, метку вершины g делаем постоянной; б) меняем метку у вершины f, метку вершины b делаем постоянной

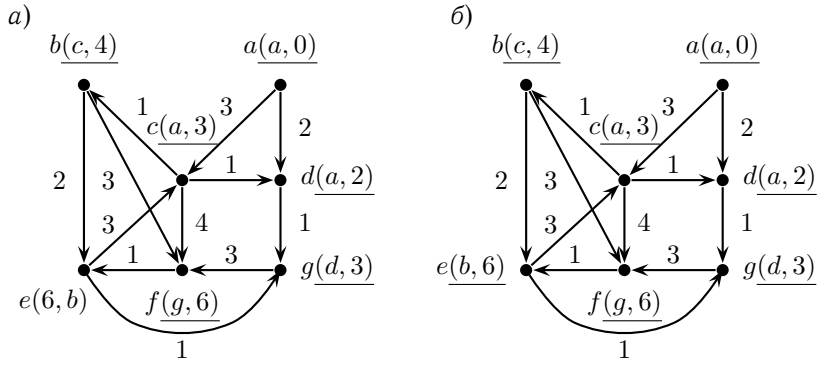


Рис. 3.62
а) меняем метку у вершины e, метку вершины f делаем постоянной; б) метку вершины e делаем постоянной

2. Пользуясь алгоритмом Дейкстры, найдем взвешенные расстояния от вершины b до прочих вершин графа, заданного матрицей длин дуг A (в первом столбце и первой строке стоят метки вершин):

$$A = \begin{pmatrix} & a & b & c & d & e & f & g \\ a & \infty & 2 & 5 & 4 & \infty & 2 & \infty \\ b & 4 & \infty & \infty & 1 & \infty & 4 & 1 \\ c & 2 & 1 & \infty & 1 & 2 & 1 & 1 \\ d & \infty & 3 & \infty & \infty & \infty & 3 & 2 \\ e & \infty & 1 & \infty & \infty & \infty & 3 & 4 \\ f & 3 & \infty & \infty & 3 & \infty & \infty & 1 \\ g & 5 & \infty & 2 & 1 & \infty & 2 & \infty \end{pmatrix}.$$

Шаг 1. Ставим метку у вершины b и делаем ее постоянной (табл. 3.1).

Таблица 3.1

	a	b	c	d	e	f	g
1		$\boxed{b,0}$					

Шаг 2. Переносим в таблицу из строки b матрицы A все отличные от ∞ числа, отметив, что соответствующие дуги выходят из вершины b. Одну из временных меток с минимальным весом, например метку при вершине d, делаем постоянной (табл. 3.2).

Таблица 3.2

	a	b	c	d	e	f	g
1		$\boxed{b,0}$					
2	$b,4$			$\boxed{b,1}$		$b,4$	$b,1$

Шаг 3. Ко всем отличным от ∞ числам из строки d матрицы A прибавляем 1 и переносим их в таблицу, отмечая, что соответствующие дуги выходят из вершины d. В каждом столбце оставляем метку с наилучшим весом, остальные вычеркиваем. В метке $b,1$ указан наименьший вес. Делаем ее постоянной (табл. 3.3).

Шаг 4. Ко всем отличным от ∞ числам из строки g матрицы A прибавляем 1 и переносим их в таблицу, отмечая, что соответствующие дуги выходят из вершины g. В каждом столбце оставляем метку с наилучшим весом, остальные вычеркиваем. Метку при вершине c делаем постоянной (табл. 3.4).

Шаг 5. Ко всем отличным от ∞ числам из строки c матрицы A прибавляем 3 и переносим их в таблицу, отмечая, что соответствующие

Таблица 3.3

	a	b	c	d	e	f	g
1		$\boxed{b,0}$					
2	$b,4$			$\boxed{b,1}$		$b,4$	$\boxed{b,1}$
3		~~$d,4$~~				~~$d,4$~~	~~$d,3$~~

Таблица 3.4

	a	b	c	d	e	f	g
1		$\boxed{b,0}$					
2	$b,4$			$\boxed{b,1}$		~~$b,4$~~	$\boxed{b,1}$
3		~~$d,4$~~				~~$d,4$~~	~~$d,3$~~
4	~~$g,6$~~		$\boxed{g,3}$	~~$g,2$~~		$g,3$	

дуги выходят из вершины c. В каждом столбце оставляем метку с наилучшим весом, остальные вычеркиваем. Метку при вершине f делаем постоянной (табл. 3.5).

Таблица 3.5

	a	b	c	d	e	f	g
1		$\boxed{b,0}$					
2	$b,4$			$\boxed{b,1}$		~~$b,4$~~	$\boxed{b,1}$
3		~~$d,4$~~				~~$d,4$~~	~~$d,3$~~
4	~~$g,6$~~		$\boxed{g,3}$	~~$g,2$~~		$\boxed{g,3}$	
5	~~$e,5$~~	~~$e,4$~~		~~$e,4$~~	$e,5$	~~$e,4$~~	~~$e,4$~~

Шаг 6. Ко всем отличным от ∞ числам строки f матрицы A прибавляем 3 и переносим их в таблицу. В каждом столбце оставляем метку с наилучшим весом. Метку при вершине a делаем постоянной (табл. 3.6).

Шаг 7. Ко всем отличным от ∞ числам строки a матрицы A прибавляем 4 и переносим их в таблицу. В каждом столбце оставляем метку с наилучшим весом. Метку при вершине a делаем постоянной (табл. 3.7).

Все вершины получили постоянные метки. Взвешенное расстояние каждой вершины от вершины b указано в постоянной метке.

Таблица 3.6

	a	b	c	d	e	f	g
1		$\boxed{b,0}$					
2	$\boxed{b,4}$			$\boxed{b,1}$		~~$b,4$~~	$\boxed{b,1}$
3		~~$d,4$~~				~~$d,4$~~	~~$d,3$~~
4	~~$g,6$~~		$\boxed{g,3}$	~~$g,2$~~		$\boxed{g,3}$	
5	~~$c,5$~~	~~$c,4$~~		~~$c,4$~~	$c,5$	~~$c,4$~~	~~$c,4$~~
6	~~$f,6$~~			~~$f,6$~~			~~$f,4$~~

Таблица 3.7

	a	b	c	d	e	f	g
1		$\boxed{b,0}$					
2	$\boxed{b,4}$			$\boxed{b,1}$		~~$b,4$~~	$\boxed{b,1}$
3		~~$d,4$~~				~~$d,4$~~	~~d,3~~
4	~~$g,6$~~		$\boxed{g,3}$	~~$g,2$~~		$\boxed{g,3}$	
5	~~$c,5$~~	~~$c,4$~~		~~$c,4$~~	$\boxed{c,5}$	~~$c,4$~~	~~$c,4$~~
6	~~$f,6$~~			~~$f,6$~~			~~$f,4$~~
7		~~$a,6$~~	~~$a,9$~~	~~$a,8$~~		~~$a,6$~~	

3. Для графа из примера 1 вместо работы с метками у вершин можно построить табл. 3.8. Число меток в строке равно степени вершины. Если полученный на очередном шаге вес, указанный в метке, не лучше веса в какой-либо метке, расположенной выше в том же столбце, метку зачеркиваем. В ином случае зачеркиваем метки, находящиеся выше. На каждом шаге одну из временных меток с наименьшим весом делаем постоянной.

В каждой строке и каждом столбце табл. 3.8 имеется только одна постоянная метка (в рамке), указывающая взвешенное расстояние от соответствующей вершины до вершины a. □

Таблица 3.8

	a	b	c	d	e	f	g
1	$\boxed{a,0}$						
2			$\boxed{a,3}$	$\boxed{a,2}$			
3							$\boxed{d,3}$
4		$\boxed{c,4}$		~~$e,4$~~		~~$e,7$~~	
5						$\boxed{g,6}$	
6					$\boxed{b,6}$	~~$b,7$~~	

3.9.6. *Остовное дерево взвешенного графа с минимальной суммой длин принадлежащих ему ребер называется* ***минимальным остовным деревом*** (МОД) *этого графа.*

Согласно 3.7.10 каждый связный граф G содержит остовные деревья. Поскольку граф G конечный, то и число таких деревьев конечно и среди них можно найти минимальное.

3.9.7. *Пусть D_1 — поддерево минимального остовного дерева D связного нагруженного графа G, r — такое ребро графа G, что*

а) *один конец ребра r принадлежит, а второй не принадлежит дереву D_1;*

б) *ребро r имеет наименьшую длину среди всех ребер, удовлетворяющих условию* а).

Граф D_2, полученный добавлением к D_1 ребра r и его конца, не принадлежащего D_1, является поддеревом некоторого МОД *графа G.*

Доказательство. Согласно 3.7.3 D_2 является деревом. Если ребро r принадлежит МОД D, о котором говорится в 3.9.7, то D_2 является поддеревом дерева D, и потому утверждение справедливо.

Предположим, что D не содержит ребра r. При добавлении этого ребра к D получится какой-то граф F, содержащий простой цикл C, проходящий через r (см. 3.7.4). Этот цикл можно задать последовательностью

$$z_1, z_2, \ldots, z_{k-1}, z_k, z_1$$

принадлежащих ему вершин. Так как ребро r входит в цикл, будем считать, что $r = (z_1, z_2)$. Один из концов ребра r принадлежит дереву D_1. Пусть это будет z_1. Тогда по условию z_2 не принадлежит D_1.

Ребра

$$(z_1, z_2), (z_2, z_3), \ldots, (z_{k-1}, z_k)$$

попарно различны, так как принадлежат циклу (см. 3.4.3, 3.4.5). Отсюда следует, что $(z_1, z_2) \neq (z_2, z_3)$, вследствие чего $z_1 \neq z_3$ и $k \geqslant 3$.

Цикл C простой, поэтому вершины

$$z_2, \ldots, z_{k-1}, z_k, z_1 \tag{3.9.1}$$

попарно различны. Вершина z_1 принадлежит D_1, а z_2 нет. Это означает, что, перебирая слева направо члены последовательности (3.9.1), мы обязательно встретим вершину z_s, обладающую следующими свойствами:

а) эта вершина принадлежит дереву D_1,

б) вершина z_{s-1} не принадлежит D_1.

Обозначим через r_1 ребро (z_{s-1}, z_s). Оно принадлежит циклу C и отлично от ребра $r = (z_2, z_1)$. Все отличные от r ребра цикла C содержатся в D, следовательно, ребро r_1 принадлежит D. Обозначим через T граф, полученный из D удалением ребра r_1. Ребро r_1, принадлежащее циклу C, не является мостом (см. 3.4.10), поэтому граф T связный (см. 3.4.6).

Добавим к T ребро $r = (z_1, z_2)$. Это можно сделать, так как вершины z_1 и z_2 принадлежат и графу D, и графу T. Полученный граф обозначим через T_1. Число вершин и число ребер у графа T_1 такие же, как и у графа D, значит, T_1 является деревом (см. 3.7.1). У каждого из ребер r и r_1 один конец принадлежит дереву D_1, а второй нет. Согласно условию среди всех ребер, обладающим таким свойством, ребро r обладает наименьшей длиной. Ввиду этого сумма длин ребер графа T_1 не превосходит суммы длин ребер графа D. Поскольку D — МОД, эти две суммы совпадают и граф T_1 также является МОД. Граф D_1 является поддеревом МОД T_1, что и требовалось. □

3.9.8. *Следующие действия приводят к построению минимального остовного дерева связного нагруженного графа G.*

1. *Выбираем в G ребро наименьшей длины. Обозначаем через G_1 граф, состоящий из этого ребра и его концов.*

2. *Пусть уже построен граф G_k. Просматриваем все ребра, соединяющие вершины этого графа с вершинами, ему не принадлежащими. Выбираем ребро наименьшей длины. Добавляем к G_k это ребро и тот его конец, который не принадлежит G_k. Полученный граф обозначаем через G_{k+1}.*

3. *Работа заканчивается, когда на очередном шаге к строящемуся графу будет добавлена последняя вершина графа G.*

Доказательство. Каждое МОД графа G содержит все его вершины, поэтому любая вершина этого графа является поддеревом каждого МОД.

Пусть среди всех ребер графа G наименьшую длину имеет ребро r. Обозначим через G_0 граф, состоящий из вершины v, принадлежащей ребру r. Граф, состоящий из вершины v, обозначим через G_0. Каждое МОД графа G содержит все его вершины, поэтому G_0 — поддерево некоторого МОД графа G.

Согласно 3.7.3 после добавления к G_0 ребра r вместе с его концом, не принадлежащим G_0, получим поддерево G_1 некоторого МОД. Далее последовательно строим графы G_2, G_3, ..., каждый из которых является поддеревом некоторого МОД также вследствие 3.7.3. Так как граф G конечный, через конечное число шагов будет построено МОД этого графа. □

В графах, у которых ребрам не приписано никаких весов, также можно искать минимальные расстояния между вершинами. Напомним, что длиной маршрута называется число входящих в него ребер (см. 3.4.2). Любая цепь является маршрутом, поэтому длина цепи совпадает с числом ее ребер. Эти определения согласуются с предыдущими, если считать, что в графе вес каждого ребра равен единице.

3.9.9. *Длина кратчайшей цепи, соединяющей две вершины графа, называется* ***расстоянием*** *между этими вершинами. Расстояние от вершины v до v считается равным нулю. Расстояние между вершинами, принадлежащими разным связным компонентам графа, считается равным ∞.*

3.9.10. ***Матрицей расстояний*** *графа G с вершинами $v_1, \ldots, v_n$ называется квадратная матрица порядка n, в которой для $i, j = \overline{1, n}$ элемент i-й строки и j-го столбца равен расстоянию между вершинами v_i и v_j.*

Нетрудно заметить, что эти определения получаются из ранее приведенных, если договориться, что вес каждого ребра графа равен единице. Это означает, в частности, что и алгоритмы нахождения расстояний во взвешенных графах пригодны в рассматриваемом случае.

Глава 4

МАТЕМАТИЧЕСКАЯ ЛОГИКА

4.1. ВЫСКАЗЫВАНИЯ, СВЯЗКИ, ФОРМУЛЫ

Слово «мышление» означает сложный процесс познания явлений окружающей нас действительности. С разных точек зрения мышление изучают представители разных разделов науки: психологи, физиологи, философы и т. д. Слово «логика» происходит от *logos*, означающего *разум*, *рассуждение*. Логика занимается анализом законов, форм и приемов правильного мышления.

Наблюдая некоторое явление или узнав о некотором событии, мы не просто запоминаем это, но и пытаемся установить какие-то связи с тем, что мы уже знаем из жизненного опыта. Основой для установления таких связей служит рассуждение. Рассуждая, мы последовательно переходим от одного суждения к другому, используя некоторые связи, которые принято называть **логическими законами**.

Примеры. 1. По понедельникам студенты экономического факультета должны слушать лекцию профессора Сидорова. Вася — студент экономического факультета. В понедельник он будет слушать лекцию Сидорова.

2. Если олово нагреть до 250 градусов, то оно расплавится. Неизвестный металл нагрели до 250 градусов, и он не расплавился. Этот металл не олово. □

Иногда в рассуждениях используют неправильные умозаключения, лишь по форме напоминающие логические законы, и получают неверные утверждения.

Пример. Когда вода теплая, многие люди купаются в реке. Вася искупался в реке. Значит, вода в реке теплая. □

Свои суждения мы высказываем или записываем в виде повествовательных предложений, которые могут быть истинными или ложными. При изучении логики высказываний нас будут интересовать только эти свойства и мы не будем анализировать строение таких предложений.

4.1.1. *Истинное или ложное повествовательное предложение называется* ***высказыванием****.*

Таким образом, высказыванием мы называем утверждение, смысл которого не вызывает сомнений и о котором можно однозначно сказать, что оно в данный момент является либо истинным, либо ложным.

Примеры. 1. Следующие предложения являются высказываниями.

Волк — плотоядное животное.

Заяц больше лошади.

В Якутии живут мамонты.

От перемены местами слагаемых сумма не меняется.

2. Не являются высказываниями такие предложения.

Сумма двух целых чисел равна пяти.

Мойте руки перед едой.

Как ты думаешь, сдам ли я этот экзамен?

Завтра может пойти дождь, возьми зонтик.

Какой красивый цветок! □

Разумеется, истинность или ложность приведенных выше высказываний обусловлена дополнительными обстоятельствами, которые либо подразумеваются, либо явно указываются. Например, в Якутии когда-то действительно жили мамонты. Однако в приведенном выше высказывании подразумевается, что мамонты там живут сейчас, и потому это высказывание является ложным.

Некоторые высказывания состоят из более коротких высказываний, соединенных связками. Например, высказывание

$$\textit{на обед дают борщ и котлеты} \tag{4.1.1}$$

получено соединением связкой *и* высказываний

$$\textit{на обед дают борщ, на обед дают котлеты.} \tag{4.1.2}$$

Получив борщ и котлеты, вы конечно признаете высказывание (4.1.1) истинным. Однако если вы пришли пообедать и обнаружили, что нет борща или котлет, или того и другого, то наверняка скажете, что вас обманули. Высказывание (4.1.1) является истинным тогда и только тогда, когда истинны оба высказывания (4.1.2).

Из простых высказываний (4.1.2) можно построить и такое высказывание:

$$\textit{на обед дают борщ или котлеты.} \tag{4.1.3}$$

Его можно признать ложным только в том случае, когда ни борща, ни котлет вам не досталось.

Если кто-то вам скажет

неверно, что на обед дают борщ, (4.1.4)

а борщ в обед все же подали, то высказывание (4.1.4) было ложным. В ином случае оно истинное.

Предположим, что вы увидели такое объявление:

если вы покупаете у нас телевизор,
то получаете видеоплеер в подарок. (4.1.5)

Вы спешите в магазин, покупаете телевизор, и неожиданно вам говорят: «Извините, плееры кончились». Несомненно, вы сочтете себя обманутыми, а высказывание (4.1.5) ложным. Истинным оно окажется тогда, когда вместе с телевизором вы получите плеер.

Не будет оснований считать предложение (4.1.5) ложным и в том случае, когда окажется, что телевизоры в магазине кончились и вы остались без телевизора и плеера. Маловероятным является случай, в котором при попытке купить телевизор вам скажут: «Извините, телевизоров больше нет, но, чтобы вы не огорчались, возьмите плеер». Скорее всего, и теперь высказывание (4.1.5) вы сочтете истинным.

Составляя из двух высказываний одно более сложное, мы можем получать высказывания, звучащие очень странно, например:

если Коля проснется рано,
то занятия в университете отменят;
на улице лежит снег или кошка поймала мышь.

Если Коля будет спать долго, то первое высказывание придется признать истинным независимо от того, состоятся занятия или нет. Второе высказывание истинно в случае истинности любого из двух предложений, его составляющих. Все дело в том, что мы исследуем формальные способы образования новых высказываний из уже имеющихся и не проверяем, насколько получившиеся предложения согласуются со здравым смыслом.

Введем для связок следующие обозначения:

$$\text{и} — \&, \quad \text{или} — \vee, \quad \text{если, то} — \supset, \quad \text{не} — \neg.$$

Высказывания (4.1.1), (4.1.3), (4.1.5), (4.1.4) можно записать в виде следующих формул:

$$x\&y, \quad x \vee y, \quad \neg x, \quad x \supset y. \tag{4.1.6}$$

Из них в первых трех буквой x обозначено первое из высказываний в (4.1.2), а буквой y — второе. В последней формуле буквой x обозначено высказывание «Вы покупаете у нас телевизор», а буквой y — высказывание «Вы получаете видеоплеер в подарок».

Как отмечено выше, истинность или ложность получившихся сложных высказываний полностью определяется тем, каким способом они образованы из высказываний x и y, и истинностью или ложностью этих высказываний независимо от их содержания. Переменные x и y в формулах (4.1.6) не обязательно должны обозначать высказывания (4.1.2). Их значениями могут быть любые высказывания. Подобные переменные часто называют **пропозициональными**.

Английское слово *propositional* можно перевести как *относящийся к высказываниям*.

Обозначая через 1 истинное высказывание и через 0 — ложное, зададим соответствующие формулам (4.1.6) функции при помощи табл. 4.1, называемых **таблицами истинности**. Наш выбор обозначе-

Таблица 4.1

x	$\neg x$	x	y	$x\&y$	$x \vee y$	$x \supset y$
0	1	0	0	0	0	1
1	0	0	1	0	1	1
		1	0	0	1	0
		1	1	1	1	1

ний для истинных и ложных высказываний субъективен; встречаются также обозначения Т, И, 0 для истинных и F, Л, 1 для ложных высказываний. Выбранные обозначения удобны тем, что в вычислительной технике принято оперировать с нулями и единицами; к тому же значение конъюнкции оказывается равным произведению значений аргументов. Указанное свойство позволяет называть конъюнкцию **логическим умножением**.

Табл. 4.1 показывает, что сложные высказывания, получающиеся при помощи связок из более простых, являются функциями, определенными на множестве $E_2 = \{0, 1\}$, со значениями в том же множестве. Такие функции называются **операциями на множестве** $\{0, 1\}$.

Раздел математической логики, в котором высказывания изучаются как операции, определенные на множестве из двух элементов, и при

этом используются алгебраические методы, называется **алгеброй логики**.

Связкам $\&$, $\vee$, $\supset$, $\neg$ при помощи таблиц истинности были сопоставлены определенные операции. Перечислим все одноместные и двуместные операции, которые можно задать на множестве E_2.

4.1.2. *На множестве E_2 можно задать только* 4 *одноместные операции:*

$$c_0^1(x) = 0, \quad c_1^1(x) = 1, \quad e_1^1(x) = x, \quad \neg x\,. \qquad \square$$

4.1.3. *На множестве E_2 можно задать лишь* 16 *различных двуместных операций.*

Доказательство. Существует четыре различные пары чисел 0, 1:

$$(0,0), \quad (0,1), \quad (1,0), \quad (1,1)\,.$$

Каждой паре можно сопоставить либо 0, либо 1. По правилу произведения заключаем, что на множестве $\{0,1\}$ всего можно задать $2^4 = 16$ различных операций. $\square$

Все операции, о которых говорится в 4.1.3, перечислены в табл. 4.2 и 4.3, называемых **таблицами истинности**. Некоторым из указанных в этих таблицах функциям мы присвоим особые обозначения и названия, учитывая при этом, что часть из них уже упоминалась в табл. 4.1.

Таблица 4.2

x	y	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

4.1.4. *В табл.* 4.4 *указаны операции из табл.* 4.2 *и* 4.3*, чаще других упоминаемые в дальнейшем, их стандартные обозначения и названия.*

Отметим, что формулу $a \supset b$ можно прочитать как «**из** a **следует** b» или как «a **влечет** b», формулу $a \sim b$ — как «a **эквивалентно** b» или как «a **равнозначно** b», формулу $a \mid b$ — как «a **несовместно с** b», формулу $a \oplus b$ — как «a **неравнозначно** b» или как «a **плюс** b».

Таблица 4.3

x	y	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

Таблица 4.4

Операция	Обозначение	Название
f_1	$c_0^2(x,y)$	константа 0
f_{16}	$c_1^2(x,y)$	константа 1
f_2	$x \ \& \ y$	конъюнкция
f_8	$x \vee y$	дизъюнкция
f_{14}	$x \supset y$	импликация
f_{10}	$x \sim y$	эквивалентность
f_7	$x \oplus y$	сложение по модулю 2
f_{15}	$x \mid y$	штрих Шеффера

4.1.5. *Переменная x_i функции $f(x_1, \ldots, x_n)$ называется* ***существенной****, если найдутся такие числа $\alpha_1, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_n$, что*

$$f(\alpha_1, \ldots, \alpha_{i-1}, 0, \alpha_{i+1}, \ldots, \alpha_n) \neq f(\alpha_1, \ldots, \alpha_{i-1}, 1, \alpha_{i+1}, \ldots, \alpha_n)\,.$$

Несущественные переменные называются ***фиктивными****.*

Если некоторая переменная функции f существенна, то говорят, что f ***существенно зависит*** *от этой переменной.*

Пример. Рассмотрим функцию $f(x,y,z) = (x \vee y)\&(z \vee \neg z)$. Так как

$$f(0,0,0) = 0 \neq 1 = f(1,0,0), \quad f(0,0,0) = 0 \neq 1 = f(0,1,0)\,,$$

переменные x и y существенны. Равенства

$$f(0,0,0) = f(0,0,1) = 0, \quad f(1,0,0) = f(1,0,1) = 1\,,$$

$$f(0,1,0) = f(0,1,1) = 1, \quad f(1,1,0) = f(1,1,1) = 1$$

показывают, что значения функции f не зависят от z, т. е. что переменная z фиктивная. □

Нетрудно заметить, что указанные в табл. 4.2 и 4.3 функции f_4, f_6, f_{13}, f_{11} равны соответственно x, y, $\neg x$, $\neg y$, т. е. существенно зависят только от одного аргумента, а функции c_0^2 и c_1^2 вовсе не зависят от своих аргументов, их аргументы являются фиктивными.

Соединяя несколько высказываний связками, мы можем строить сложные высказывания, например

$$\neg(x\&y) \supset z\,.$$

Это высказывание записано в виде последовательности символов, принадлежащих множеству

$$\{x, y, z, \neg, \&, \supset, (,)\}\,, \tag{4.1.7}$$

т. е. является словом в этом алфавите. Его значение при любых заданных значениях переменных x, y, z можно найти с помощью табл. 4.5.

Таблица 4.5

x	y	z	$x\&y$	$\neg(x\&y)$	$\neg(x\&y) \supset z$
0	0	0	0	1	0
0	0	1	0	1	1
0	1	0	0	1	0
1	0	0	0	1	0
0	1	1	0	1	1
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	0	1

Опытный читатель может записывать таблицы истинности компактней, помещая столбцы значений формул под соответствующими связками. Так, вместо табл. 4.5 можно построить табл. 4.6.

Не всякое слово в алфавите (4.1.7) задает некоторое однозначно определенное высказывание. Например, слово $x\&y \supset z$ можно истолковать как высказывание $(x\&y) \supset z$, а также как высказывание $x\&(y \supset z)$, а слово $xyz\& \supset$ и вовсе трудно как-либо истолковать.

Таблица 4.6

x	y	z	$(x \,\&\, y)$	$\neg$	$\supset z$
0	0	0	0	1	0
0	0	1	0	1	1
0	1	0	0	1	0
1	0	0	0	1	0
0	1	1	0	1	1
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	0	1

Далее мы будем рассматривать только такие слова, которые однозначно описывают какие-либо высказывания. Эти слова называются **формулами**.

4.1.6. *Для построения формул используется алфавит, состоящий из следующих трех групп символов, называемых* ***буквами*** *этого алфавита.*

1. *Строчные латинские буквы с нижними индексами или без индексов, например* x, y, z, x_1, x_2. *Эти буквы называются* ***переменными***.

2. ***Связки*** *— символы, обозначающие операции, определенные на множестве* E_2:

а) *символ одноместной операции* $\neg$;

б) *символы* $\&$, $\vee$, $\supset$, $\sim$, $\oplus$, $|$ *двуместных операций;*

в) *другие символы, вводимые по мере необходимости.*

3. ***Синтаксические символы*** (*(левая скобка),*) *(правая скобка).*

4.1.7. а) *Каждая переменная является формулой.*

б) *Если* $\mathcal{A}$ *и* $\mathcal{B}$ *— формулы, то* $(\mathcal{A} * \mathcal{B})$, *где* $*$ *— одна из двуместных связок, также является формулой.*

в) *Если* $\mathcal{A}$ *— формула, то* $(\neg\mathcal{A})$ *также является формулой.*

Пример. Слово $(\neg((\neg x) \supset x))$ является формулой, а слова

$$(\neg(\neg x \supset x), \quad (x\neg y \& z), \quad ((\& xy) \vee z)$$

формулами не являются. □

Поскольку каждой связке соответствует конкретная операция на множестве E_2, каждая формула также однозначно задает некоторую

операцию, определенную на том же множестве. Чтобы понять, какую именно операцию задает формула, обычно составляют таблицу значений всех меньших формул, из которых данная формула составлена.

Пример. Все значения функции, задаваемой формулой

$$(y \vee (\neg((\neg x) \supset y)))\,,$$

видны из табл. 4.7. □

Таблица 4.7

x	y	$(\neg x)$	$(\neg x \supset y)$	$(\neg((\neg x) \supset y))$	$(y \vee (\neg((\neg x) \supset y)))$
0	0	1	0	1	1
0	1	1	1	0	1
1	0	0	1	0	0
1	1	0	1	0	1

Запись формул можно сделать более простой, если придерживаться следующих соглашений.

4.1.8. *Внешние скобки в формулах можно не писать.*

4.1.9. *Рассмотрим следующую последовательность связок:*

$$\neg, \quad \&, \quad \vee, \quad \supset\,. \tag{4.1.8}$$

Будем говорить, что каждая связка из этой последовательности ***сильнее*** *связки, расположенной правее, но* ***слабее*** *связки, расположенной левее. Слово*

$$\mathcal{A}s_1\mathcal{B}s_2\mathcal{C}\,,$$

в котором $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ *— формулы, а* s_1 *и* s_2 *— двуместные связки из последовательности* (4.1.8), *следует считать краткой записью формулы*

$$(\mathcal{A}s_1\mathcal{B})s_2\mathcal{C}\,,$$

если связка s_1 *сильнее связки* s_2, *и записью формулы*

$$\mathcal{A}s_1(\mathcal{B}s_2\mathcal{C})$$

в противном случае. Слово

$$\neg\mathcal{A}s\mathcal{B}\,,$$

включающее в себя формулы $\mathcal{A}$, $\mathcal{B}$ и двуместную связку s, является сокращенной записью формулы

$$((\neg\mathcal{A})s\mathcal{B})\,.$$

Примеры 1. Слово $x\&y \supset y$ следует считать сокращенной записью формулы $((x\&y) \supset y)$.

2. Пользуясь указанными в 4.1.8 и 4.1.9 правилами, формулу

$$(y \vee (\neg((\neg x) \supset y)))$$

можно записать в виде

$$y \vee \neg(\neg x \supset y)\,. \qquad \square$$

4.2. РАВНОСИЛЬНЫЕ ПРЕОБРАЗОВАНИЯ ФОРМУЛ

4.2.1. *Формулы $\mathcal{A}$ и $\mathcal{B}$, задающие одну и ту же функцию, называются* ***равносильными****. Запись $\mathcal{A} \equiv \mathcal{B}$ означает, что формулы $\mathcal{A}$ и $\mathcal{B}$ равносильны, и называется* ***равносильностью****.*

Отметим, что знак $\equiv$ не является связкой и часто вместо него используют знак равенства. *Мы будем использовать знак равенства в утверждениях о том, что некоторую функцию можно задать определенной формулой.*

4.2.2. *Следующие равносильности являются верными:*

$$x \supset y \equiv \neg x \vee y, \quad \neg(x\&y) \equiv \neg x \vee \neg y, \quad \neg(x \vee y) \equiv \neg x\&\neg y\,.$$

Доказательство. Нетрудно убедиться в справедливости утверждения, придавая переменным x и y все возможные значения и вычисляя каждый раз значения функций, задаваемых формулами справа и слева от знака $\equiv$. Например, можно убедиться в том, что $x \supset y \equiv \neg x \vee y$, составив табл. 4.8. $\square$

Таблица 4.8

x	y	$x \supset y$	$\neg x$	$\neg x \vee y$
0	0	1	1	1
0	1	1	1	1
1	0	0	0	0
1	1	1	0	1

Как уже было сказано ранее в 4.1.6, длинные формулы строятся из более коротких при помощи связок. Эти более короткие формулы часто называют подформулами. Чтобы отчетливо представлять, на какие части можно разделить формулу, вводим следующее определение.

4.2.3. *Если формула представляет собой переменную, то ее подформулой является только она сама.*

Подформулами формулы вида $\neg\mathcal{A}$ *являются формулы* $\neg\mathcal{A}$, $\mathcal{A}$ *и все подформулы формулы* $\mathcal{A}$.

Подформулами формулы вида $\mathcal{A}*\mathcal{B}$, *где* $*$ *— одна из (двуместных) связок, являются формулы* $\mathcal{A}*\mathcal{B}$, $\mathcal{A}$, $\mathcal{B}$ *и все подформулы формул* $\mathcal{A}$ *и* $\mathcal{B}$.

Пример. Формула

$$y \vee (\neg(\neg x \supset y))$$

содержит следующие подформулы:

$$y \vee (\neg(\neg x \supset y)), \quad y, \quad \neg(\neg x \supset y), \quad \neg x \supset y, \quad \neg x, \quad x\,. \qquad \square$$

4.2.4. *Преобразования формул, при которых подформулы этих формул заменяются равносильными формулами, называются* ***равносильными.***

4.2.5. *Если формула* $\mathcal{B}$ *получена из формулы* $\mathcal{A}$ *равносильными преобразованиями, то функция, задаваемая формулой* $\mathcal{A}$, *совпадает с функцией, задаваемой формулой* $\mathcal{B}$.

Доказательство. Совершая равносильные преобразования, мы заменяем подформулу, задающую некоторую функцию, формулой, задающей ту же функцию. $\square$

Пример. Формула

$$\mathcal{B} = (\neg x \vee y)\&(\neg x\&\neg y)$$

получена из формулы

$$\mathcal{A} = (x \supset y)\&\neg(x \vee y)$$

заменой подформул

$$x \supset y, \quad \neg(x \vee y)$$

равносильными им формулами

$$\neg x \vee y, \quad \neg x\&\neg y.$$

Ввиду этого формулы $\mathcal{A}$ и $\mathcal{B}$ равносильны, а определяемые ими функции совпадают. $\square$

4.2.6. *Формула называется* ***тождественно истинной (общезначимой, тавтологией)****, если при любых значениях содержащихся в ней переменных ее значение равно* 1*. Формула называется* ***тождественно ложной****, если при любых значениях содержащихся в ней переменных ее значение равно* 0*.*

Термин «тавтология» образован из греческих слов *tauto* — то же самое и *logos* — слово. Каждая тождественно истинная формула выражает некоторый логический закон.

Примеры. Нетрудно убедиться в том, что формула $x \vee \neg x$ является тождественно истинной, а формула

$$\neg((x \supset y) \supset (\neg x \vee y))$$

— тождественно ложной. Формула

$$(x \vee y) \supset (\neg x \& \neg y)$$

при $x = 1$, $y = 1$ принимает значение 0, а при $x = 0$, $y = 0$ принимает значение 1, поэтому она не является ни тождественно истинной, ни тождественно ложной. □

4.2.7. *В случаях, когда это не вызывает недоразумений, тождественно истинную формулу мы можем обозначать через* 1*, а тождественно ложную формулу — через* 0*.*

Выяснить, является ли формула тождественно истинной или тождественно ложной, всегда возможно, и даже различными способами. Один способ очевиден — составить таблицу истинности для этой формулы. Другой способ заключается в преобразовании формулы к некоторому специальному виду и будет описан позднее. Сейчас рассмотрим лишь два простых случая.

4.2.8. 1) $x \& \neg x \equiv 0$ *(закон противоречия);*
2) $x \vee \neg x \equiv 1$ *(закон исключенного третьего).*

Доказательство. Справедливость утверждений легко проверить, составив таблицы истинности для формул, находящихся слева от знака $\equiv$. □

Следующие соотношения позволяют упрощать формулы, содержащие тождественно истинные или тождественно ложные подформулы.

4.2.9. 1) $x \& 1 \equiv x$;
2) $x \vee 1 \equiv 1$;
3) $x \& 0 \equiv 0$;
4) $x \vee 0 \equiv x$.

Доказательство. Утверждения непосредственно следуют из определений функций $x\&y$ и $x \vee y$. Например, если $y = 1$, то значение функции $x\&y$ для каждого x совпадает с x, а при $y = 0$ эта функция не может быть равной 1. □

В 4.2.2 показано, что формулу, содержащую связку $\supset$, можно заменить равносильной формулой со связками $\vee$, $\neg$. Следующие соотношения показывают, что в формулах можно избавиться от связок $\oplus$, $\sim$ и $|$, воспользовавшись связками $\&$, $\vee$, $\neg$.

4.2.10. 1) $x \sim y \equiv (x\&y) \vee (\neg x\&\neg y)$;
2) $x \mid y \equiv \neg(x\&y)$;
3) $x + y \equiv \neg(x \sim y) \equiv (\neg x \vee \neg y)\&(x \vee y)$.

Доказательство. Каждый из этих законов можно проверить, вычисляя значения функций, задаваемых формулами, стоящими справа и слева от знака $\equiv$. □

4.2.11. *Функции $\&$, $\vee$ обладают свойствами*
1) *коммутативности:*

$$x\&y \equiv y\&x, \quad x \vee y \equiv y \vee x;$$

2) *ассоциативности:*

$$(x\&y)\&z \equiv x\&(y\&z),$$

$$(x \vee y) \vee z \equiv x \vee (y \vee z);$$

3) *поглощения:*

$$x\&(x \vee y) \equiv x, \quad x \vee (x\&y) \equiv x;$$

4) *дистрибутивности:*

$$x\&(y \vee z) \equiv (x\&y) \vee (x\&z),$$

$$x \vee (y\&z) \equiv (x \vee y)\&(x \vee z);$$

5) $x\&x \equiv x, \quad x \vee x \equiv x$.
Функция $\neg$ подчиняется ***закону снятия двойного отрицания:***

$$\neg\neg x \equiv x.$$

Доказательство аналогично 4.2.10. □

4.3. НОРМАЛЬНЫЕ ФОРМЫ

Ранее были сформулированы некоторые правила, позволяющие несколько упростить запись формул. Записи будут еще проще, если придерживаться следующих соглашений.

4.3.1. *Введем следующее обозначение:*

$$x_i^{\alpha_i} = \begin{cases} x_i, & \text{если } \alpha_i = 1, \\ \neg x_i, & \text{если } \alpha_i = 0. \end{cases}$$

Пример. Формула

$$x\&\neg y\&z\&\neg u \vee \neg x\&\neg z$$

может быть записана также в виде

$$x^1\&y^0\&z^1\&u^0 \vee x^0\&z^0 . \qquad \square$$

4.3.2. *Связку $\&$ в случаях, когда это не может вызвать недоразумений, будем заменять точкой или вовсе опускать.*

Пример. Формулу

$$x\&\neg y\&z\&\neg u \vee \neg x\&\neg y\&\neg z\&u$$

можно теперь записать в виде слова $x\neg yz\neg u \vee \neg x\neg y\neg zu$, а также в виде слова $x^1y^0z^1u^0 \vee x^0y^0z^0u^1$. $\square$

4.3.3. *Элементарной конъюнкцией или конъюнктом и элементарной дизъюнкцией или дизъюнктом называются формулы, имеющие соответственно следующий вид:*

$$\mathop{\&}_{i=1}^{n} x_i^{\alpha_i} = x_1^{\alpha_1}\&x_2^{\alpha_2}\&\ldots\&x_n^{\alpha_n} = x_1^{\alpha_1}x_2^{\alpha_2}\ldots x_n^{\alpha_n} , \tag{4.3.1}$$

$$\bigvee_{i=1}^{n} x_i^{\alpha_i} = x_1^{\alpha_1} \vee x_2^{\alpha_2} \ldots \vee x_n^{\alpha_n} . \tag{4.3.2}$$

Отметим, что в формулах (4.3.1) и (4.3.2) переменные x_i и x_j могут совпадать даже при $i \neq j$. Иными словами, в элементарных конъюнкциях и дизъюнкциях одна и та же переменная может встречаться несколько раз как с отрицанием, так и без него.

4.3.4. *Литералом называется формула, представляющая собой либо переменную, либо отрицание переменной.*

Теперь можно сказать, что конъюнктом называется конъюнкция литералов, а дизъюнктом — дизъюнкция литералов.

Примеры. Формула

$$x^1y^0z^1u^0 = x\neg yz\neg u$$

является элементарной конъюнкцией (конъюнктом), а формула

$$x^1 \vee x^0 \vee y^1 \vee y^0 = x \vee \neg x \vee y \vee \neg y$$

— элементарной дизъюнкцией (дизъюнктом). □

4.3.5. ***Дизъюнктивной нормальной формой*** (ДНФ) *называется формула, имеющая вид дизъюнкции элементарных конъюнкций.*

Пример. Дизъюнктивной нормальной формой является формула

$$x^1y^0z^1u^0 \vee x^0y^0z^1u^0 \vee x^0y^0z^0u^1 = x\neg yz\neg u \vee \neg x\neg yz\neg u \vee \neg x\neg y\neg zu\,. \quad \square$$

4.3.6. ***Конъюнктивной нормальной формой*** (КНФ) *называется формула, являющаяся конъюнкцией элементарных дизъюнкций.*

Пример. В конъюнктивной нормальной форме находится формула

$$(x \vee \neg y \vee z \vee \neg u)\&(\neg x \vee \neg y \vee z \vee \neg u)\&(\neg x \vee \neg y \vee \neg z \vee u)\,. \quad \square$$

Обычно предполагается, что входящие в ДНФ (КНФ) элементарные конъюнкции (дизъюнкции) попарно различны.

4.3.7. *Любую формулу можно равносильными преобразованиями привести к дизъюнктивной или конъюнктивной нормальной форме.*

Доказательство. Достаточно произвести следующие действия.

1. Пользуясь равносильностями, указанными в 4.2.2 и 4.2.10, заменяем все подформулы, содержащие связки $\supset$, $|$, $\oplus$, формулами, содержащими связки $\&$, $\vee$, $\neg$.

2. Используя 4.2.2, последовательно заменяем подформулы вида $\neg(\mathcal{A}\&\mathcal{B})$ на $\neg\mathcal{A} \vee \neg\mathcal{B}$ и подформулы вида $\neg(\mathcal{A} \vee \mathcal{B})$ на $\neg\mathcal{A}\&\neg\mathcal{B}$.

3. Выполнив указанные выше действия, мы получим формулу, обладающую следующим свойством: если какая-то ее подформула имеет вид $\neg\mathcal{A}$, то формула $\mathcal{A}$ не содержит символов $\&$ и $\vee$. Теперь последовательно заменим все подформулы вида $\neg\neg\mathcal{B}$ на $\mathcal{B}$ (используется закон снятия двойного отрицания, см. 4.2.11).

4. Пользуясь свойствами дистрибутивности и ассоциативности (см. 4.2.11), получаем ДНФ или КНФ.

Пример.

$$\begin{aligned}
&\neg((x \vee \neg y) \supset (x \supset y)) \vee (x\&y) \equiv \\
&\equiv \neg((x \vee \neg y) \supset (\neg x \vee y)) \vee (x\&y) \equiv \\
&\equiv \neg(\neg(x \vee \neg y) \vee (\neg x \vee y)) \vee (x\&y) \equiv \\
&\equiv (\neg\neg(x \vee \neg y)\&\neg(\neg x \vee y)) \vee (x\&y) \equiv \\
&\equiv (\neg\neg(x \vee \neg y)\&(\neg\neg x\&\neg y)) \vee (x\&y) \equiv \\
&\equiv ((x \vee \neg y)\&(x\&\neg y)) \vee (x\&y) \equiv \\
&\equiv x\&x\&\neg y \vee \neg y\&x\&\neg y \vee x\&y \equiv \\
&\equiv x\&\neg y \vee x\&\neg y \vee x\&y \equiv \\
&\equiv x\&\neg y \vee x\&y.
\end{aligned}$$

□

4.3.8. *Для того чтобы* КНФ (ДНФ) *была тождественно истинной (тождественно ложной), необходимо и достаточно, чтобы каждая входящая в нее элементарная дизъюнкция (конъюнкция) содержала некоторую переменную вместе с ее отрицанием.*

Доказательство. ($\Leftarrow$) Из 4.2.8 и 4.2.9 следует, что если элементарная дизъюнкция содержит как переменную, так и ее отрицание, то она является тождественно истинной. Видим, что рассматриваемая КНФ является конъюнкцией тождественно истинных формул, следовательно, она также тождественно истинна.

($\Rightarrow$) Предположим, что одна из входящих в КНФ элементарных дизъюнкций не содержит никакой переменной вместе с ее отрицанием. Пусть эта элементарная дизъюнкция имеет вид

$$\bigvee_{i=1}^{n} x_i^{\alpha_i} = x_1^{\alpha_1} \vee x_2^{\alpha_2} \ldots \vee x_n^{\alpha_n}. \tag{4.3.3}$$

Положим $x_1 = 1-\alpha_1, \ldots, x_n = 1-\alpha_n$, после чего получим конъюнкцию

$$(1-\alpha_1)^{\alpha_1} \vee (1-\alpha_2)^{\alpha_2} \ldots \vee (1-\alpha_n)^{\alpha_n}. \tag{4.3.4}$$

Нетрудно заметить, что для $i = \overline{1,n}$

$$(1-\alpha_i)^{\alpha_i} = \begin{cases} 0, & \text{если } \alpha_i = 1, \\ \neg 1, & \text{если } \alpha_i = 0. \end{cases}$$

Это означает, что значением элементарной дизъюнкции (4.3.3) при указанных значениях переменных является 0. Например, элементарная дизъюнкция

$$\mathcal{A} = x \vee \neg y \vee z \vee \neg u = x^1 \vee y^0 \vee z^1 \vee u^0$$

не содержит никакой переменной вместе с ее отрицанием. При

$$x = 1 - 1 = 0, \quad y = 1 - 0 = 1, \quad z = 1 - 1 = 0, \quad u = 1 - 0 = 1$$

формула $\mathcal{A}$ имеет следующее значение:

$$0 \vee \neg 1 \vee 0 \vee \neg 1 = 0 \vee 0 \vee 0 \vee 0 = 0\,.$$

КНФ представляет собой конъюнкцию элементарных дизъюнкций. Если значение одной из этих элементарных дизъюнкций равно нулю, то и значение всей формулы равно нулю.

Доказательство. ($\Leftarrow$) утверждения об ДНФ проводится аналогично. □

Пример. Тождественно истинна КНФ

$$(x \vee \neg x \vee y \vee \neg z)\&(\neg y \vee z \vee \neg z \vee u)\,.$$

Не является тождественно истинной КНФ

$$(\neg x \vee y \vee \neg z)\&(\neg y \vee \neg z \vee u)\,. \qquad \square$$

4.3.9. ДНФ (КНФ) *называется* ***совершенной****, если каждая переменная формулы входит в каждую элементарную конъюнкцию (дизъюнкцию) ровно один раз.*

Примеры. Не являются совершенными КНФ из предыдущего примера. Совершенна КНФ $(\neg x \vee y \vee \neg z \vee u)\&(x \vee \neg y \vee \neg z \vee u)$. □

4.3.10. *Операции на множестве $E_2 = \{0, 1\}$ называются* ***функциями алгебры логики****, или* ***булевыми функциями****.*

Булевыми функции названы в честь английского математика Дж. Буля (середина XIX в.).

Докажем, что с помощью операций конъюнкции, дизъюнкции и отрицания можно задать любую функцию алгебры логики.

4.3.11. *Если не все значения булевой функции равны нулю, то ее можно представить совершенной дизъюнктивной нормальной формой.*

Доказательство. Рассмотрим функцию $f(x_1, \ldots, x_n)$, принимающую хотя бы один раз значение 1. Нетрудно проверить, что формула

$$x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n}$$

принимает значение 1 тогда и только тогда, когда

$$x_1 = \alpha_1, \quad x_2 = \alpha_2, \ \ldots, \ x_n = \alpha_n \,,$$

формула

$$x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n} \vee x_1^{\beta_1} x_2^{\beta_2} \ldots x_n^{\beta_n}$$

принимает значение 1 тогда и только тогда, когда либо

$$x_1 = \alpha_1, \quad x_2 = \alpha_2, \ \ldots, \ x_n = \alpha_n,$$

либо

$$x_1 = \beta_1, \quad x_2 = \beta_2, \ \ldots, \ x_n = \beta_n$$

и т. д. Формула

$$\bigvee_{\substack{(\alpha_1,\ldots,\alpha_n),\\ f(\alpha_1,\ldots,\alpha_n)=1}} x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n} \tag{4.3.5}$$

принимает значение 1 в точности тогда, когда равна единице функция f. □

4.3.12. *Формула* (4.3.5) *называется* ***совершенной дизъюнктивной нормальной формой*** (СДНФ) *функции* f.

Таблица 4.9

x	y	z	$f(x,y,z)$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Пример. Пусть функция f задана табл. 4.9. Она равна единице на наборах $(0,0,1)$, $(0,1,0)$, $(1,1,1)$. Записываем ее СДНФ:

$$f(x_1,\ldots,x_n) = x^0 y^0 z^1 \vee x^0 y^1 z^0 \vee x^1 y^1 z^1 = \\ = \neg x \neg y z \vee \neg x y \neg z \vee x y z \,.$$ □

4.3.13. *Если не все значения булевой функции f равны единице, то ее можно представить совершенной конъюнктивной нормальной формой:*

$$f(x_1,\dots,x_n)=\mathop{\&}_{\substack{(\alpha_1,\dots,\alpha_n),\\ f(\alpha_1,\dots,\alpha_n)=0}}(x_1^{1-\alpha_1}\vee x_2^{1-\alpha_2}\vee\dots\vee x_n^{1-\alpha_n}).$$

Доказательство проводится аналогично доказательству утверждения 4.3.11. □

Таблица 4.10

x	y	z	$f(x,y,z)$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Пример. Пусть функция f задана табл. 4.10. Она равна нулю на наборах $(0,1,1)$, $(1,0,0)$, $(1,1,0)$. Записываем ее СКНФ:

$$f(x_1,\dots,x_n)=(x^1\vee y^0\vee z^0)\&(x^0\vee y^1\vee z^1)\&(x^0\vee y^0\vee z^1)=$$
$$=(x\vee\neg y\vee\neg z)\&(\neg x\vee y\vee z)\&(\neg x\vee\neg y\vee z).$$ □

4.3.14. *Не содержащая отрицаний формула, представляющая собой либо константу, равную* 0 *или* 1, *либо сумму элементарных конъюнкций с такой константой, называется* ***полиномом Жегалкина****.*

Полиномом, т. е. многочленом, формула указанного вида называется потому, что представляет собой сумму членов, а каждый член является произведением переменных, так как на множестве $\{0,1\}$ произведение xy совпадает с конъюнкцией $x\&y$.

Пример. Формула $y\oplus xy\oplus xz\oplus xyz\oplus 1$ представляет собой полином Жегалкина. □

4.3.15. *Каждая булева функция может быть представлена полиномом Жегалкина.*

Доказательство. Если функция f не равна тождественно нулю, то ее можно представить СДНФ:

$$f(x_1,\dots,x_n)=\bigvee_{\substack{(\alpha_1,\dots,\alpha_n),\\ f(\alpha_1,\dots,\alpha_n)=1}} x_1^{\alpha_1}x_2^{\alpha_2}\dots x_n^{\alpha_n}\,. \tag{4.3.6}$$

Какие бы мы ни выбрали значения переменных, из всех содержащихся в СДНФ элементарных конъюнкций лишь одна может быть равна единице, а остальные равны нулю. Заметим, что в таком случае конъюнкция не отличается от сложения. Например, рассмотрим случай, когда переменных всего две:

$$0\vee 0=0\oplus 0=0,\quad 0\vee 1=1\vee 0=0\oplus 1=1\oplus 0=1\,.$$

Это означает, что в формуле (4.3.6) знаки дизъюнкции можно заменить символами сложения. Поскольку $\neg x=x\oplus 1$, то все отрицания переменных в полученной формуле можно заменить суммой переменной с единицей. Возникает формула, в которой используются символы переменных, конъюнкции, сложения и скобки. Поскольку на множестве E_2 конъюнкция не отличается от умножения, можно воспользоваться обычными свойствами сложения и умножения и преобразовать формулу к виду, указанному в 4.3.14.

Если же функция $f(x_1,\dots,x_n)$ тождественно равна нулю, то она представима в виде 0. □

В случае, когда функция задана формулой, отличной от СДНФ, можно либо привести ее к ДНФ, а затем поступить так, как указано выше, либо напрямую воспользоваться равносильностями, позволяющими избавляться от связок, отличных от $\oplus$, $\&$.

Полезно также заметить, что

$$g(x_1,\dots,x_n)\oplus g(x_1,\dots,x_n)\equiv 0$$

для любой булевой функции g. Отсюда следует, что если в сумме нечетное число одинаковых слагаемых, то остается одно из них, а если четное, то все исчезают.

Примеры. 1. Представим многочленом Жегалкина функцию

$$x\neg y\neg z\vee\neg xyz\,.$$

Данная формула находится в СДНФ, поэтому можно сразу избавиться от знаков дизъюнкции и отрицания:

$$\begin{aligned} x\neg y\neg z \vee \neg xyz &\equiv x(y\oplus 1)(z\oplus 1)\oplus (x\oplus 1)yz \equiv \\ &\equiv (xy\oplus x)(z\oplus 1)\oplus (xyz\oplus yz) \equiv \\ &\equiv xyz\oplus xz\oplus xy\oplus x\oplus xyz\oplus yz \equiv \\ &\equiv x\oplus xy\oplus xz\oplus yz\,. \end{aligned}$$

2. Представим многочленом Жегалкина функцию

$$x\neg y \vee y\neg z \vee \neg xz\,.$$

Формула не находится в СДНФ, но ее легко преобразовать к нужному виду:

$$\begin{aligned} &x\neg y \vee y\neg z \vee \neg xz \equiv \\ &\equiv x\neg yz \vee x\neg y\neg z \vee xy\neg z \vee \neg xy\neg z \vee \neg xyz \vee \neg x\neg yz \equiv \\ &\equiv x\neg yz \oplus x\neg y\neg z \oplus xy\neg z \oplus \neg xy\neg z \oplus \neg xyz \vee \neg x\neg yz \equiv \\ &\equiv x(y\oplus 1)z \oplus x(y\oplus 1)(z\oplus 1)\oplus xy(z\oplus 1)\oplus \\ &\quad \oplus (x\oplus 1)y(z\oplus 1)\oplus (x\oplus 1)yz\oplus (x\oplus 1)(y\oplus 1)z \equiv \\ &\equiv xyz\oplus xz\oplus xyz\oplus xy\oplus xz\oplus x\oplus xyz\oplus xy\oplus \\ &\quad \oplus xyz\oplus xy\oplus yz\oplus y\oplus xyz\oplus yz\oplus xyz\oplus xz\oplus yz\oplus z \equiv \\ &\equiv x\oplus y\oplus z\oplus xy\oplus xz\oplus yz\,. \end{aligned}$$

□

4.4. ЗАМКНУТЫЕ КЛАССЫ ФУНКЦИЙ

Следующее утверждение очевидно.

4.4.1. *Слово, полученное из формулы F_1 заменой всех вхождений какой-либо из ее переменных формулой F_2, также является формулой.* □

Согласно 4.1.7 переменные являются формулами, поэтому из формулы $x\&y$ по указанному правилу мы получаем, например, формулы

$$x\&x, \quad y\&y, \quad z\&z.$$

Эти случаи, когда переменная в формуле заменяется переменной, выделим особо, назвав **отождествлением** и **переименованием** переменных.

4.4.2. *Пусть формулы F_1, F_2 и F_3 задают функции f_1, f_2 и f_3 соответственно и формула F_3 получена из F_1 заменой всех вхождений*

некоторой переменной формулой F_2. Будем говорить, что функция f_3 получена ***подстановкой*** *из функций f_1 и f_2. То, что функция f получена из функции $g(x_1,\dots,x_n)$ подстановкой вместо переменной x_i функции $h(y_1,\dots,y_m)$, записывается следующим образом:*

$$f(x_1,\dots,x_{i-1},x_{i+1},\dots,x_n,y_1,\dots,y_m) = \\ = g(x_1,\dots,x_{i-1},h(y_1,\dots,y_m),x_{i+1},\dots,x_n)\,. \qquad (4.4.1)$$

4.4.3. *Выражение* (4.4.1) *показывает, что множество связок, используемых при построении формул, задающих функции g и h, совпадает с множеством связок, используемых при построении формулы, задающей функцию f.* □

Заметим, что множества переменных функций f и g в (4.4.1) могут иметь общие элементы.

4.4.4. *Совокупность всех булевых функций, т. е. функций, определенных на множестве чисел $E_2 = \{0,1\}$ и принимающих значения в том же множестве, будет обозначаться через $\mathbf{P}_2$.*

4.4.5. *Класс $\mathbf{K}$ функций алгебры логики назовем* ***замкнутым****, если любая функция, полученная либо подстановкой из двух функций, принадлежащих $\mathbf{K}$, либо отождествлением переменных, либо переименованием переменных из функции, принадлежащей $\mathbf{K}$, снова принадлежит $\mathbf{K}$.*

Укажем некоторые примеры замкнутых классов.

4.4.6. *Говорят, что булева функция $f(x_1,\dots,x_n)$* ***сохраняет нуль****, если выполняется равенство $f(0,\dots,0) = 0$, и* ***сохраняет единицу****, если $f(1,\dots,1) = 1$.*

Примеры. Функции $x\&y$ и $x\vee y$ сохраняют как нуль, так и единицу.
Функция $x \supset y$ сохраняет единицу, но не сохраняет нуль.
Функция $\neg x$ не сохраняет ни нуль, ни единицу. □

4.4.7. *Множество $\mathbf{T}_0$ всех булевых функций, сохраняющих нуль, образует замкнутый класс.*

Множество $\mathbf{T}_1$ всех булевых функций, сохраняющих единицу, также является замкнутым классом.

Доказательство. Пусть функция $f(x_1,\dots,x_n)$ сохраняет нуль и g — функция, полученная из функции f отождествлением и переименованием переменных. Последнее означает, что какие-то переменные

функции f получили новые обозначения, причем некоторые из них обозначены одинаково, хотя и были разными. Это можно записать так:

$$g(x_{i_1},\dots,x_{i_k}) = f(x_{i_1},\dots,x_{i_k})\,.$$

Тогда, очевидно,

$$g(0,\dots,0) = f(0,\dots,0) = 0\,,$$

функция g также сохраняет нуль.

Возьмем теперь функции $p(x_1,\dots,x_n)$ и $q(y_1,\dots,y_m)$, сохраняющие нуль, и положим

$$h(x_1,\dots,x_{i-1},x_{i+1},\dots,x_n,y_1,\dots,y_m) = \\ = p(x_1,\dots,x_{i-1},q(y_1,\dots,y_m),x_{i+1},\dots,x_n)\,.$$

Так как $q(0,\dots,0) = 0$ и $p(0,\dots,0) = 0$, то

$$h(0,\dots,0) = p(0,\dots,0,q(0,\dots,0),0,\dots,0) = 0\,.$$

Видим, что и функция h сохраняет нуль.

Второе утверждение доказывается аналогично. □

4.4.8. *Функция* $\neg(f(\neg x_1,\dots,\neg x_n))$ *называется* ***двойственной*** *к булевой функции* $f(x_1,\dots,x_n)$. *Обозначение:* $f^*(x_1,\dots,x_n)$.

Пример. Двойственной к функции $x\&y$ является функция $\neg(\neg x\&\neg y)$, т. е. функция $x\vee y$. □

4.4.9. *Пусть булева функция* f *задана таблицей истинности. Если в этой таблице все нули заменить единицами, а единицы — нулями, то получим таблицу истинности двойственной функции* f^*.

Доказательство. Рассмотрим одну строку из таблицы истинности функции f:

$$(\alpha_1,\dots,\alpha_n,\gamma),\quad \gamma = f(\alpha_1,\dots,\alpha_n)\,.$$

Заменяя нули единицами, а единицы — нулями, получим строку

$$(\neg\alpha_1,\dots,\neg\alpha_n,\neg\gamma)\,. \tag{4.4.2}$$

Так как

$$f^*(\neg\alpha_1,\dots,\neg\alpha_n) = \neg f(\neg\neg\alpha_1,\dots,\neg\neg\alpha_n) = \neg f(\alpha_1,\dots,\alpha_n) = \neg\gamma\,,$$

то строка (4.4.2) принадлежит таблице истинности функции f^*.

Повторяя это рассуждение для каждой строки из таблицы истинности функции f, докажем нужное утверждение. □

4.4.10. *Назовем наборы* $(\alpha_1, \ldots, \alpha_n)$ *и* $(\beta_1, \ldots, \beta_n)$ *противоположными, если* $\beta_i = \neg\alpha_i$, $i = \overline{1,n}$.

4.4.11. *Пусть таблица истинности булевой функции* f *устроена так, что противоположными являются первый и последний наборы переменных, второй и предпоследний и т. д. Если в столбце значений функции* f *все нули заменить единицами, а единицы — нулями, и переставить элементы столбца в обратном порядке, то получим таблицу истинности двойственной функции* f^*. □

4.4.12. *Если* $f^* = f$, *то булева функция* f *называется* ***самодвойственной.***

Пример. Самодвойственной является функция $x \oplus y \oplus z$. Действительно,

$$\neg(\neg x \oplus \neg y \oplus \neg z) = ((x \oplus 1) \oplus (y \oplus 1) \oplus (z \oplus 1)) \oplus 1 = \\ = x \oplus y \oplus z \oplus 1 \oplus 1 \oplus 1 \oplus 1 = x \oplus y \oplus z\,. \qquad \square$$

4.4.13. *Если функция* $f(x_1, \ldots, x_n)$ *самодвойственная, то*

$$f(x_1, \ldots, x_n) \neq f(\neg x_1, \ldots, \neg x_n)\,.$$

Доказательство. Согласно 4.4.12

$$f(x_1, \ldots, x_n) = \neg f(\neg x_1, \ldots, \neg x_n) \neq f(\neg x_1, \ldots, \neg x_n)\,. \qquad \square$$

4.4.14. *Пусть булева функция на* m *наборах значений переменных принимает значение* 0 *и на* n *наборах —* 1. *Если* $m \neq n$, *то эта функция не самодвойственная.*

Доказательство. Предположим, что самодвойственная функция $f(x_1, \ldots, x_k)$ равна единице на наборах

$$\sigma_1 = (\alpha_{11}, \ldots, \alpha_{1n}), \ldots, \sigma_m = (\alpha_{m1}, \ldots, \alpha_{mn})\,,$$

в остальных n случаях равна нулю и $m < n$. Ввиду 4.4.13 на наборах

$$\bar{\sigma}_1 = (\neg\alpha_{11}, \ldots, \neg\alpha_{1n}), \ldots, \bar{\sigma}_m = (\neg\alpha_{m1}, \ldots, \neg\alpha_{mn})$$

функция f равна нулю. Так как $2m < m + n$, есть еще наборы

$$\delta_1 = (\beta_{11}, \ldots, \beta_{1n}), \ldots, \delta_s = (\beta_{s1}, \ldots, \beta_{sn})\,,$$

на которых функция f равна нулю. Очевидно, набор $\bar{\delta}_1 = (\neg\beta_{11}, \ldots, \neg\beta_{1n})$ принадлежит множеству $\delta_1, \ldots, \delta_s$ и потому

$$f(\beta_{11}, \ldots, \beta_{1n}) = f(\neg\beta_{11}, \ldots, \neg\beta_{1n}),$$

что противоречит 4.4.13.

Случай $m > n$ аналогичен рассмотренному. □

Равенство числа единиц функции числу нулей является необходимым признаком самодвойственности функции, но не достаточным. Дальнейшую проверку можно производить, сравнивая значения функции на противоположных наборах значений переменных. Согласно 4.4.13 эти значения не должны совпадать.

Примеры. 1. Самодвойственной является функция, заданная табл. 4.11.

Таблица 4.11

x	y	z	$f(x, y, z)$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

2. Функция $f(x, y, z)$, равная нулю на наборах

$$(0, 1, 1), \quad (1, 0, 1), \quad (1, 1, 1)$$

и единице в остальных случаях, не самодвойственная, так как принимает три раза значение 0 и пять раз значение 1. □

4.4.15. *Множество* **S** *всех булевых самодвойственных функций образует замкнутый класс.*

Доказательство. Это утверждение доказывается так же, как и 4.4.7. □

4.4.16. *Будем, как обычно, считать, что нуль меньше единицы. Булева функция* $f(x_1, \ldots, x_n)$ *называется* ***монотонной****, если из* $x_i \le y_i$, $i = \overline{1,n}$, *следует, что*

$$f(x_1, \ldots, x_n) \le f(y_1, \ldots, y_n)\,.$$

Другими словами, функция является монотонной тогда и только тогда, когда она монотонно не убывает по каждому из своих аргументов.

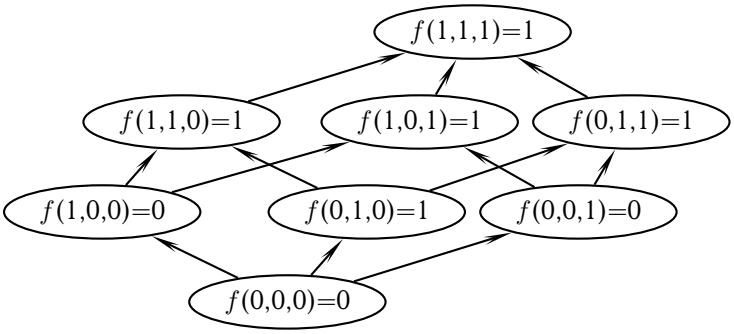


Рис. 4.1

Пример. Монотонными являются элементарные дизъюнкции и элементарные конъюнкции, не содержащие отрицаний, т. е. имеющие вид

$$x_1 \vee \ldots \vee x_k\,, \quad x_1 \& \ldots \& x_k\,.$$

Функции с тремя переменными удобно проверять на монотонность, нарисовав диаграмму. На рис. 4.1 изображена диаграмма монотонной функции. Путь из каждой вершины графа к вершине $f(1,1,1) = 1$, в которой функция равна единице, проходит только через вершины, в которых функция также равна единице. Функция на рис. 4.2 не монотонна, так как путь из вершины с пометкой $f(1,0,0) = 1$ в вершину $f(1,1,1) = 1$ содержит вершину с пометкой $f(1,1,0) = 0$. □

4.4.17. *Класс* $\mathbf{M}$ *всех булевых монотонных функций замкнут.*

Доказательство. Данное утверждение доказывается так же, как и 4.4.7. □

4.4.18. *Если функция* $f(x_1, \ldots, x_n)$ *не монотонная, то найдутся такие наборы*

$$\bar{u} = (\alpha_1, \ldots, \alpha_{i-1}, 0, \alpha_{i+1}, \ldots, \alpha_n)\,,$$

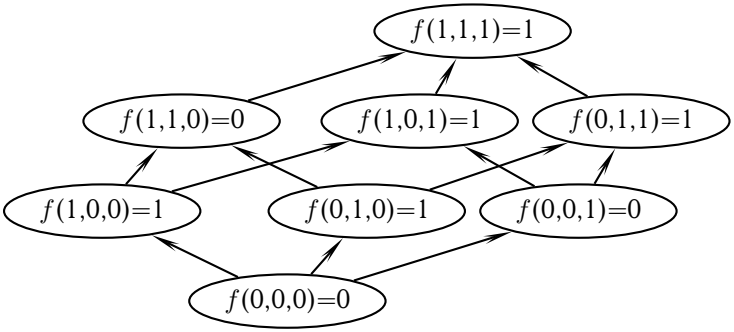


Рис. 4.2

$$\bar{v} = (\alpha_1, \ldots, \alpha_{i-1}, 1, \alpha_{i+1}, \ldots, \alpha_n)$$

из нулей и единиц, что

$$f(\alpha_1, \ldots, \alpha_{i-1}, 0, \alpha_{i+1}, \ldots, \alpha_n) = 1\,,$$

$$f(\alpha_1, \ldots, \alpha_{i-1}, 1, \alpha_{i+1}, \ldots, \alpha_n) = 0\,.$$

Доказательство. Поскольку функция f не монотонная, найдутся такие числа $x_1, \ldots, x_n, y_1, \ldots, y_n$, что $x_i \le y_i$, $i = \overline{1,n}$, и

$$f(x_1, \ldots, x_n) > f(y_1, \ldots, y_n)\,.$$

Введем обозначения

$$\bar{p} = (x_1, \ldots, x_n), \quad \bar{q} = (y_1, \ldots, y_n)\,.$$

Из $x_i \le y_i$, $i = \overline{1,n}$, следует, что если в некотором разряде набора $\bar{p}$ находится единица, то в том же разряде набора $\bar{q}$ также стоит единица. По той же причине если в некотором разряде набора $\bar{q}$ имеется нуль, то в том же разряде набора $\bar{p}$ также стоит нуль. Заменяя последовательно нули единицами в тех разрядах набора $\bar{p}$, где у $\bar{q}$ стоят единицы, получим цепочку наборов $\bar{p}$, $\bar{p}_1$, $\ldots$, $\bar{p}_{m-1}$, $\bar{p}_m = \bar{q}$ и последовательность значений

$$f(\bar{p}),\ f(\bar{p}_1),\ \ldots,\ f(\bar{p}_{m-1}),\ f(\bar{p}_m)\,.$$

Поскольку $f(\bar{p}) = 1$ и $f(\bar{p}_m) = f(\bar{q}) = 0$, найдется такое $j \in \{0, \ldots \ldots, m-1\}$, что $f(\bar{p}_j) = 1$ и $f(\bar{p}_{j+1}) = 0$. Полагая $\bar{u} = \bar{p}_j$ и $\bar{v} = \bar{p}_{j+1}$, завершим доказательство. $\square$

4.4.19. *Булева функция* $f(x_1, \ldots, x_n)$ *называется* ***линейной****, если ее можно представить в виде*

$$f(x_1, \ldots, x_n) = \alpha_0 \oplus \alpha_1 x_1 \oplus \alpha_2 x_2 \oplus \ldots \oplus \alpha_n x_n \,, \tag{4.4.3}$$

где $\alpha_i \in \{0, 1\}$, $i = \overline{0, n}$.

Так как $\alpha_i \in \{0, 1\}$, линейная функция представляет собой сумму каких-то переменных и нуля или единицы. Полагая в формуле (4.4.3)

$$\alpha_1 = \alpha_2 = \ldots = \alpha_n = 0 \,,$$

убеждаемся в том, что *функции, принимающие только одно значение*, т. е. константы 0 и 1, также *являются линейными*. Сравнивая определения 4.4.19 и 4.3.14, легко убедиться в том, что *каждая линейная функция представима полиномом Жегалкина, в котором отсутствуют конъюнкции переменных*. Это позволяет проверять линейность функции, заданной некоторой формулой, путем преобразования этой формулы в равносильный ей полином Жегалкина.

Пример. Проверим, является ли функция $xy\neg z \vee \neg xyz$ линейной:

$$\begin{aligned} xy\neg z \vee \neg xyz &\equiv xy(y \oplus 1) \oplus (x \oplus 1)yz \equiv \\ &\equiv xyz \oplus xy \oplus xyz \oplus yz \equiv xy \oplus yz \,. \end{aligned}$$

Функция нелинейная. □

Имеется и другой способ проверки линейности. Если функция $f(x_1, \ldots, x_n)$ линейная, то должно выполняться равенство (4.4.3), а вместе с ним и равенства

$$\begin{aligned} f(0, 0, \ldots, 0) = \alpha_0, \;\; f(1, 0, \ldots, 0) = \alpha_0 \oplus \alpha_1, \;\; \ldots \\ \ldots, \;\; f(0, 0, \ldots, 0, 1) = \alpha_0 \oplus \alpha_n \,. \end{aligned} \tag{4.4.4}$$

Построив таблицу истинности для функции f, можно легко найти все коэффициенты в (4.4.3), используя соотношения (4.4.4). Теперь мы имеем две функции: проверяемую f и многочлен в правой части равенства (4.4.3). Их значения совпадают на наборах значений переменных, использованных в (4.4.4). Остается проверить, равны ли значения функций при других значениях переменных.

Пример. Проверим, является ли линейной функция

$$g(x, y, z) = (\neg x\neg y \vee xy)\neg z \vee (x \oplus y)z \,.$$

Составляем для нее таблицу истинности 4.12. Строки 1, 5, 3, 2 этой

Таблица 4.12

x	y	z	$g(x,y,z)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

таблицы показывают, что

$$\alpha_0 = g(0,0,0) = 1, \quad \alpha_0 \oplus \alpha_1 = g(1,0,0) = 1\,,$$

$$\alpha_0 \oplus \alpha_2 = g(0,1,0) = 0, \quad \alpha_0 \oplus \alpha_3 = g(0,0,1) = 0\,.$$

Получаем следующие значения коэффициентов многочлена Жегалкина:

$$\alpha_0 = \alpha_1 = \alpha_2 = \alpha_3 = 1\,.$$

Таким образом, если функция $g(x,y,z)$ линейная, то она должна совпасть с функцией

$$1 \oplus x \oplus y \oplus z\,.$$

Нетрудно проверить, что эти две функции действительно совпадают. □

4.4.20. *Класс* $\mathbf{L}$ *всех булевых линейных функций замкнут.*

Доказательство. Утверждение доказывается так же, как и 4.4.7. □

4.4.21. *Говорят, что множество функций* F ***порождает замкнутый класс*** $\mathbf{A}$*, если* $\mathbf{A}$ *содержит те и только те функции, которые можно получить из функций, принадлежащих множеству* F*, с помощью подстановок. Множество функций* $F \subseteq \mathbf{A}$*, порождающее замкнутый класс* $\mathbf{A}$*, называется* ***полным в*** $\mathbf{A}$*.*

Примеры. Для каждой булевой функции можно построить либо ее СДНФ, либо ее СКНФ (4.3.11, 4.3.13), поэтому система функций, содержащая конъюнкцию, дизъюнкцию и отрицание, является полной в $\mathbf{P}_2$. Конъюнкция выражается через дизъюнкцию и отрицание, а дизъюнкция — через конъюнкцию и отрицание, следовательно, полными в $\mathbf{P}_2$ являются системы

$$\{\&,\ \neg\},\quad \{\vee,\ \neg\}.$$

Так как

$$x\&y \equiv (x \mid y) \mid (x \mid y),\quad x \vee y \equiv (x \mid x) \mid (y \mid y),\quad \neg x \equiv x \mid x,$$

система $\{|\}$, содержащая лишь штрих Шеффера, также является полной в $\mathbf{P}_2$.

Системы $\{\&\}$, $\{\vee\}$, $\{\neg\}$ полными в $\mathbf{P}_2$ не являются. Действительно, из конъюнкции можно получить лишь функцию x и элементарные конъюнкции вида $x_1\&x_2\&\ldots\&x_n$. Все эти функции сохраняют единицу. Дизъюнкция порождает лишь функции, сохраняющие нуль. С помощью отрицания можно получить только функции x, $\neg x$ и функции, отличающиеся от них фиктивными переменными. □

4.4.22 (Теорема Поста)**.** *Система булевых функций порождает класс* $\mathbf{P}_2$ *тогда и только тогда, когда она содержит:*

1) *функцию, не сохраняющую нуль;*
2) *функцию, не сохраняющую единицу;*
3) *несамодвойственную функцию;*
4) *немонотонную функцию;*
5) *нелинейную функцию.* □

Доказательство. ($\Rightarrow$) Предположим, что некоторая система функций S не содержит функции, не сохраняющей нуль. Это означает, что каждая функция из S нуль сохраняет и потому содержится в классе $\mathbf{T}_0$. Поскольку класс $\mathbf{T}_0$ замкнутый, при помощи подстановок из функций, входящих в S, можно получить лишь функции, принадлежащие $\mathbf{T}_0$, и нельзя получить функцию, не сохраняющую нуль. Это доказывает, что система S не является полной в $\mathbf{P}_2$.

Необходимость остальных условий, указанных в теореме, доказывается аналогично.

($\Leftarrow$) Пусть система функций Q удовлетворяет условиям 1–5 теоремы 4.4.22. Докажем, что подстановками из функций, принадлежащих Q, можно получить систему функций $\{\neg x, x_1\&x_2\}$, полную в $\mathbf{P}_2$. Доказательство разобьем на части.

1. Имея функцию, не сохраняющую нуль, функцию, не сохраняющую единицу и немонотонную функцию, можно получить отрицание.

Если функция $f(x_1,\dots,x_n)\in Q$ не сохраняет нуль, то $f(0,\dots,0)=$ $=1$. Имеются две возможности.

а) Из $f(1,\dots,1)=0$ следует $f(x,\dots,x)=\neg x$.

б) При $f(1,\dots,1)=1$ получаем $f(x,\dots,x)=c_1^1(x)$.

Система Q содержит какую-то функцию $g(x_1,\dots,x_m)$, не сохраняющую единицу. Из нее конструируем функцию

$$g(c_1^1(x),\dots,c_1^1(x))=c_0^1(x)\,.$$

Теперь используем немонотонную функцию $h(x_1,\dots,x_l)$, имеющиюся в Q. Согласно 4.4.18 существуют два набора

$$(\alpha_1,\dots,\alpha_{i-1},0,\alpha_{i+1},\dots,\alpha_l),\quad (\alpha_1,\dots,\alpha_{i-1},1,\alpha_{i+1},\dots,\alpha_l)$$

из нулей и единиц, обладающие свойством

$$h(\alpha_1,\dots,\alpha_{i-1},0,\alpha_{i+1},\dots,\alpha_l)=1\,,$$

$$h(\alpha_1,\dots,\alpha_{i-1},1,\alpha_{i+1},\dots,\alpha_l)=0\,.$$

Очевидно,

$$h(c_{\alpha_1}(x),\dots,c_{\alpha_{i-1}}(x),x,c_{\alpha_{i+1}}(x),\dots,c_{\alpha_l}(x))=\neg x\,.$$

2. Из отрицания и несамодвойственной функции можно получить константы $c_0^1(x)$ и $c_1^1(x)$.

Если функция $s(x_1,\dots,x_p)\in Q$ несамодвойственная, то найдутся такие числа $\alpha_1,\dots,\alpha_p$, что

$$p(\alpha_1,\dots,\alpha_p)\neq\neg p(\neg\alpha_1,\dots,\neg\alpha_p)\,.$$

Отсюда следует, что

$$p(\alpha_1,\dots,\alpha_p)=p(\neg\alpha_1,\dots,\neg\alpha_p)\,,$$

и потому каждая из функций

$$p(x^{\alpha_1},\dots,x^{\alpha_p})\,,\quad \neg p(x^{\alpha_1},\dots,x^{\alpha_p})\,,$$

где

$$x^{\alpha_i}=\begin{cases}x, & \text{если } \alpha_i=0,\\ \neg x, & \text{если } \alpha_i=1,\end{cases}$$

принимает только одно значение и одна из них совпадает с $c_0^1(x)$, а другая — с $c_1^1(x)$.

3. Имея константы $c_0^1(x)$ и $c_1^1(x)$ и нелинейную функцию, можно получить конъюнкцию $x_1 \& x_2$.

Пусть функция $t(x_1, \ldots, x_r)$ нелинейная и принадлежит системе Q. Она представима многочленом Жегалкина, и этот многочлен содержит нелинейную часть, т. е. хотя бы одну элементарную конъюнкцию вида $x_{i_1} x_{i_2} \cdots x_{i_m}$, где $m > 1$. Переименовав переменные функции t следующим образом:

$$\begin{aligned} &x_{i_1} \to x_1, \quad x_{i_2} \to x_2, \quad \ldots, \quad x_{i_m} \to x_m\,, \\ &x_1 \to x_{i_1}, \quad x_2 \to x_{i_2}, \quad \ldots, \quad x_m \to x_{i_m}\,, \\ &x_j \to x_j \quad \text{для } j \notin \{i_1, i_2, \ldots, i_m\}\,, \end{aligned}$$

получим функцию $t_1(x_1, \ldots, x_r)$, у которой в многочлене Жегалкина есть слагаемое $x_1 x_2 \cdots x_m$. Поскольку многочлен Жегалкина представляет собой сумму каких-то элементарных конъюнкций и константы, которая может быть нулем или единицей, функция

$$u(x_1, x_2) = t_1(\underbrace{x_1, x_2, c_1^1(x_1), c_1^1(x_1), \ldots, c_1^1(x_1)}_{m}, \underbrace{c_0^1(x_1), \ldots, c_0^1(x_1)}_{r-m})$$

содержится в следующем списке:

$$\begin{aligned} &x_1 x_2, \quad x_1 \oplus x_1 x_2, \quad x_2 \oplus x_1 x_2, \quad x_1 \oplus x_2 \oplus x_1 x_2\,, \\ &1 \oplus x_1 x_2, \quad 1 \oplus x_1 \oplus x_1 x_2, \quad 1 \oplus x_2 \oplus x_1 x_2, \quad 1 \oplus x_1 \oplus x_2 \oplus x_1 x_2\,. \end{aligned}$$

Если она попала во вторую строку, то функция $\neg u(x_1, x_2)$ содержится в первой строке, поскольку $\neg x = x \oplus 1$ и $1 \oplus 1 = 0$. Так как

$$\begin{aligned} &x_1 \oplus x_1 \neg x_2 = x_1 \oplus x_1(1 \oplus x_2) = (x_1 \oplus x_1) \oplus x_1 x_2 = x_1 x_2\,, \\ &x_2 \oplus \neg x_1 x_2 = x_2 \oplus (1 \oplus x_1) x_2 = (x_2 \oplus x_2) \oplus x_1 x_2 = x_1 x_2\,, \\ &\neg(\neg x_1 \oplus \neg x_2 \oplus \neg x_1 \neg x_2) = \\ &\qquad = 1 \oplus (1 \oplus x_1) \oplus (1 \oplus x_2) \oplus (1 \oplus x_1)(1 \oplus x_2) = \\ &\qquad = 1 \oplus x_1 \oplus x_2 \oplus 1 \oplus x_1 \oplus x_2 \oplus x_1 x_2 = x_1 x_2\,, \end{aligned}$$

каждая из функций, перечисленных в первой строке, позволяет получить конъюнкцию. □

В табл. 4.13 указано распределение некоторых булевых функций по классам $\mathbf{T}_0$, $\mathbf{T}_1$, $\mathbf{L}$, $\mathbf{M}$, $\mathbf{S}$.

Таблица 4.13

	$\mathbf{T}_0$	$\mathbf{T}_1$	$\mathbf{L}$	$\mathbf{M}$	$\mathbf{S}$
$c_0^0(x)$	+	−	+	+	−
$c_0^1(x)$	−	+	+	+	−
$e_1^1(x)$	+	+	+	+	+
$\neg x$	−	−	+	−	+
$x_1 \& x_2$	+	+	−	+	−
$x_1 \vee x_2$	+	+	−	+	−
$x_1 \supset x_2$	−	+	−	−	−
$x_1 \oplus x_2$	+	−	+	−	−
$x_1 \mid x_2$	−	−	−	−	−
$x_1 \sim x_2$	−	+	+	−	−

4.4.23. *Система* $\{\&,\ \oplus,\ 1\}$ *порождает* $\mathbf{P}_2$.

Доказательство. Система $\{\&,\ \neg\}$ является полной. Так как $\neg x = x \oplus 1$, то система $\{\&,\ \oplus,\ 1\}$ также полная. □

Пример. Выясним, каким из классов $\mathbf{T}_0$, $\mathbf{T}_1$, $\mathbf{L}$, $\mathbf{M}$, $\mathbf{S}$ принадлежит функция

$$f(x,y,z) = \neg x \neg y z \vee \neg x y \neg z \vee x \neg y \neg z \vee xyz\,.$$

1. Так как

$$f(0,0,0) = \neg 0 \neg 00 \vee \neg 00 \neg 0 \vee 0 \neg 0 \neg 0 \vee 000 = 0\,,$$

то $f \in \mathbf{T}_0$.

2. Функция f принадлежит классу T_1, поскольку

$$f(1,1,1) = \neg 1 \neg 11 \vee \neg 11 \neg 1 \vee 1 \neg 1 \neg 1 \vee 111 = 1\,.$$

3. Ищем представляющий функцию f полином Жегалкина:

$$\begin{aligned}
&\neg x \neg y z \vee \neg x y \neg z \vee x \neg y \neg z \vee xyz \equiv \\
&\equiv \neg x \neg y z \oplus \neg x y \neg z \oplus x \neg y \neg z \oplus xyz \equiv \\
&\equiv (x \oplus 1)(y \oplus 1)z \oplus (x \oplus 1)y(z \oplus 1) \oplus x(y \oplus 1)(z \oplus 1) \oplus xyz \equiv \\
&\equiv xyz \oplus xz \oplus yz \oplus z \oplus xyz \oplus xy \oplus \\
&\qquad \oplus yz \oplus y \oplus xyz \oplus xy \oplus xz \oplus x \equiv x \oplus y \oplus z\,.
\end{aligned}$$

Видим, что функция f линейная и потому принадлежит классу $\mathbf{L}$.

4. Функция f немонотонная и не принадлежит классу $\mathbf{M}$. Действительно, пусть

$$x_1 = 0,\ y_1 = 0,\ z_1 = 1,\ x_2 = 0,\ y_2 = 1,\ z_2 = 1,$$

тогда

$$x_1 \le x_2,\ y_1 \le y_2,\ z_1 \le z_2,\quad 1 = f(x_1, y_1, z_1) \ge f(x_2, y_2, z_2) = 0\,.$$

5. Мы уже выяснили, что функция $f(x, y, z)$ может быть задана полиномом Жегалкина $x \oplus y \oplus z$. Двойственной к ней является функция

$$f^*(x, y, z) = \neg f(\neg x, \neg y, \neg z) = \neg(\neg x \oplus \neg y \oplus \neg z) =$$
$$= (x \oplus 1) \oplus (y \oplus 1) \oplus (z \oplus 1) \oplus 1 = x \oplus y \oplus z = f(x, y, z)\,.$$

Функция f является самодвойственной и принадлежит классу $\mathbf{S}$. □

4.4.24. *Множество функций F из замкнутого класса $\mathbf{A}$ называется* ***базисом****, если оно обладает следующими свойствами:*

1) *множество F является полным в $\mathbf{A}$;*

2) *после удаления из F любой функции получается множество, не порождающее $\mathbf{A}$.*

4.4.25. *Любой базис в $\mathbf{P}_2$ содержит не более четырех функций.*

Доказательство. Предположим, что система функций $f_1, \ldots, f_n$ порождает класс $\mathbf{P}_2$ и $n > 5$. Среди этих функций имеются не обязательно различные функции $f_{i_1}, \ldots, f_{i_5}$, обладающие следующими свойствами:

$$f_{i_1} \notin \mathbf{T}_0,\quad f_{i_2} \notin \mathbf{T}_1,\quad f_{i_3} \notin \mathbf{L},\quad f_{i_4} \notin \mathbf{M},\quad f_{i_5} \notin \mathbf{S}.$$

Согласно 4.4.22 множество функций $f_{i_1}, \ldots, f_{i_5}$ тоже порождает класс $\mathbf{P}_2$. Разумеется, некоторые из функций $f_{i_1}, \ldots, f_{i_5}$ могут совпадать. Это означает, что из любого множества функций, содержащего более пяти функций и порождающего $\mathbf{P}_2$, можно выделить подмножество, содержащее не более пяти функций и также порождающее $\mathbf{P}_2$.

Пусть мы имеем систему $\{g_1, \ldots, g_5\}$ из пяти функций, порождающую $\mathbf{P}_2$. Докажем, что среди этих функций хотя бы одна не принадлежит одновременно двум из классов $\mathbf{T}_0$, $\mathbf{T}_1$, $\mathbf{L}$, $\mathbf{M}$, $\mathbf{S}$. Действительно, пусть это неверно. Какая-то из функций $g_1, \ldots, g_5$ не принадлежит классу $\mathbf{T}_0$. Предположим, что этим свойством обладает функция g_1.

Она должна принадлежать остальным классам, в том числе классам $\mathbf{T}_1$ и $\mathbf{S}$. Двойственной к g_1 является функция

$$g_1^*(x_1,\dots,x_n) = \neg g_i(\neg x_1,\dots,\neg x_n)\,.$$

При этих предположениях справедливы следующие соотношения:

$$g_1(0,\dots,0) = 1, \quad g_1(1,\dots,1) = 1,$$
$$g_1^*(0,\dots,0) = \neg g_1(\neg 0,\dots,\neg 0) = \neg g_1(1,\dots,1) = 0\,.$$

Поскольку $g_1^*(0,\dots,0) \neq g_1(0,\dots,0)$, функция g_1 несамодвойственная и не принадлежит классу $\mathbf{S}$, что противоречит предположению.

Таким образом, одна из функций $g_1,\dots,g_5$, например g_{i_0}, не принадлежит одновременно двум из классов $\mathbf{T}_0$, $\mathbf{T}_1$, $\mathbf{L}$, $\mathbf{M}$, $\mathbf{S}$. Остальные три класса не содержат какие-то функции g_{i_1}, g_{i_2}, g_{i_3} из того же списка, и потому множество g_{i_0}, g_{i_1}, g_{i_2}, g_{i_3} порождает $\mathbf{P}_2$. Видим, что для порождения класса $\mathbf{P}_2$ требуется не более четырех функций. □

4.5. КОНТАКТНЫЕ СХЕМЫ

4.5.1. *Контактной схемой* (КС) *называется мультиграф, в котором ребра помечены переменными или отрицаниями переменных и выделено некоторое подмножество вершин, называемых* ***полюсами****. Ребро, помеченное переменной (отрицанием переменной), называется* ***замыкающим (размыкающим) контактом****.*

Пример контактной схемы приведен на рис. 4.3. □

Контактная схема представляет собой математическую модель некоторых электронных устройств. Первоначально такие устройства собирались из **электромеханических реле**. Реле состоит из катушки с сердечником, якоря и контактов. Контакты делятся на два класса: **замыкающие** и **размыкающие**. Если через катушку не проходит ток, то замыкающие контакты разомкнуты, а размыкающие замкнуты. При пропускании через катушку тока якорь притягивается к сердечнику и воздействует на контакты. Замыкающие контакты замкнутся, а размыкающие разомкнутся.

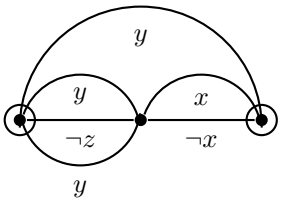


Рис. 4.3

Соединяя контакты нескольких реле, получим электронное устройство, которое также назовем **контактной схемой**.

Пример. Стандартные изображения реле с замыкающими и размыкающими контактами приведены на рис. 4.4. На рис. 4.5 изображена контактная схема, собранная из таких реле. Помеченный мультиграф, соответствующий этой схеме, изображен на рис. 4.6*а*. Особо подчеркнем, что контакты, принадлежащие одному реле, помечаются одной и той же буквой с отрицанием или без отрицания. При подаче тока в катушку реле они срабатывают одновременно. □

Реле с замыкающим контактом

Реле с размыкающим контактом

Рис. 4.4
Виды контактов

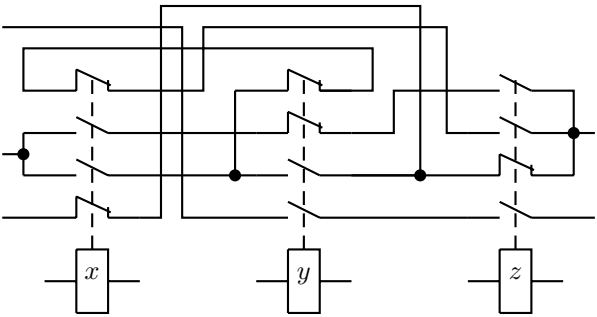


Рис. 4.5

Разумеется, теперь вместо описанных электромеханических реле используются реле электронные, но принцип работы остается прежним: при подаче напряжения на управляющий вывод между некоторыми парами выводов ток проходит, а между другими проходить перестает.

Реле можно заменить двухпозиционными переключателями с несколькими контактами. Одно положение переключателя обозначается через 0, другое — через 1. Контакты, замкнутые в положении 1, замыкающие, замкнутые в положении 0, — размыкающие.

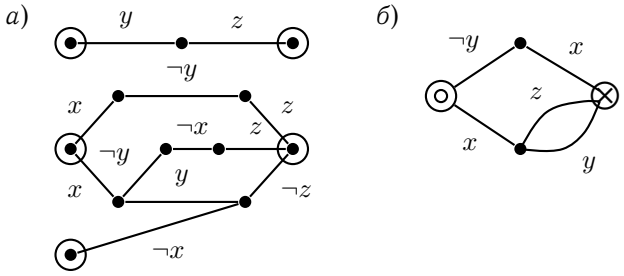


Рис. 4.6

4.5.2. *Контактная схема, множество полюсов которой разбито на два подмножества, называемых* ***множеством входов*** *и* ***множеством выходов*** *и содержащих r и s элементов соответственно, называется* (***контактным***) (r, s)-***полюсником***.

Далее мы будем рассматривать только $(1, 1)$-полюсники, т. е. контактные схемы с одним входом и одним выходом. Такие схемы называют **двухполюсными**. Вход и выход контактной схемы будем обозначать значками $\circledcirc$ и $\otimes$, а вершины КС, не являющиеся полюсами, — значком $\bullet$.

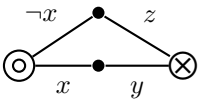
Пример. Изображенная на рис. 4.6*б* контактная схема двухполюсная. □

Выделим два важных способа получения новой КС из двух имеющихся.

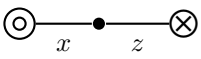
4.5.3. *Говорят, что* КС C *получена из* КС A *и* B ***последовательным соединением***, *если граф C получен из дизъюнктного объединения графов A и B отождествлением выхода схемы A со входом схемы B. Входом схемы C считается вход схемы A, а выходом — выход схемы B.*

4.5.4. *Будем говорить, что* КС C *получена из* КС A *и* B ***параллельным соединением***, *если граф схемы C получен из дизъюнктного объединения графов A и B отождествлением входа схемы A со входом схемы B и выхода схемы A с выходом схемы B. Первая из полученных вершин считается входом схемы C, а вторая — выходом.*

Пример. Соединяя изображенные на рис. 4.7 КС A и B последовательно и параллельно, получим соответственно КС на рис. 4.8 и 4.9. □



Контактная схема A



Контактная схема B

Рис. 4.7
Исходные контактные схемы

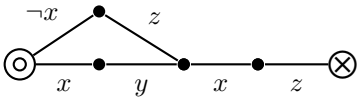


Рис. 4.8
Последовательное соединение КС A и B

Предположим, что из n реле собрана двухполюсная контактная схема. Обозначим эти реле буквами $x_1, \ldots, x_n$. Пропустим ток через обмотки некоторых реле. Сопоставим этому событию набор $\alpha_1, \ldots, \alpha_n$ из нулей и единиц, полагая

$$\alpha_i = \begin{cases} 0, & \text{если через обмотку реле } x_i \text{ не пропущен ток,} \\ 1, & \text{если через обмотку реле } x_i \text{ пропущен ток.} \end{cases}$$

Замыкающие контакты реле, через обмотки которых пропущен ток, замкнутся, размыкающие — разомкнутся. У остальных реле замыкающие контакты будут разомкнуты, а размыкающие — замкнуты. Сопоставим набору $\alpha_1, \ldots, \alpha_n$ число 1, если от входа к выходу КС может пройти ток, число 0 в противном случае. Перебирая все возможные комбинации включенных и выключенных реле, мы каждому набору из нулей и единиц однозначно сопоставим либо нуль, либо единицу. Тем самым рассматриваемой КС будет однозначно сопоставлена некоторая функция алгебры логики $f(x_1, \ldots, x_n)$, называемая **функцией проводимости** этой схемы.

Определение функции проводимости КС дано для случая, когда КС представляет собой некоторое физическое устройство, собранное из электромеханических реле. Приведем теперь определение для математической модели такой КС, т. е. для помеченного мультиграфа.

4.5.5. *Пусть M — простая цепь, связывающая два полюса контактной схемы. Конъюнкция переменных и отрицаний переменных, которыми*

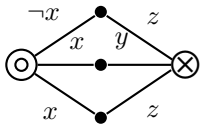


Рис. 4.9
Параллельное соединение КС A и B

помечены входящие в M ребра, называется ***проводимостью*** *данной* ***цепи***. ***Проводимостью между двумя полюсами*** *называется дизъюнкция проводимостей всех цепей, связывающих эти полюсы. Соответствующая этой проводимости функция алгебры логики называется* ***функцией проводимости между*** *указанными* ***полюсами***.

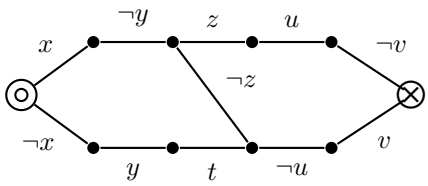


Рис. 4.10
Двухполюсник K

Пример. В контактной схеме на рис. 4.10 вход и выход соединяют четыре простые цепи, поэтому функция проводимости схемы описывается следующей формулой:

$$x\neg yzu\neg v \vee x\neg y\neg z\neg uv \vee \neg xyt\neg uv \vee \neg xyt\neg zzu\neg v\,. \qquad \square$$

Возникают следующие три задачи.

1. КС задана в виде помеченного мультиграфа. Описать ее функцию проводимости формулой.

2. Задана булева функция f. Построить КС, для которой f является функцией проводимости.

3. КС задана в виде помеченного мультиграфа. Построить новую КС с меньшим числом контактов, но с той же функцией проводимости.

Эти проблемы мы рассмотрим лишь для особых двухполюсных контактных схем, описанных ниже.

4.5.6. *Каждый двухполюсник, содержащий единственный контакт, называется* П-***схемой.*** КС, *полученная последовательным или параллельным соединением двух* П-*схем, также называется* П-***схемой.*** *Других* П-*схем нет.*

П-схемы известны также под названием **параллельно-последовательных схем**.

Поскольку каждая отличная от двухполюсника П-схема получена последовательным или параллельным соединением двух меньших П-схем, то ее можно разделить на две части. Затем для каждой из этих меньших П-схем также можно определить, соединением каких П-схем они получены, и т. д.

Пример. На рис. 4.11*а* изображена П-схема. Контактная схема на рис. 4.11*б* не является П-схемой. □

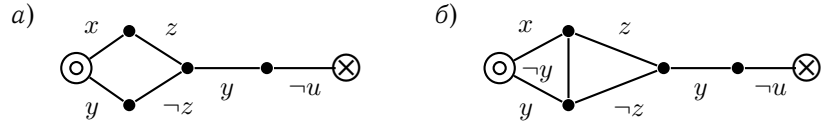


Рис. 4.11

4.5.7. ***Функцией проводимости*** П-***схемы*** *называется функция проводимости между входом и выходом этой схемы.*

4.5.8. *Для любой функции алгебры логики f можно построить* П-*схему, функция проводимости которой совпадает с f.*

Доказательство. Согласно 4.3.11 если функция f хотя бы один раз принимает значение 1, то ее можно представить СДНФ. Предположим, что этой функции соответствует СДНФ F и $x_1^{\sigma_1}x_2^{\sigma_2}\dots x_n^{\sigma_n}$ — одна из входящих в F элементарных конъюнкций. При последовательном соединении контактов с пометками $x_1^{\sigma_1}, x_2^{\sigma_2}, \dots, x_n^{\sigma_n}$ получится П-схема (рис. 4.12), проводимость которой совпадает с $x_1^{\sigma_1}x_2^{\sigma_2}\dots x_n^{\sigma_n}$.

Пусть $S_1, \dots, S_k$ — П-схемы, построенные указанным способом и отвечающие всем элементарным конъюнкциям, входящим в F. Соединяя их параллельно, получим П-схему, изображенную на рис. 4.13, функция проводимости которой совпадает с f.

Если же функция $f(x_1, \dots, x_n)$ тождественно равна нулю, то такой функцией проводимости обладает, например, П-схема, изображенная на рис. 4.14. □

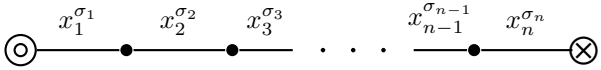


Рис. 4.12

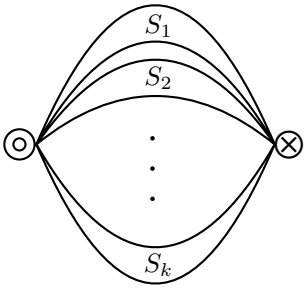


Рис. 4.13
Параллельное соединение схем S_1 – S_k

Пример. Построим П-схему, функция проводимости которой равна единице только в случаях, указанных в табл. 4.14. Запишем СДНФ этой функции:

$$\neg x\&y\&\neg z\&u \vee x\&y\&\neg z\&\neg u \vee x\&y\&\neg z\&u \vee x\&y\&z\&u.$$

Таблица 4.14

x	y	z	u
0	1	0	1
1	1	0	0
1	1	0	1
1	1	1	1

Этой функции отвечает П-схема на рис. 4.15. □

4.5.9. *Функцию проводимости каждой* П*-схемы можно задать формулой со связками* &, ∨, ¬, *в которой сумма числа вхождений всех переменных и отрицаний переменных совпадает с числом контактов этой схемы.*

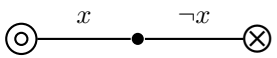


Рис. 4.14
Контактная схема с нулевой проводимостью

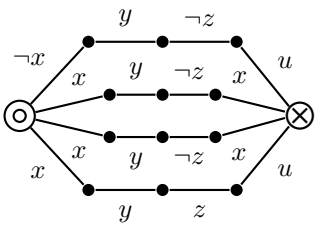


Рис. 4.15

Доказательство. Пусть дана П-схема P. Если она имеет более одного контакта, то согласно 4.5.6 P получена соединением двух П-схем P_1^0 и P_2^0. Запишем это утверждение в виде формулы

$$P = (P_1^0 s_0 P_2^0)\,,$$

в которой s_0 — знак конъюнкции (дизъюнкции), если соединение схем P_1^0 и P_2^0 последовательное (параллельное). Аналогично получаем формулы

$$P_1 = (P_1^1 s_1 P_2^1), \quad P_2 = (P_1^2 s_2 P_2^2)\,,$$
$$P = ((P_1^1 s_1 P_2^1) s_1 (P_1^2 s_2 P_2^2))$$

и т. д. Процесс построения формул оборвется тогда, когда каждая буква P_i^j будет обозначать схему, имеющую один контакт. Заменяя все эти буквы переменными с отрицаниями или без отрицаний, которыми помечены соответствующие контакты, получим формулу, задающую функцию проводимости схемы P и удовлетворяющую условиям теоремы. □

Пример. Проводимость П-схемы на рис. 4.16 описывается формулой

$$(x \vee y \vee \neg v \& \neg x) \& (u \vee z \vee \neg y) \& (u \vee v)\,.$$ □

Проблема построения КС с минимальным числом контактов, функция проводимости которой совпадает с функцией проводимости заданной КС, является весьма сложной. Такая задача всегда может быть решена перебором всех вариантов, но для достаточно больших схем

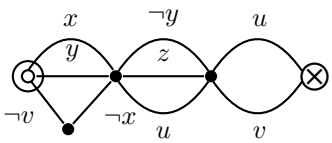


Рис. 4.16

число вариантов столь велико, что метод становится непригодным. Разработан ряд менее трудоемких алгоритмов, позволяющих упрощать контактные схемы, однако они не гарантируют построение схемы с минимальным числом контактов. Если же КС невелика, то упростить ее можно, не используя такие алгоритмы.

Пример. Упростим КС на рис. 4.17*а*. Для этого преобразуем формулу ее проводимости:

$$xyz \vee \neg xyz \vee x\neg yz \equiv (xy \vee \neg xy \vee x\neg y)z \equiv$$

$$\equiv ((x \vee \neg x)y \vee x\neg y)z \equiv (y \vee x\neg y)z \equiv (y \vee x)(y \vee \neg y)z \equiv (x \vee y)z\,.$$

Получаем КС, изображенную на рис. 4.17*б*. □

4.6. ИСЧИСЛЕНИЕ ВЫСКАЗЫВАНИЙ

Теперь мы познакомимся с **дедуктивными системами**. Латинское слово *deduktio* означает *выведение*, и под дедукцией понимают переход от общего к частному. Дедуктивные системы, называемые также **формальными системами, исчислениями**, строятся следующим образом. Указываются некоторые начальные элементы, называемые **аксиомами**, и **правила вывода**, т. е. способы получения новых элементов системы из аксиом и уже построенных элементов. Общими считаются аксиомы, частными — те элементы, которые можно построить указанным способом за конечное число шагов.

Добавим, что греческое слово *aksiōma* переводится как *достойное признания*. Под аксиомой обычно понимают исходное положение теории, в которой оно принимается без доказательства.

Исчисление, которое мы будем рассматривать, строится по следующей схеме.

1. Выбирается некоторый алфавит.

2. Указываются правила построения из букв этого алфавита слов, называемых **правильно построенными формулами**, или просто **форму-**

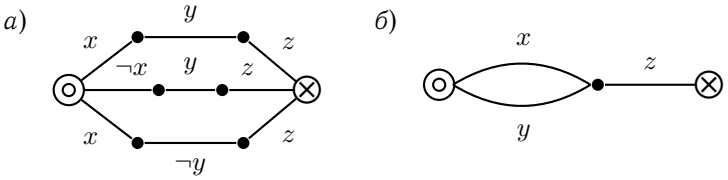


Рис. 4.17

лами. Эти правила образуют **синтаксис** исчисления. (Греческое слово *syntaxis* означает *составление, устройство*.)

3. В множестве формул выделяется некоторое подмножество (обычно конечное или счетное), элементы которого называются **аксиомами**.

4. Задаются **правила вывода** новых формул из аксиом и уже выведенных формул.

После этого множество всех правильно построенных формул распадается на два подмножества. К одному принадлежат формулы, которые при помощи правил вывода можно получить (вывести) из аксиом за конечное число шагов, а к другому — все остальные формулы. Тем самым мы моделируем процесс построения математической теории: при таком построении из некоторых утверждений, принимаемых без доказательства, по правилам формальной логики выводятся теоремы.

Следует отметить, что одно и то же множество выводимых формул можно получить при разных списках аксиом и правил вывода. Ниже приводится лишь один из вариантов.

Алфавит, с которым мы будем работать, отличается от введенного в 4.1.6 меньшим количеством связок и тем, что множество логических символов фиксировано.

4.6.1. *Алфавит исчисления высказываний образован тремя группами букв.*

1. *Переменные — строчные латинские буквы с нижними индексами или без индексов.*

2. *Логические символы* $\&$ (*и*), $\vee$ (*или*), $\supset$ (*влечет*), $\neg$ (*не*).

3. *Синтаксические символы* (,).

Определение формулы отличается от определения 4.1.6 только списком используемых связок.

4.6.2. а) *Каждая переменная является* ***формулой****.*

б) *Если* $\mathcal{A}$ *и* $\mathcal{B}$ *— формулы, то*

$$(\mathcal{A}\&\mathcal{B}),\quad (\mathcal{A}\vee\mathcal{B}),\quad (\mathcal{A}\supset\mathcal{B}),\quad \neg\mathcal{A}$$

также являются ***формулами****.*

в) *Не существует никаких формул, кроме построенных согласно правилам* а) *и* б)*.*

С целью упрощения записей мы будем далее пользоваться прежними правилами устранения лишних скобок в формулах. В частности, не будем писать внешние скобки.

4.6.3. *Следующие выражения называются* ***схемами аксиом****.*

$s1.$ $\mathcal{A}\supset(\mathcal{B}\supset\mathcal{A})$.
$s2.$ $(\mathcal{A}\supset\mathcal{B})\supset((\mathcal{A}\supset(\mathcal{B}\supset\mathcal{C}))\supset(\mathcal{A}\supset\mathcal{C}))$.
$s3.$ $(\mathcal{A}\&\mathcal{B})\supset\mathcal{A}$.
$s4.$ $(\mathcal{A}\&\mathcal{B})\supset\mathcal{B}$.
$s5.$ $(\mathcal{A}\supset\mathcal{B})\supset((\mathcal{A}\supset\mathcal{C})\supset(\mathcal{A}\supset(\mathcal{B}\&\mathcal{C})))$.
$s6.$ $\mathcal{A}\supset(\mathcal{A}\vee\mathcal{B})$.
$s7.$ $\mathcal{B}\supset(\mathcal{A}\vee\mathcal{B})$.
$s8.$ $(\mathcal{A}\supset\mathcal{C})\supset((\mathcal{B}\supset\mathcal{C})\supset((\mathcal{A}\vee\mathcal{B})\supset\mathcal{C}))$.
$s9.$ $(\mathcal{A}\supset\neg\mathcal{B})\supset(\mathcal{B}\supset\neg\mathcal{A})$.
$s10.$ $\neg\neg\mathcal{A}\supset\mathcal{A}$.

Аксиомой *называется слово, возникающее в результате подстановки в схему аксиомы вместо букв* $\mathcal{A},\mathcal{B},\mathcal{C}$ *конкретных формул.*

4.6.4. *Единственным* ***правилом вывода*** *исчисления высказываний является правило*

$$\frac{\mathcal{A},\ \mathcal{A}\supset\mathcal{B}}{\mathcal{B}}\,, \tag{4.6.1}$$

в котором $\mathcal{A}$ *и* $\mathcal{B}$ *— формулы. Говорят, что формула* $\mathcal{B}$ ***получена из*** $\mathcal{A}$ *и* $\mathcal{A}\supset\mathcal{B}$ ***по правилу*** (4.6.1)*, а также что формула* $\mathcal{B}$ *является* ***непосредственным следствием*** *формул* $\mathcal{A}$ *и* $\mathcal{A}\supset\mathcal{B}$*.*

Указанное в 4.6.4 правило называется modus ponens, т. е. положительный способ (заключения) (лат.). Вариант названия — утверждающий модус. При ссылках на него будем использовать сокращение MP. Известно также правило modus tollens — отрицательный способ (или отрицающий модус). Оно имеет вид

$$\frac{\mathcal{A}\supset\mathcal{B},\ \neg\mathcal{B}}{\neg\mathcal{A}}\,.$$

4.6.5. *Конечная последовательность формул исчисления высказываний, каждый член которой является либо аксиомой, либо непосредственным следствием предыдущих формул, называется* ***выводом****, а ее члены называются* ***выводимыми формулами****. Запись* $\vdash \mathcal{A}$ *означает, что формула* $\mathcal{A}$ *выводима в исчислении высказываний.*

Пример. Выведем формулу $x \supset x$:

$$
\begin{aligned}
&x \supset (x \supset x) \quad (s1)\,,\\
&(x \supset (x \supset x)) \supset ((x \supset ((x \supset x) \supset x)) \supset (x \supset x)) \quad (s2)\,,\\
&(x \supset ((x \supset x) \supset x)) \supset (x \supset x) \quad (\mathrm{MP})\,,\\
&x \supset ((x \supset x) \supset x) \quad (s1)\,,\\
&x \supset x \quad (\mathrm{MP})\,.
\end{aligned}
$$

Первая из формул в этой последовательности получена по схеме аксиомы $s1$ заменой $\mathcal{A}$ и $\mathcal{B}$ на x, вторая — из $s2$ заменой $\mathcal{B}$ на $x \supset x$, $\mathcal{A}$ и $\mathcal{C}$ на x.

Третья формула получена из двух предыдущих по правилу вывода. Заменяя в $s1$ $\mathcal{A}$ на x и $\mathcal{B}$ на $x \supset x$, получаем четвертую формулу. Пятая получена из четвертой и третьей применением правила вывода. □

4.6.6. ***Выводом из множества формул*** Γ *называется такая последовательность формул, каждый член которой является либо аксиомой, либо одной из формул, принадлежащих* Γ*, либо непосредственным следствием предыдущих формул. Члены этой последовательности называются* ***формулами, выводимыми из*** Γ*. Чтобы показать, что формула* $\mathcal{A}$ *выводима из* Γ*, пишут* $\Gamma \vdash \mathcal{A}$*.*

Нетрудно заметить, что если $\Gamma \vdash \mathcal{A}$, то формула $\mathcal{A}$ выводима в исчислении, полученном из исчисления высказываний добавлением к множеству аксиом формул, принадлежащих Γ.

Пример. Докажем, что $\mathcal{P}, \mathcal{Q} \vdash \mathcal{P}\&\mathcal{Q}$:

$$
\begin{aligned}
&\mathcal{P} \supset (\mathcal{Q} \supset \mathcal{P}) \quad (s1)\,,\\
&\mathcal{P}\,,\\
&\mathcal{Q} \supset \mathcal{P} \quad (\mathrm{MP})\,,\\
&(\mathcal{Q} \supset \mathcal{P}) \supset ((\mathcal{Q} \supset \mathcal{Q}) \supset (\mathcal{Q} \supset (\mathcal{P}\&\mathcal{Q}))) \quad (s5)\,,\\
&(\mathcal{Q} \supset \mathcal{Q}) \supset (\mathcal{Q} \supset (\mathcal{P}\&\mathcal{Q})) \quad (\mathrm{MP})\,,\\
&\mathcal{Q} \supset (\mathcal{Q} \supset \mathcal{Q}) \quad (s1)\,,\\
&\mathcal{Q}
\end{aligned}
$$

$$\mathcal{Q} \supset \mathcal{Q} \quad (\mathrm{MP}),$$
$$\mathcal{Q} \supset (\mathcal{P}\&\mathcal{Q}) \quad (\mathrm{MP}),$$
$$\mathcal{P}\&\mathcal{Q} \quad (\mathrm{MP}). \qquad \square$$

4.6.7. *Если* $\Gamma_1 \vdash \mathcal{A}$ *и* $\Gamma_2 \supseteq \Gamma_1$, *то* $\Gamma_2 \vdash \mathcal{A}$. *В частности, если* $\vdash \mathcal{A}$, *то* $\Gamma \vdash \mathcal{A}$ *для любого множества формул* Γ.

Доказательство. Согласно 4.6.6 если $\Gamma_1 \vdash \mathcal{A}$, то существует такая последовательность формул, каждый член которой является либо аксиомой, либо одной из формул, принадлежащих Γ_1, либо непосредственным следствием предыдущих формул. По условию все принадлежащие Γ_1 формулы этой последовательности принадлежат и множеству Γ_2, поэтому данная последовательность является также выводом из Γ_2.

Если положить $\Gamma_1 = \varnothing$ и $\Gamma_2 = \Gamma$, то второе утверждение получается из первого. $\square$

4.7. СЕМАНТИКА ИСЧИСЛЕНИЯ ВЫСКАЗЫВАНИЙ

Попав в страну, где жители говорят на незнакомом языке, и гуляя по улицам какого-либо города, испытываешь большие неудобства. Вот дверь, над ней вывеска, но что на ней написано? Булочная, парикмахерская, ресторан, клуб? Если язык коренным образом отличается от родного, то все варианты возможны. Таким образом, смысл написанного может быть самый разный.

Но и в родном языке немало примеров, показывающих, что одно и то же написанное слово может иметь разный смысл. В предложениях:

Господин N купил замок на южном побережье Франции.
На двери избы висел замок, следовательно, хозяев не было дома,

слово «замок» означает совершенно разные предметы.

В математике одна и та же буква x в одной задаче может означать скорость, в другой — количество студентов, в третьей — температуру и т. д. Даже символом сложения $+$ в разных случаях могут быть обозначены совершенно разные операции.

Приведенные примеры показывают, что буквам и составленным из них словам можно придавать различные значения для достижения определенных целей. Раздел логики, в котором исследуется смысл логических знаков и выражений, называется **семантикой** (от греческого *semanticos* — обозначающий).

4.7.1. *Придание математическим выражениям определенного смысла называется* ***интерпретацией*** *этих выражений.*

Латинское слово *interpretatio* означает *объяснение смысла* чего-либо. В математической логике семантикой называют изучение интерпретаций логического исчисления.

Интерпретируя формулы исчисления высказываний, обычно задают некоторое множество A и считают, что все встречающиеся в формулах переменные изменяются на этом множестве. Логическим символам сопоставляются операции, определенные на том же множестве и имеющие соответствующую арность. Используя метод математической индукции и определение 4.6.2, нетрудно обнаружить, что после этого каждой формуле также оказывается сопоставлена конкретная операция, определенная на A. Сама же формула просто указывает последовательность действий, которые надо совершить, чтобы найти значение этой операции при заданных значениях переменных.

Среди всех возможных интерпретаций наиболее употребительна одна, называемая **стандартной** (или **главной**) интерпретацией, которую мы сейчас и рассмотрим. Будем предполагать, что переменные формул могут принимать два значения — 0 и 1. Логическим связкам $\&$, $\vee$, $\supset$, $\neg$ сопоставим операции на множестве $E_2 = \{0, 1\}$, которые были уже введены в разделе 4.1 (табл. 4.1). Многие свойства формул, возникающие при такой интерпретации, уже были рассмотрены в разделе 4.1–4.3. В частности, там было приведено определение 4.2.6 тождественно истинной и тождественно ложной формул. Теперь посмотрим, какими свойствами обладают выводимые формулы и как этими свойствами можно воспользоваться.

4.7.2. *Каждая формула, выводимая в исчислении высказываний, является тождественно истинной.*

Доказательство. Каждая аксиома исчисления высказываний является тождественно истинной формулой. Докажем это, например, для аксиом, полученных по схеме $s6$. Составим следующую таблицу, показывающую, что при любых значениях, которые могут принимать формулы $\mathcal{A}$ и $\mathcal{B}$, аксиома, полученная по этой схеме, принимает значение 1. Аналогично доказывается тождественная истинность аксиом, полученных по остальным схемам.

Теперь докажем, что если формулы $\mathcal{A}$ и $\mathcal{A} \supset \mathcal{B}$ тождественно истинны, то тождественно истинной является и формула $\mathcal{B}$, полученная из них по правилу 4.6.4. Действительно, предположим, что формулы $\mathcal{A}$ и $\mathcal{A} \supset \mathcal{B}$ тождественно истинны, а формула $\mathcal{B}$ при некоторых значениях переменных принимает значение 0. Из табл. 4.1 видно, что если при тех же значениях переменных значение формулы $\mathcal{A} \supset \mathcal{B}$ равно 1,

Таблица 4.15

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A} \vee \mathcal{B}$	$\mathcal{A} \supset (\mathcal{A} \vee \mathcal{B})$
0	0	0	1
1	0	1	1
0	1	1	1
1	1	1	1

то формула $\mathcal{A}$ должна иметь значение 0, что противоречит нашему предположению.

Видим, что все аксиомы исчисления высказываний являются тождественно истинными и что каждая формула, выводимая из какой-либо совокупности тождественно истинных формул, также является тождественно истинной. □

4.7.3. *Каждая тождественно истинная формула выводима в исчислении высказываний.*

Доказательство. Это утверждение мы примем без доказательства ввиду значительного объема последнего. □

4.7.4. *Исчисление называется* ***противоречивым****, если в нем найдется такая выводимая формула $\mathcal{A}$, что формула $\neg\mathcal{A}$ также выводима. В ином случае исчисление называется* ***непротиворечивым****.*

4.7.5. *Исчисление высказываний непротиворечиво.*

Доказательство. Предположим, что для некоторой выводимой формулы $\mathcal{A}$ формула $\neg\mathcal{A}$ также является выводимой. Согласно 4.7.2 обе формулы являются тождественно истинными. Это невозможно, так как значения формул $\mathcal{A}$ и $\neg\mathcal{A}$ противоположны при любых значениях входящих в них переменных. □

4.7.6. *Множество всех формул, выводимых в некотором исчислении, называется* ***разрешимым****, если существует алгоритм, позволяющий для каждой правильно построенной формулы определить, принадлежит ли она этому множеству.*

4.7.7. *Множество выводимых формул исчисления высказываний разрешимо.*

Доказательство. Чтобы узнать, выводима ли некоторая формула в исчислении высказываний, достаточно выяснить, является ли она тождественно истинной. Для этого надо для каждого набора значений переменных вычислить значение этой формулы. Поскольку переменные могут принимать лишь два значения, это всегда можно сделать за конечное число шагов. □

Вопрос о том, существует ли алгоритм, позволяющий для каждой формулы узнать, выводима ли она в рассматриваемом исчислении, называют **проблемой разрешимости**. Ввиду этого утверждение 4.7.7 можно сформулировать следующим образом: *проблема разрешимости исчисления высказываний имеет положительное решение.*

4.7.8. *Множество схем аксиом исчисления называется* ***независимым****, если для каждой схемы существует полученная по этой схеме аксиома, не выводимая в другом исчислении, отличающемся от рассматриваемого лишь отсутствием этой схемы.*

Независимость схем аксиом исчисления высказываний устанавливается с помощью нестандартных интерпретаций.

4.7.9. *Множество схем аксиом исчисления высказываний является независимым.*

Доказательство. Докажем, например, независимость схемы $s3$ от остальных схем. Рассмотрим следующую интерпретацию исчисления. Будем по-прежнему считать, что переменные всех формул изменяются на множестве E_2. Формулам

$$x \supset y, \quad x \vee y, \quad \neg x$$

сопоставим те же функции, что и в табл. 4.1 из раздела 4.1, а формуле $x\&y$ — функцию, отличную от указанной в стандартной интерпретации: будем считать, что значение формулы $x\&y$ всегда совпадает со значением переменной y.

Интерпретация связок $\vee$, $\supset$, $\neg$ совпадает со стандартной, поэтому аксиомы, полученные по схемам, отличным от схем $s3$, $s4$ и $s5$, являются тождественно истинными (см. доказательство утверждения 4.7.2) В схемах $s4$ и $s5$ содержится связка $\&$, поэтому надо убедиться в том, что аксиомы, полученные по этим схемам, также являются тождественно истинными.

Рассмотрим схему $s4$:

$$\mathcal{A}\&\mathcal{B} \supset \mathcal{B}\,. \tag{4.7.1}$$

Формула, полученная по этой схеме, может быть ложной только в том случае, когда $\mathcal{B} = 0$ и $\mathcal{A}\&\mathcal{B} = 1$. Однако в нашей интерпретации значения формул $\mathcal{A}\&\mathcal{B}$ и $\mathcal{B}$ совпадают, поэтому если $\mathcal{A}\&\mathcal{B} = 1$, то $\mathcal{B} = 1$. Полученное противоречие доказывает, что формула (4.7.1) не может быть ложной ни при каких значениях переменных.

Формула

$$(\mathcal{A} \supset \mathcal{B}) \supset ((\mathcal{A} \supset \mathcal{C}) \supset (\mathcal{A} \supset (\mathcal{B}\&\mathcal{C}))),$$

полученная по схеме $s5$, принимает значение 0 только тогда, когда

$$\mathcal{A} \supset \mathcal{B} = 1, \quad (\mathcal{A} \supset \mathcal{C}) \supset (\mathcal{A} \supset (\mathcal{B}\&\mathcal{C})) = 0.$$

Второе равенство справедливо только при

$$\mathcal{A} \supset \mathcal{C} = 1, \quad \mathcal{A} \supset (\mathcal{B}\&\mathcal{C}) = 0.$$

Из последнего равенства заключаем, что

$$\mathcal{A} = 1, \quad \mathcal{B}\&\mathcal{C} = 0.$$

Так как

$$\mathcal{B}\&\mathcal{C} = \mathcal{C},$$

то $\mathcal{C} = 0$. Это противоречит равенству $\mathcal{A} \supset \mathcal{C} = 1$, поскольку мы установили, что $\mathcal{A} = 1$, а при $\mathcal{A} = 1$ оно справедливо лишь при $\mathcal{C} = 1$.

Мы убедились в том, что все аксиомы, полученные по схемам, отличным от $s3$, являются тождественно истинными. Рассуждение, проведенное при доказательстве утверждения 4.7.2, показывает, что все формулы, получаемые по правилу modus ponens из тождественно истинных формул, также являются тождественно истинными. Это означает, что все формулы, выводимые без использования аксиом, полученных по схеме $s3$, также являются тождественно истинными.

Формула $x\&y \supset x$ получена по схеме $s3$ и является аксиомой. При $x = 0$, $y = 1$ ее значение равно 0, значит, ее нельзя вывести из аксиом, полученных по схемам, отличным от $s3$. Это доказывает независимость схемы $s3$.

Независимость остальных схем аксиом доказывается аналогично. Укажем лишь подходящие интерпретации. Для схем $s4$–$s9$ все функции, за исключением одной, определены так же, как и при стандартной интерпретации. Особый вид имеют функции, указанные ниже для каждой из этих схем:

$s4$. $x\&y = x$.

$s5$. $x\&y = 0$.

$s6$. $x \vee y = y$.
$s7$. $x \vee y = x$.
$s8$. $x \vee y = 1$.
$s9$. $\neg x = 0$.

Для схем $s1$, $s2$ и $s10$ интерпретации выглядят сложнее. Связкам сопоставляются функции, определенные на множестве $E_3 = \{0, 1, 2\}$. Одно из чисел, принадлежащих E_3, объявляется выделенным значением. Все формулы, выводимые без использования испытуемой схемы аксиом, принимают только это значение, а аксиома, полученная по этой схеме, принимает также и другое значение.

$s10$. 2 — выделенное значение,

$$x\&y = \min(x, y)\,, \quad x \vee y = \max(x, y)\,,$$

$$x \supset y = \begin{cases} 2, \text{ если } x \leqslant y, \\ y, \text{ если } x > y, \end{cases} \qquad \neg x = \begin{cases} 0, \text{ если } x > 0, \\ 2, \text{ если } x = 0. \end{cases}$$

$s1$. 2 — выделенное значение,

$$x\&y = \min(x, y), \quad x \vee y = \max(x, y), \quad \neg x = 2 - x,$$

$$x \supset y = \begin{cases} 2, \text{ если } x \leqslant y, \\ y, \text{ если } x > y. \end{cases}$$

$s2$. 0 — выделенное значение,

$$x\&y = \max(x, y)\,, \quad x \vee y = \min(x, y)\,,$$

$$x \supset y = \max(0, y - x)\,, \quad \neg x = 2 - x\,. \qquad \square$$

4.8. ЯЗЫК ЛОГИКИ ПРЕДИКАТОВ

До сих пор мы обозначали переменными целые высказывания и нас интересовало лишь то, истинными они являются или ложными. Однако каждое высказывание представляет собой некоторое суждение о предмете суждения (**субъекте**). Предметы, о которых делается суждение, могут быть самой различной природы, например натуральные числа, люди, живущие на земном шаре, и т. д. При построении математической теории фиксируют совокупность изучаемых предметов. Такая совокупность называется **предметной областью**, или **универсом**. Последнее название происходит от латинского слова *universalis*, означающего *общий, всеобщий*.

То, что в предложении говорится о предмете (или нескольких предметах), в логике называется **предикатом**. Латинское слово *praedicatum* означает *сказуемое*.

4.8.1. *Предикатом на множестве A называется функция, определенная на A и сопоставляющая каждому набору значений переменных некоторое высказывание об этом наборе. Одноместные предикаты называются* ***свойством****, многоместные —* ***отношением****.*

Иногда вводятся также нульместные предикаты. Их обыкновенно называют **константами**. Константы представляют собой имена конкретных предметов, принадлежащих универсу.

Примеры. Пусть универсом является множество $\mathbb{N}$ натуральных чисел.

1. Числа 5 и 22 являются константами, т. е. нульместными предикатами.

2. Высказывание «x является простым числом» служит примером одноместного предиката. Этот предикат является свойством, которым одни натуральные числа обладают, а другие не обладают. Подставляя вместо x натуральное число, например 8, получаем высказывание «8 является простым числом». Иначе говоря, это высказывание сопоставлено значению переменной x, равному 8.

3. Двуместным отношением является, например, предложение «x делится на y». □

Возьмем в качестве предметной области множество $\mathbb{R}$ и рассмотрим высказывание «сумма $x + y$ больше числа x». Придавая переменным x и y какие-либо значения, мы получим высказывание о конкретных числах, например «сумма $5 + 3$ больше числа 5». Таким образом, мы имеем дело с предикатом, имеющим две переменные. Обозначим его через $P(x, y)$. Существует два важных способа получения из предиката P предикатов с меньшим числом переменных.

а) Заменим приведенное выше высказывание высказыванием «для каждого x сумма $x + y$ больше числа x». Это утверждение можно записать в следующем виде:

$$\forall x P(x, y).$$

Символ $\forall$ заменяет слова «для каждого». Полученное утверждение представляет собой одноместный предикат, в котором придавать значения можно только переменной y.

б) Из предиката P можно получить другой одноместный предикат, заменив первоначальное высказывание высказыванием «существует такое x, что сумма $x + y$ больше числа x». Его можно записать так:

$$\exists x P(x, y).$$

Здесь символ $\exists$ заменяет слово «существует». Добавление к первоначальному предложению слов «такое» и «что» диктуется правилами грамматики русского языка, но и без них смысл предложения $\exists x P(x, y)$ понятен.

4.8.2. *Символ $\forall$ называется* ***квантором всеобщности****, читается «для каждого». Символ $\exists$ называется* ***квантором существования****, читается «существует».*

Термин «квантор» образован от латинского слова *quantum* — сколько.

При формальной записи за квантором всегда следует некоторая переменная, которая называется **связанной** (этим квантором). Про сам квантор говорят, что он (действует) **по этой переменной**.

Пример. В слове $\forall z Q(x, y, z)$ мы видим квантор $\forall$ по переменной z. Переменная z является связанной. □

Связывание некоторой переменной квантором превращает n-местный предикат в $(n-1)$-местный и потому является операцией над предикатами.

Пример. Пусть, как и ранее, $P(x, y)$ означает высказывание $x + y > x$ и предметной областью является множество $\mathbb{R}$. Предикат $\forall x P(x, y)$ при $y = 0$ превращается в ложное высказывание, а при $y = 1$ — в истинное. Отсюда следует, что высказывание $\forall x \exists y P(x, y)$ является истинным, а высказывание $\forall x \forall y P(x, y)$ — ложным. □

Определенные на предметной области функции делятся на два класса. К одному классу относятся функции, значениями которых являются истина и ложь (или 1 и 0, поскольку именно так мы обозначаем истину и ложь). Для записи таких функций используются слова, называемые формулами.

К другому классу относятся функции, значения которых принадлежат той же предметной области, на которой эти функции определены. Слова, употребляемые для записи этих функций, называются термами. Это название происходит от латинских слов *terminus* — предел, граница и *terminalis* — конечный. О том, какие именно слова считаются термами и формулами, говорится в следующих определениях.

4.8.3. *Мы будем рассматривать слова в алфавите, состоящем из следующих пяти групп символов:*

а) ***Предметные переменные*** x_i, $i \in \{0, 1, 2, \ldots\}$.

б) ***Функциональные переменные*** f_m^n, $m, n \in \{0, 1, 2, \ldots\}$. *Число n называется* ***арностью*** *(или* ***местностью****) функциональной переменной f_m^n.*

в) ***Предикатные переменные*** P_m^n, $m, n \in \{0, 1, 2, \ldots\}$. *Число* n *называется* ***арностью*** *(или* ***местностью****) предикатной переменной* P_m^n.

г) ***Логические символы*** $\&$, $\vee$, $\supset$, $\neg$, $\forall$, $\exists$, $=$.

д) ***Служебные*** *(или* ***вспомогательные****) символы* (,), ,.

4.8.4. 1. *Каждая предметная переменная* x_i *и каждая нульарная функциональная переменная* f_i^0 *являются* ***термами****.*

2. *Если слова* $T_1, \ldots, T_n$ *являются термами, то* ***термом*** *является также слово* $f_m^n(T_1, \ldots, T_n)$.

3. *Иных термов нет.*

Нижние и верхние индексы у предметных и функциональных переменных часто опускают.

Пример. Слова

$$f_1^4(f_1^4(x_1, x_2, f_2^1(x_1), f_2^1(x_4)), x_3, x_2, x_1), \quad f_1^1(f_2^3(x_1, x_2, x_1))$$

являются термами. Не являются термами слова

$$f_1^2(x_1, P_1^2(x_1, x_2)), \quad f_1^2(x_1)\,. \qquad \square$$

Сопоставляя каждому входящему в терм функциональному символу некоторую операцию той же арности, мы сопоставим определенную операцию и самому терму.

Пример. Рассмотрим терм

$$f_1^2(x_4, f_2^2(f_3^2(x_1, x_2), x_3))\,.$$

Придадим входящим в него функциональным символам следующие значения: f_1^2 — умножение, f_2^2 — возведение первого аргумента в степень, равную второму аргументу, f_3^2 — сложение. Терм превращается в запись следующей операции:

$$x_4(x_1 + x_2)^{x_3}.$$

Полагая

$$x_1 = x_2 = 1, \quad x_3 = x_4 = 2\,,$$

получаем значение терма при заданных значениях функциональных и предметных переменных. Оно равно восьми. $\square$

4.8.5. *Пусть заданы значения всех функциональных и предметных переменных, входящих в терм. Чтобы получить значение терма, следует выполнить все операции, сопоставленные функциональным переменным.* $\square$

4.8.6. 1) *Каждый предикатный символ арности нуль является* ***формулой***.

2) *Если* P_i^n *— n-арный предикатный символ и* $T_1, \ldots, T_n$ *— термы, то слово* $P_i^n(T_1, \ldots, T_n)$ *является* ***формулой***.

3) *Если* F_1 *и* F_2 *— формулы, то и* $\neg F_1$, $(F_1 \supset F_2)$, $(F_1 \& F_2)$, $(F_1 \vee F_2)$ *являются* ***формулами***.

4) *Если* F *— формула, а* x *— входящая в нее предметная переменная, то* $\forall x F$ *и* $\exists x F$ *—* ***формулы***.

5) *Иных формул нет.*

Пример. Слова

$$(P_1^2(x_1, x_2) \& P_2^2(x_1, x_2)), \quad (\forall x_1 P_1^2(x_1, x_2) \& \exists x_1 P_2^2(x_1, x_2)),$$

$$(\forall x_1 \exists x_2 P_1^2(x_1, x_2) \vee P_2^2(x_3, x_4)).$$

являются формулами. Не являются формулами слова

$$(f_1^2(x_1, x_2) \& P_2^2(x_1, x_2)), \quad P_1^3(x, P_2^2(x, y), z),$$

$$(\forall x_1 P_1^2(x_1, x_2) \& \exists x_2 f_2^2(x_1, x_2)). \qquad \square$$

Впредь в записи формул часть скобок мы будем опускать, пользуясь прежними правилами. Для упрощения записей часто опускают также верхние и нижние индексы переменных.

4.8.7. *В формулах* $\forall x F$ *и* $\exists x F$ *формула* F *называется* ***областью действия кванторов*** $\forall x$ *и* $\exists x$ *соответственно. Вхождение предметной переменной в формулу называется* ***связанным****, если оно находится в области действия какого-либо квантора, в ином случае вхождение называется* ***свободным***.

Пример. В формуле

$$\forall x(P(x) \supset \exists y(Q(x, y, z) \& R(x, y, z)))$$

областью действия квантора $\forall x$ является формула

$$P(x) \supset \exists y(Q(x, y, z) \& R(x, y, z)),$$

а областью действия квантора $\exists y$ — формула

$$Q(x, y, z) \& R(x, y, z).$$

Все вхождения переменных x и y связанные, все вхождения переменной z свободные. $\square$

4.8.8. *Если в формуле имеется свободное (связанное) вхождение некоторой предметной переменной, то эта переменная называется* ***свободной (связанной)****.*

Из определений 4.8.7 и 4.8.8 следует, что одна и та же переменная может быть в формуле свободной и связанной одновременно.

Пример. В формуле

$$\forall x P(x,z) \,\&\, \exists x Q(x,y,z) \,\&\, \exists x \exists y R(x,y,z)$$

имеются свободные и связанные вхождения переменной y, все вхождения переменной x связанные, все вхождения переменной z свободные. □

Любая формула указывает определенный способ построения предикатов. Чтобы его реализовать, следует задать некоторую **интерпретацию**. Для этого каждому входящему в формулу предикатному символу надо сопоставить конкретный предикат той же арности и каждому входящему в нее функциональному символу сопоставить конкретную операцию соответствующей арности.

Пример. Рассмотрим формулу

$$P(f(x,y),y)\,. \tag{4.8.1}$$

Если понимать под $P(x,y)$ отношение $x > y$, а под f — операцию сложения, то формула (4.8.1) превращается в утверждение

$$x + y > y\,.$$

Если же под $P(x,y)$ подразумевать отношение $x = y$, а под f — операцию вычитания, то получим другое утверждение:

$$x - y = y\,.$$ □

Предположим, что всем входящим в формулу функциональным и предикатным переменным сопоставлены конкретные операции и предикаты. Зафиксировав универс и придав всем свободным переменным какие-то значения, мы можем определить, истинна ли формула при этих значениях или ложна.

Пример. Пусть в формуле

$$\forall x(P(f(x,y),y) \vee Q(x,y))$$

P означает отношение «больше», Q — отношение «$x \neq y$», а операция f — «умножение». Она превращается в утверждение

$$\forall x((x \cdot y > y) \vee (x \neq y))\,.$$

Универсом будем считать множество $\mathbb{N}$ натуральных чисел. Очевидно, наше утверждение справедливо, если $y \neq 1$, и ложно при $y = 1$. □

Пусть $P(x_1, \ldots, x_n)$ — предикат в смысле определения 4.8.1. В большинстве случаев интерес представляет не высказывание, сопоставленное набору значений переменных $x_1, \ldots, x_n$, а лишь то, является ли оно истинным или ложным. При этом удобнее заменить функцию P функцией Q, определенной следующим образом:

$$Q(x_1, \ldots, x_n) = \begin{cases} 1, & \text{если высказывание } P(x_1, \ldots, x_n) \text{ истинно,} \\ 0, & \text{если высказывание } P(x_1, \ldots, x_n) \text{ ложно.} \end{cases}$$

Функцию Q также называют **предикатом**. Очевидно, чтобы задать эту функцию на множестве A, достаточно указать те наборы переменных $(x_1, \ldots, x_n)$, на которых эта функция равна единице. Иначе говоря, надо задать некоторое подмножество множества A^n, называемое **областью истинности** предиката Q.

Таким образом, термин «предикат» может иметь различные значения. Мы будем придерживаться следующих теоретико-множественных определений предиката и отношения.

4.8.9. *Функция, отображающая множество A^n в множество $\{0, 1\}$, называется n-местным **предикатом** на множестве A. Подмножество множества A^n называется n-местным **отношением** на A.*

Если потребуется, будем использовать и логическое определение 4.8.1, но в этом случае будем говорить «предикат в смысле определения 4.8.1».

В приведенных выше двух примерах рассматривались интерпретации готовых формул. Чаще приходится задавать формулой какое-то отношение между элементами универса.

Пример. Обозначим через $P(x, y, z)$ и $Q(x, y, z)$ предикаты, определенные на множестве положительных целых чисел $\mathbb{N}_0$ следующим образом:

$$P(x, y, z) = 1 \iff x + y = z, \quad Q(x, y, z) = 1 \iff xy = z.$$

Утверждения (т. е. унарные предикаты) $x = 0$, $x = 1$, $x = 2$ можно задать формулами

$$\forall y P(x, y, y), \quad \forall y Q(x, y, y), \quad \exists z (\forall y Q(z, y, y) \,\&\, P(z, z, x)). \qquad \square$$

Следующие свойства формул с кванторами всеобщности и существования видны непосредственно из определения.

4.8.10. *Для любых формул $F(x)$, $G(x)$, $P(x,y)$ справедливы следующие равносильности:*

$$\begin{aligned}
\neg\forall x F(x) &\equiv \exists x \neg F(x)\,,\\
\neg\exists x F(x) &\equiv \forall x \neg F(x)\,.\\
\forall x(F(x)\&G(x)) &\equiv \forall x F(x)\&\forall x G(x)\,,\\
\exists x(F(x)\vee G(x)) &\equiv \exists x F(x)\vee\exists x G(x))\,,\\
\forall x\forall y P(x,y) &\equiv \forall y\forall x P(x,y)\,,\\
\exists x\exists y P(x,y) &\equiv \exists y\exists x P(x,y)\,.
\end{aligned}$$

□

Замечание. Кванторы всеобщности и существования переставлять в общем случае нельзя:

$$\forall x\exists y P(x,y) \not\equiv \exists y\forall x P(x,y)\,.$$

Например, пусть $P(x,y)$ означает «натуральное число x делится на натуральное число y». На множестве $\{2,3,\ldots\}$ формула $\forall x\exists y P(x,y)$ истинна, а формула $\exists y\forall x P(x,y)$ ложна. □

4.9. МНОГОЗНАЧНЫЕ ЛОГИКИ

Стандартная интерпретация исчисления высказываний основывается на предположении о том, что каждое высказывание может быть либо истинным, либо ложным. Это известный закон классической логики *tertium non datur* — третьего не дано, принцип исключенного третьего. Однако при доказательстве независимости схем аксиом $s1$, $s2$ и $s10$ из 4.6.3 приходится прибегать к интерпретациям, в которых переменные и формулы могут принимать три значения. Более того, нетрудно обнаружить, что многие из высказываний, с которыми мы имеем дело, являются или бессмысленными, или неопределенными, или правдоподобными и т. д. Рассмотрим, например, высказывание «завтра будет дождь». Его истинность или ложность в момент обнародования неизвестна, лишь на следующий день будет известно, верен ли прогноз. Значением высказывания в данный момент является *tertium*, т. е. *ни ложь, ни истина*. Очевидно, если мы хотим принимать во внимание такие высказывания, требуется по крайней мере трехзначная логика.

Если указанное выше высказывание прозвучало в прогнозе погоды, то у человека, составлявшего прогноз, можно спросить: «Какова вероятность осуществления вашего прогноза?» В ответ может прозвучать что-то вроде: «70 процентов». Это сразу наводит на мысль ввести для

прогнозов шкалу степени достоверности $0,\ 1,\ldots,\ 10$ и присвоить рассматриваемому высказыванию значение 7. Чтобы изучать все подобные высказывания, нужна десятизначная логика.

Считается, что современная теория многозначных логик возникла в 1920 году, когда Э. Пост и Лукашевич независимо опубликовали статьи, в которых такие логики рассматривались. Сразу началось интенсивное изучение этих логик. Интерес к ним возрос после обнаружения связей с теорией управляющих систем. Во второй половине XX в. выявились также важные приложения в алгебре. Таким образом, в настоящее время теорию многозначных логик разрабатывают и логики, и кибернетики, и алгебраисты.

В разд. 4.4. были введены операции отождествления и переименования переменных в формуле, а также подстановки одной функции в другую вместо некоторой переменной. Определения и утверждения 4.4.1–4.4.3 не зависят от интерпретаций и остаются справедливыми. В частности, выражение

$$\begin{aligned} &f(x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_n,y_1,\ldots,y_m)= \\ &=g(x_1,\ldots,x_{i-1},h(y_1,\ldots,y_m),x_{i+1},\ldots,x_n) \end{aligned} \tag{4.9.1}$$

по-прежнему показывает, что функция f получена из $g(x_1,\ldots,x_n)$ подстановкой вместо переменной x_i функции $h(y_1,\ldots,y_m)$ и что множество связок, используемых при построении формул, задающих функции g и h, совпадает с множеством связок, используемых при построении формулы, задающей функцию f.

Рассматривая некоторую интерпретацию, мы задаем множество и всем связкам, участвующим в построении формул, сопоставляем конкретные операции на этом множестве. В результате каждой формуле также оказывается сопоставленной операция на том же множестве, и эта операция не является булевой, если заданное множество содержит более двух элементов. Возникает следующая проблема: какие функции мы можем изобразить формулами, если задано множество связок и фиксирована интерпретация?

4.9.1. *Класс* $\mathbf{M}$ *функций, определенных на одном и том же множестве, назовем* ***замкнутым****, если любая функция, полученная либо подстановкой из двух функций, принадлежащих* $\mathbf{M}$*, либо отождествлением переменных, либо переименованием переменных из функции, принадлежащей* $\mathbf{M}$*, снова принадлежит* $\mathbf{M}$*.*

Далее мы будем рассматривать функции, определенные на множестве чисел $E_k=\{0,1,\ldots,k-1\}$, у которых все значения также

принадлежат E_k. Совокупность всех таких функций будет обозначаться через $\mathbf{P}_k$. Укажем некоторые примеры замкнутых классов.

4.9.2. *Класс $\mathbf{P}_k$ является замкнутым.* □

4.9.3. *Пусть $\mathbf{M} \subseteq \mathbf{P}_k$ — замкнутый класс. Класс $\mathbf{M}^{(s)}$ всех функций, принадлежащих $\mathbf{M}$ и принимающих не более s значений, является замкнутым.*

Доказательство. Так как класс $\mathbf{M}$ замкнутый, то из двух функций, принадлежащих $\mathbf{M}$, подстановкой мы всегда получим функцию, также принадлежащую $\mathbf{M}$. Число значений функции, возникающей в результате подстановки в функцию g вместо одной из переменных функции h, не может превышать числа значений функции g. Это означает, что если $g \in \mathbf{M}^{(s)}$, то при подстановке в нее любой функции из $\mathbf{M}^{(s)}$ также получится функция из $\mathbf{M}^{(s)}$. Очевидно также, что отождествление и переименование переменных не может увеличить число значений возникающей функции. □

4.9.4. *Функция, существенно зависящая не более чем от одного аргумента, называется* ***существенно одноместной****. Функция, существенно зависящая более чем от одного аргумента, называется* ***существенно многоместной****.*

4.9.5. *Для любого замкнутого класса $\mathbf{M}$ множество $\mathbf{M}^1$ всех существенно одноместных функций из $\mathbf{M}$ является замкнутым классом.*

Доказательство. Пусть функция $f(x_1, \ldots, x_n)$ существенно зависит лишь от x_i, а функция $g(y_1, \ldots, y_m)$ существенно зависит только от y_k. Подставим функцию g в f вместо переменной x_j. Если эта переменная фиктивная, то значения функции f от нее не зависят, и у полученной функции по-прежнему будет лишь одна существенная переменная. Если же $i = j$, то g займет место существенной переменной, и если возникающая функция будет иметь существенную переменную, то ею может быть только y_k. □

4.9.6. *Пусть $a = (\alpha_1, \ldots, \alpha_n)$, $b = (\beta_1, \ldots, \beta_n)$ — наборы чисел, принадлежащих множеству E_k, и $k > 1$. Будем считать, что*

$$a \geqslant b \iff \alpha_i \geqslant \beta_i, \quad i = \overline{1, n}\,.$$

Назовем функцию $f \in \mathbf{P}_k$ ***монотонной****, если она удовлетворяет условию*

$$(\alpha_1, \ldots, \alpha_n) \geqslant (\beta_1, \ldots, \beta_n) \Rightarrow f(\alpha_1, \ldots, \alpha_n) \geqslant f(\beta_1, \ldots, \beta_n)\,.$$

4.9.7. *Множество всех монотонных функций из* $\mathbf{P}_k$ *замкнуто.*

Доказательство. Докажем, что функция

$$g(x_1,\dots,x_{i-1},h(y_1,\dots,y_m),x_{i+1},\dots,x_n)\,,$$

полученная подстановкой из монотонных функций g и h, монотонна. Пусть

$$(\alpha_1^1,\dots,\alpha_{i-1}^1,\alpha_{i+1}^1,\dots,\alpha_n^1,\beta_1^1,\dots,\beta_m^1)\geq \\ \geq(\alpha_1^2,\dots,\alpha_{i-1}^2,\alpha_{i+1}^2,\dots,\alpha_n^2,\beta_1^2,\dots,\beta_m^2)\,,$$

тогда

$$h(\beta_1^1,\dots,\beta_m^1)\geq h(\beta_1^2,\dots,\beta_m^2)$$

и потому

$$g(\alpha_1^1,\dots,\alpha_{i-1}^1,h(\beta_1^1,\dots,\beta_m^1),\alpha_{i+1}^1,\dots,\alpha_n^1)\geq \\ \geq g(\alpha_1^2,\dots,\alpha_{i-1}^2,h(\beta_1^2,\dots,\beta_m^2),\alpha_{i+1}^2,\dots,\alpha_n^2)\,.$$

Для операций отождествления и переименования переменных доказательство аналогично. □

Замечание. Основой для построения этого примера служит естественное упорядочение чисел $0,1,2,\dots,k-1$. Однако ввести отношение частичного порядка на множестве E_k можно и иначе. Для каждого такого отношения указанным выше способом задается свой класс монотонных относительно него функций.

4.9.8. *Пусть* L *— непустое подмножество множества* E_k*, отличное от* E_k*. Обозначим через* $\mathbf{I}_k(L)$ *множество всех тех функций из* $\mathbf{P}_k$*, значения которых принадлежат* L*, если значения аргументов выбраны в* L*.*

Пример. Если $L=\{1\}$, то $\mathbf{I}_k(L)$ состоит из всех функций $f\in\mathbf{P}_k$, удовлетворяющих равенству $f(1,\dots,1)=1$. □

4.9.9. *Класс* $\mathbf{I}_k(L)$ *замкнут.*

Доказательство. Пусть $g,h\in\mathbf{I}_k(L)$ и

$$f(x_1,\dots,x_{i-1},x_{i+1},\dots,x_n,y_1,\dots,y_m)= \\ =g(x_1,\dots,x_{i-1},h(y_1,\dots,y_m),x_{i+1},\dots,x_n)\,.$$

Если значения всех аргументов функции f выбраны в L, то значение функции $h(y_1,\dots,y_m)$ принадлежит L, поэтому

$$g(x_1,\dots,x_{i-1},h(y_1,\dots,y_m),x_{i+1},\dots,x_n)\in L\,.$$ □

4.9.10. *Говорят, что множество функций $F \subseteq \mathbf{P}_k$ **порождает замкнутый класс** $\mathbf{A} \subseteq \mathbf{P}_k$, если $\mathbf{A}$ содержит те и только те функции, которые можно получить из функций, принадлежащих множеству F, с помощью подстановок. Множество функций $F \subseteq \mathbf{A}$, порождающее замкнутый класс $\mathbf{A}$, называется **полным** в $\mathbf{A}$.*

Пример. Для каждой булевой функции можно построить либо ее СДНФ, либо ее СКНФ (4.3.11, 4.3.13), поэтому система функций, содержащая конъюнкцию, дизъюнкцию и отрицание, является полной в $\mathbf{P}_2$. Конъюнкция выражается через дизъюнкцию и отрицание, а дизъюнкция выражается через конъюнкцию и отрицание, следовательно, полными в $\mathbf{P}_2$ являются системы

$$\{\&,\ \neg\}, \quad \{\vee,\ \neg\}.$$

Так как

$$x\&y \equiv (x \mid y) \mid (x \mid y), \quad x \vee y \equiv (x \mid x) \mid (y \mid y), \quad \neg x \equiv x \mid x,$$

система $\{|\}$, содержащая лишь штрих Шеффера, также является полной в $\mathbf{P}_2$.

Системы $\{\&\}$, $\{\vee\}$, $\{\neg\}$ полными в $\mathbf{P}_2$ не являются. Действительно, из конъюнкции можно получить лишь функцию x и функции вида $x_1\&x_2\&\ldots\&x_n$. Все эти функции равны единице только в том случае, когда единице равна каждая переменная. Дизъюнкция порождает функции, равные нулю лишь тогда, когда значения всех переменных равны нулю. С помощью отрицания можно получить только функции x, $\neg x$ и функции, отличающиеся от них фиктивными переменными. $\square$

Приведенные примеры показывают, что если из конечной системы функций, порождающих замкнутый класс, последовательно удалять функции, которые можно получить подстановками из оставшихся, то можно прийти к такой системе, которая порождает тот же класс, но не может быть уменьшена без потери этого свойства.

4.9.11. *Система функций $S \subset \mathbf{P}_k$, порождающая замкнутый класс $\mathbf{A}$, называется **базисом** класса $\mathbf{A}$, если никакая подсистема системы S класс $\mathbf{A}$ не порождает.*

4.9.12. *Замкнутый класс функций, имеющий конечный базис, называется **конечнопорожденным**. Замкнутый класс, не имеющий конечного базиса, называется **бесконечнопорожденным**.*

Возникают следующие вопросы:

а) Существуют ли замкнутые классы, не имеющие конечного базиса?

б) Существуют ли замкнутые классы, не имеющие никакого базиса?

в) Каково число различных замкнутых классов функций, определенных на множестве E_k?

Сначала получим частичный ответ на последний вопрос.

4.9.13. *Мощность множества всех попарно различных замкнутых классов функций, определенных на множестве E_k, не может быть больше континуума.*

Доказательство. Для каждого $n \in \mathbb{N}$ имеется лишь конечное число функций, определенных на множестве E_k и зависящих от n аргументов. Это означает, что на множестве E_k можно задать лишь счетное число функций, которые можно расположить в некоторую последовательность π. В силу сказанного любой замкнутый класс также содержит не более чем счетное число функций, которые образуют подпоследовательность последовательности π. Множество всех подпоследовательностей счетной последовательности имеет мощность континуума, следовательно, замкнутых классов функций не может быть больше. □

Введем для $i = 2, 3, \ldots$ следующие функции, определенные на E_k, $k > 2$:

$$p_i(x_1, \ldots, x_i) = \begin{cases} 1, & \text{если } x_1 = x_2 = \ldots = x_i = 0, \\ 2 & \text{в остальных случаях;} \end{cases}$$

$$q_i(x_1, \ldots, x_i) = \begin{cases} 1, & \text{если среди чисел } x_1, \ldots, x_i, \text{ нет двоек} \\ & \text{и имеется не более одной единицы,} \\ 2 & \text{в остальных случаях.} \end{cases}$$

4.9.14. *Замкнутый класс* $\mathbf{Q}$*, порождаемый всеми функциями q_i, является бесконечнопорожденным и содержит континуум различных замкнутых подклассов.*

Доказательство. Обозначим множество S через множество

$$\{q_i \mid i = 2, 3, \ldots\}.$$

Докажем, что никакую из функций, принадлежащих $S \setminus \{q_j\}$, нельзя получить подстановками, отождествлениями и переименованиями из других функций, принадлежащих $S \setminus \{q_j\}$. Пусть

$$q_j(x_1, \ldots, x_j) = q_s(t_1, \ldots, t_s), \tag{4.9.2}$$

где $t_1, \ldots, t_s$ — термы, построенные из переменных и символов функций, принадлежащих S. Очевидно, множества переменных в обеих

частях равенства должны совпадать. Предположим, что среди термов $t_1, \ldots, t_s$ по крайней мере два содержат символы функций, принадлежащих множеству S. В этом случае по меньшей мере два аргумента функции q_s не равны нулю и значение функции в правой части равенства (4.9.2) равно 2 при всех значениях аргументов. Однако функция в левой части при $x_1 = \ldots = x_j = 0$ имеет значение 1. При сделанном предположении равенство (4.9.2) не выполняется.

Пусть среди термов справа только один содержит символы функций, принадлежащих множеству S. Для упрощения записей будем считать, что этим термом является t_1. Равенство (4.9.2) приобретает вид

$$q_j(x_1, \ldots, x_j) = q_s(t_1, x_{i_2}, x_{i_3}, \ldots, x_{i_s}). \tag{4.9.3}$$

Положим $x_{i_2} = 1$, а остальным переменным придадим значение 0. Левая часть равенства (4.9.3) станет равной 1, а правая — 2, так как две переменных функции q_s не равны 0. Видим, что равенство (4.9.2) опять не выполняется.

Остается рассмотреть случай, когда все термы $t_1, \ldots, t_s$ в правой части равенства (4.9.2) являются переменными. Равенство можно теперь записать в таком виде:

$$q_j(x_1, \ldots, x_j) = q_s(x_{i_1}, x_{i_2}, \ldots, x_{i_s}). \tag{4.9.4}$$

По предположению $s \neq j$, следовательно, $s \neq j$, и так как

$$\{x_1, \ldots, x_j\} = \{x_{i_1}, \ldots, x_{i_s}\},$$

среди переменных $x_{i_1}, \ldots, x_{i_s}$ есть совпадающие. Пусть в последовательности $x_{i_1}, \ldots, x_{i_s}$ более одного раза встречается x_{i_j}. Положим $x_{i_j} = 1$, а остальные переменные сделаем равными нулю. В левой части равенства (4.9.4) будет стоять 1, а в правой — 2, что доказывает его несправедливость.

Обозначим множество всех функций q_i через T. Из сказанного выше следует, что различные подмножества множества T порождают различные замкнутые классы. В частности, никакое конечное подмножество множества T не может породить класс $\mathbf{Q}$.

Множество T счетное. Совокупность всех его конечных подмножеств имеет мощность континуума. Каждое такое подмножество порождает замкнутый класс, отличный от замкнутых классов, порождаемых другими подмножествами. Это означает, что мощность множества всех замкнутых подклассов классов $\mathbf{Q}$ и $\mathbf{P}_k \supseteq \mathbf{Q}$ не может быть меньше континуума.

В 4.9.13 было показано, что больше континуума эта мощность быть не может, следовательно, нужное утверждение доказано. □

4.9.15. *Замкнутый класс* $\mathbf{R}$*, порождаемый всеми функциями* p_i*, является бесконечнопорожденным и не имеет базиса.*

Доказательство. Предположим, что класс $\mathbf{R}$ порождается конечным множеством функций $S=\{p_{i_1},\ldots,p_{i_m}\}$. Пусть $u=\max\{i_1,\ldots,i_m\}$. Это означает, что каждая из функций системы S имеет не более u переменных. Отождествляя любые две переменные у функции p_j, мы получаем функцию p_{j-1}. Отсюда следует, что класс $\mathbf{R}$ содержит все функции p_j, у которых $j \le U$.

Нетрудно заметить, что подстановкой из двух функций, принадлежащих S, мы всегда получаем функцию, принимающую единственное значение 2, т. е. константу. При подстановке этой константы в любую функцию $p_j \in S$ также получается константа 2. Видим, что подстановками из функций, имеющих не более u существенных переменных, и констант 2 невозможно получить функцию, имеющую более u существенных переменных. Однако класс $\mathbf{R}$ содержит все функции p_i, $i=1,2,\ldots$, и все переменные любой из этих функций существенные. Это доказывает, что класс $\mathbf{R}$ не может быть конечнопорожденным.

Предположим теперь, что класс $\mathbf{R}$ имеет бесконечный базис $B = \{p_{i_1},p_{i_2},\ldots\}$. Удалим из B одну из функций, например p_{i_s}, и докажем, что полученное множество B_1 также порождает $\mathbf{R}$. Действительно, совокупность B содержит бесконечное множество функций, поэтому в ней найдется функция p_{i_m}, у которой число переменных больше, чем у p_{i_s}. Последовательно отождествляя переменные у функции p_{i_m}, получим функции

$$p_{i_m-1},\ p_{i_m-2},\ \ldots,p_{i_s}\,.$$

Это означает, что класс, порождаемый совокупностью B_1, содержит все функции, принадлежащие B, а потому и все функции из $\mathbf{R}$. Видим, что B не является базисом замкнутого класса $\mathbf{R}$. □

Все замкнутые подклассы класса $\mathbf{P}_2$ описал Э. Пост в 1941 г. Оказалось, что имеется счетное число таких подклассов и все они являются конечнопорожденными. Описать все замкнутые подклассы класса $\mathbf{P}_k$ при $k>2$ пока не удалось ввиду большого их количества (см. 4.9.13).

Еще одна проблема, касающаяся многозначных логик, особенно важна в приложениях и может быть сформулирована следующим образом: *определить, какой замкнутый класс порождают функции, принадлежащие заданному множеству* F. Частным случаем этой проблемы является следующая задача: *найти условия, необходимые и достаточные для того, чтобы множество функций* F *порождало*

класс $\mathbf{P}_k$. Она известна как **проблема полноты в многозначных логиках**. Над решением этой проблемы математики трудились много лет, и оно оказалось довольно сложным.

4.9.16. *Замкнутый класс* $\mathbf{A}$ *функций из* $\mathbf{P}_k$ *называется* ***максимальным*** *или* ***предполным****, если* $\mathbf{A} \neq \mathbf{P}_k$ *и любой другой замкнутый класс, содержащий все функции из* $\mathbf{A}$*, совпадает либо с* $\mathbf{A}$*, либо с* $\mathbf{P}_k$*.*

4.9.17. *Если* $L \subset E_k$*,* $L \neq \varnothing$*,* $L \neq E_k$*, то* $\mathbf{I}_k(L)$ *является максимальным замкнутым классом.*

Доказательство. Если $g(x_1, \ldots, x_s) \notin \mathbf{I}_k(L)$, то в L найдутся такие элементы $a_1, \ldots, a_s$, что

$$g(a_1, \ldots, a_s) = b \notin L\,.$$

Обозначим через $\mathbf{A}$ замкнутый класс, порожденный функцией g и функциями из $\mathbf{I}_k(L)$. Докажем, что $\mathbf{A} = \mathbf{P}_k$.

Так как функции $c_{a_1}(x) = a_1, \ldots, c_{a_s}(x) = a_s$ принадлежат $\mathbf{I}_k(L)$, то

$$g(c_{a_1}(x), \ldots, c_{a_s}(x)) = c_b(x)$$

содержится в $\mathbf{A}$.

Пусть $h(x_1, \ldots, x_n) \in \mathbf{P}_k$. Сопоставим ей функцию

$$f_h(x_0, x_1, \ldots, x_n) = \begin{cases} h(x_1, \ldots, x_n), & \text{если } x_0 = b, \\ a_1, & \text{если } x_0 \neq b. \end{cases}$$

Очевидно, $f_h \in \mathbf{I}_k(L)$. Однако

$$h(x_1, \ldots, x_n) = f_h(c_b(x_1), x_1, \ldots, x_n)\,,$$

поэтому $h \in \mathbf{A}$. Видим, что каждая функция из $\mathbf{P}_k$ принадлежит $\mathbf{A}$. Это означает, что $\mathbf{P}_k = \mathbf{A}$. □

Отметим, что помимо максимальных замкнутых классов, указанных в 4.9.17, в $\mathbf{P}_k$ имеются также другие максимальные замкнутые подклассы и их общее количество стремительно возрастает с ростом k.

4.9.18. *Для того чтобы система функций* S *порождала класс* $\mathbf{P}_k$*, необходимо и достаточно, чтобы эта система не содержалась целиком ни в каком максимальном замкнутом классе.*

Доказательство. ($\Rightarrow$) Если все принадлежащие системе S функции содержатся в некотором максимальном замкнутом классе $\mathbf{A}$, то S порождает какой-то подкласс класса $\mathbf{A}$ и потому не может породить класс $\mathbf{P}_k$.

($\Leftarrow$) Эту часть доказательства мы опустим. □

Глава 5

КОНЕЧНЫЕ АВТОМАТЫ

5.1. ЯЗЫКИ

Дисциплина, предметом которой являются разработка и изучение понятий, образующих основу формального аппарата для описания строения естественных языков, называется **математической лингвистикой**. В настоящее время математическую лингвистику можно рассматривать как один из разделов математической логики. Этот раздел имеет как теоретическое, так и прикладное значение.

5.1.1. *Буквой называется элементарный знак, рассматриваемый вне зависимости от выражаемого им смысла. Алфавитом называется множество букв.*

Буквы считаются неделимыми объектами в том смысле, что мы не изучаем части букв. В некоторых случаях буквы называют *символами*. Если не оговорено противное, мы будем рассматривать конечные алфавиты. Иногда нам придется вводить алфавиты бесконечные. Для обозначения алфавитов мы будем использовать прописные латинские буквы, а для обозначения букв будем применять как прописные, так и строчные латинские буквы. Иногда буквы будут обозначаться и другими знаками.

Примеры. Разумеется, латинский и русский алфавиты, записанные какими-нибудь фиксированными шрифтами, удовлетворяют определению 5.1.1. Алфавитами являются также множества $\{0,1\}$ и $\{\cdot,-\}$, содержащие по две буквы. □

5.1.2. *Последовательность букв, принадлежащих некоторому алфавиту, называется **словом** или **цепочкой** в этом алфавите.*

*Слово, не содержащее ни одной буквы, называется **пустым**. Оно обозначается символом Λ и является словом в любом алфавите.*

Для определенности будем считать, что запись букв любого непустого слова происходит слева направо и потому в нем есть буква, крайняя ***буква, крайняя слева*** *и* ***буква, крайняя справа****.*

5.1.3. *Количество букв в слове называется* ***длиной слова****. Слово длины n, имеющее вид $a \dots a$ и не содержащее букв, отличных от a, будет обозначаться через a^n.*

Примеры. Согласно этому определению $\cdot - \cdot\cdot$ и $- \cdot - \cdot$ являются словами длины 4 в алфавите $\{\cdot, -\}$, а $!!()))$ — слово длины 6 в алфавите $\{!, (,)\}$. □

Мы будем считать, что слово всегда имеет конечную длину. В математической лингвистике обычно говорят о **цепочках**, а не о словах, так как цепочки символов считаются предложениями.

5.1.4. *Пусть имеются два слова α и β в алфавите A. Записав все принадлежащие слову β буквы с соблюдением их порядка правее последней буквы слова α, мы получим слово $\alpha\beta$, называемое* ***соединением*** *(или* ***конкатенацией, сцеплением, произведением****) слов α и β.*

5.1.5. *Слово β называется* ***подсловом*** *слова δ, если слово δ получается соединением трех слов α, β, γ, т. е. если $\delta = \alpha\beta\gamma$. Может оказаться, что слова α и γ определены не однозначно. В этом случае возможные варианты слова α упорядочивают по длине, начиная с самого короткого: $\alpha_1, \alpha_2, \dots$, и говорят о* ***первом вхождении*** *слова β в слово δ,* ***втором****, . . . ,* ***последнем вхождении****.*

Для того чтобы указать конкретное вхождение подслова в слово, используют звездочки.

Пример. Если $\delta = abcbccba$, то запись $ab * cb * ccba$ указывает на первое вхождение подслова cb в слово δ, а запись $abcbc * cb * a$ — на второе вхождение. □

5.1.6. *Произвольное множество цепочек (конечной длины) в алфавите A называется* ***языком*** *в этом алфавите.*

Пример. Пустое множество цепочек и множество, состоящее из единственной цепочки Λ, являются языками в любом алфавите. Цепочки из точек и тире, образующие азбуку Морзе, в совокупности являются языком в алфавите $\{\cdot, -\}$. Множество {папа, мама} образуют язык в алфавите {а,м,п}. □

5.2. ГРАММАТИКИ

В примерах, приведенных после определения 5.1.6, языки конечные, и их можно описать, перечислив все входящие в них слова. Задача описания бесконечных языков является намного более сложной и решается различными путями. В математической логике одним из основных способов задания множества является построение некоторого исчисления. Для этого указываются исходные элементы (аксиомы) и задаются правила вывода, описывающие, как строить новые элементы множества из исходных и ранее построенных. Исчисления, используемые в математической лингвистике, называются **формальными грамматиками**. Формальные грамматики разделяются на несколько типов, из которых часть будет описана в этом разделе.

5.2.1. *Порождающей грамматикой называется упорядоченная четверка* $\Gamma = \langle V, W, I, R \rangle$. *В этой четверке* V *и* W — *непересекающиеся конечные множества, называемые соответственно* ***основным*** *и* ***вспомогательным алфавитами*** *или* ***словарями***. *Элементы этих множеств называются соответственно* ***основными (терминальными)*** *и* ***вспомогательными (нетерминальными) символами*** *или же* ***терминалами*** *и* ***нетерминалами***. *Через* I *обозначен элемент из* W, *называемый* ***начальным символом***. *Буквой* R *обозначено конечное множество слов вида* $\alpha \to \beta$, *называемых* ***правилами***, *в которых* α *и* β — *цепочки в алфавите* $V \cup W$, *а буква* $\to$ *не принадлежит этому алфавиту. Множество правил* R *называется* ***схемой грамматики***.

Как правило, в дальнейшем строчными латинскими буквами будут обозначаться основные, а прописными — вспомогательные символы алфавитов V и W.

5.2.2. *Если две цепочки* φ *и* ψ *представимы в виде*

$$\varphi = \gamma\alpha\delta, \quad \psi = \gamma\beta\delta$$

и $\alpha \to \beta$ — *одно из правил грамматики* Γ, *то говорят, что цепочка* ψ ***непосредственно выводима*** *из* φ *в* Γ. *Показать, что цепочка* ψ *непосредственно выводима в* Γ *из* φ *можно тремя способами:*

$$\varphi \underset{\Gamma}{\vdash} \psi; \quad \varphi \vdash \psi; \quad \varphi \Rightarrow \psi\,.$$

Пример. Пусть $\Gamma = \langle V, W, I, R \rangle$ и

$$V = \{a, b, c\},\ W = \{I, D\},\ R = \{I \to D,\ D \to ab,\ cac \to bbab\}\,.$$

Цепочка $aacacacb$ содержит два вхождения подслова cac. Поскольку грамматика содержит правило $cac \to bbab$, любое из этих вхождений можно заменить словом $bbab$. В первом случае получаем слово $aabbabacb$, во втором — $aacabbabb$. Оба этих слова непосредственно выводимы из $aacacacb$:

$$aacacacb \vdash aabbabacb, \quad aacacacb \vdash aacabbabb\,. \qquad \square$$

5.2.3. *Последовательность цепочек $\delta_0, \delta_1, \ldots, \delta_n$, каждый член которой начиная с δ_1 непосредственно выводим из предыдущего, называется* ***выводом*** *цепочки δ_n из δ_0 в Γ. Число n называется* ***длиной вывода****. Если существует вывод в Γ цепочки β из цепочки α, то говорят, что β* ***выводима из α в Γ****.*

5.2.4. *Вывод $\delta_0, \delta_1, \ldots, \delta_n$ называется* ***полным****, если $\delta_0 = I$ и δ_n не содержит вспомогательных символов. Последняя цепочка в полном выводе называется* ***выводимой*** *(в Γ).*

5.2.5. ***Размеченным выводом*** *называется вывод, в котором в каждой цепочке при помощи звездочек указано вхождение подслова, заменяемого на очередном шаге вывода.*

Пример. Полный вывод цепочки $ababbabb$ в грамматике

$$\Gamma = \langle V, W, I, R \rangle, \quad V = \{a, b\}, \quad W = \{I, D\},$$

$$R = \{I \to ab,\ I \to aDIb,\ d \to bIb,\ DI \to b,\ D \to \Lambda\}$$

можно записать в виде последовательности цепочек

$$I, \quad aDIb, \quad aDabb, \quad abIbabb, \quad ababbabb,$$

а также в следующем виде:

$$I \Rightarrow aDIb \Rightarrow aDabb \Rightarrow abIbabb \Rightarrow ababbabb.$$

Размеченный вывод

$$*I*, \quad aD*I*b, \quad a*D*abb, \quad ab*I*babb, \quad ababbabb$$

позволяет на каждом шаге быстро определить, какое правило применялось и к какому подслову. $\square$

5.2.6. *Множество цепочек в основном алфавите, выводимых в Γ из I, называется* ***языком, порождаемым грамматикой*** *Γ, и обозначается через $L(\Gamma)$. Две порождающих грамматики называются* ***эквивалентными****, если они порождают один и тот же язык.*

Пример. Пусть

$$V = \{0, 1\}, \quad W = \{I\}, \quad R = \{I \to 0I0,\ I \to 1\}.$$

Грамматика $\Gamma = \langle V, W, I, R\rangle$ порождает язык $L(\Gamma) = \{0^n 10^n\}$, $n = 0, 1, 2, \ldots$. В этой грамматике последовательность

$$I,\ 0I0,\ 00I00,\ 000I000,\ 0001000$$

является полным выводом цепочки 0001000.

Обозначим через Γ_1 грамматику, полученную из Γ добавлением правил $I \to 00I00$, $0I0 \to 00100$. Язык, порождаемый грамматикой Γ_1, совпадает с $L(\Gamma)$, следовательно, грамматики Γ и Γ_1 эквивалентны. □

5.2.7. *Порождающая грамматика $\Gamma = \langle V, W, I, R\rangle$, в которой каждое правило имеет вид $\gamma_1 A \gamma_2 \to \gamma_1 \beta \gamma_2$, где $\gamma_1, \gamma_2, \beta$ — цепочки в алфавите $V \cup W$, $A \in W$, $\beta \neq \Lambda$, называется* ***грамматикой составляющих (грамматикой непосредственно составляющих, контекстной грамматикой)****. Порождаемый такой грамматикой язык называется* ***НС-языком****.*

При применении правила вывода, указанного в этом определении, к некоторому слову в этом слове происходит замена вспомогательного символа A цепочкой β, но лишь там, где этот символ входит в подслово $\gamma_1 A \gamma_2$. Таким образом, возможность замены обусловлена некоторым «контекстом», откуда и произошло одно из названий грамматики.

5.2.8. *Грамматика составляющих, в которой все правила имеют вид $A \to \beta$, где A — вспомогательный символ и β — непустая цепочка в алфавите $V \cup W$, называется* ***бесконтекстной*** *(или* ***контекстно-свободной****). Правила бесконтекстных грамматик и языки, порождаемые такими грамматиками, называются* ***бесконтекстными****.*

Пример. Описанная в примере к определению 5.2.6 грамматика Γ является бесконтекстной. □

5.2.9. ***Автоматной грамматикой*** *называется бесконтекстная грамматика, в которой каждое правило имеет либо вид $A \to xB$, либо вид $A \to x$, где A и B — вспомогательные символы, x — один из основных символов.*

Отметим следующие важные свойства автоматных грамматик.

5.2.10. 1. *За исключением последней, каждая принадлежащая полному выводу цепочка содержит ровно один вспомогательный символ, и этот символ является в цепочке последним.*

2. *Если на некотором шаге использовано правило вида $A \to xB$, то все основные символы, добавляемые к полученной цепочке в дальнейшем, будут находиться правее символа x.*

3. *После применения правила вида $A \to x$ вывод заканчивается, так как в полученной цепочке нет вспомогательных символов и к ней нельзя применить ни одно из правил грамматики.*

Доказательство. Первая цепочка в полном выводе состоит из единственного начального вспомогательного символа. Применение правила вида $A \to xB$ не увеличивает числа вспомогательных символов, а после применения правила вида $A \to x$ в цепочке остаются только основные символы. Это доказывает утверждение 1. Остальные утверждения очевидны. □

5.3. ДЕРЕВЬЯ ВЫВОДОВ

В грамматике составляющих каждое правило имеет вид

$$\gamma_1 A \gamma_2 \to \gamma_1 \beta \gamma_2 \,,$$

где A — вспомогательный символ и β — непустая цепочка. Это позволяет представить полный вывод любой цепочки в виде помеченного дерева. Пусть $I, \delta_1, \ldots, \delta_n$ — полный вывод цепочки δ_n в грамматике составляющих Г. Будем предполагать, что цепочка δ_1 имеет вид $a_1 \ldots a_k$.

Построение дерева, называемого **растянутым деревом вывода**, начнем с вершины, которую пометим начальным символом I. Очевидно, на первом шаге применено правило $I \to a_1 \ldots a_k$. Добавим к графу, состоящему из вершины I, k новых вершин и столько же ребер, соединяющих эти вершины с вершиной I. Добавленные вершины пометим символами $a_1, \ldots, a_k$. Изображая дерево на рисунке, для наглядности следует располагать вершины $a_1, \ldots, a_k$ на одной линии так, чтобы они находились ниже начальной вершины и чтобы для любых $i, j \in \in \{1, \ldots, k\}$ при $i < j$ вершина a_i стояла левее вершины a_j.

Если длина вывода больше единицы, то среди символов $a_1, \ldots, a_k$ непременно есть вспомогательные, так как левая часть каждого правила рассматриваемой грамматики содержит лишь один вспомогательный символ. Предположим, что $a_i \in W$ и что на втором шаге применялось правило $\gamma_1 a_i \gamma_2 \to \gamma_1 \beta \gamma_2$, в котором

$$\gamma_1 = a_{i-p} \ldots a_{i-1}, \quad \gamma_2 = a_{i+1} \ldots a_{i+q}, \quad \beta = a_{21} \ldots a_{2m} \,.$$

Тогда цепочка δ_2 имеет вид

$$\delta_2 = a_1 \dots a_{i-1} a_{21} \dots a_{2m} a_{i+1} \dots a_k \,.$$

Добавим к уже построенному графу еще одну строчку из $k + m - 1$ вершин (столько символов входит в δ_2). На рисунке расположим добавленные вершины так, чтобы они находились ниже вершин, нарисованных ранее. Пометим эти вершины символами

$$a_1, \dots, a_{i-1}, a_{21}, \dots, a_{2m}, a_{i+1}, \dots, a_k \,. \tag{5.3.1}$$

Пометки вершин должны располагаться слева направо в таком порядке, в каком они идут в цепочке (5.3.1). Соединим вершины

$$a_1, \dots, a_{i-1}, a_{i+1} \dots a_k$$

с одноименными вершинами графа, построенного на предыдущем шаге. Вершины $a_{21} \dots a_{2m}$ соединим ребрами с вершиной a_i.

Так шаг за шагом будет построено все растянутое дерево вывода.

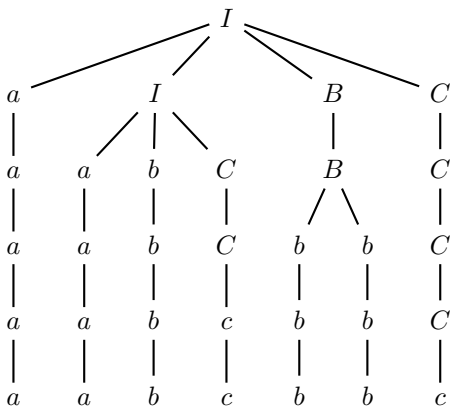


Рис. 5.1

Пример. Изображенное на рис. 5.1 растянутое дерево соответствует размеченному выводу

$$I,\ a*I*BC,\quad aab*CBC*,\quad aa*bC*bbC,$$
$$aabcb*bC*,\quad aabcbbc \tag{5.3.2}$$

в грамматике составляющих со следующими правилами:

$$I \to aIBC,\ I \to abC,\ CBC \to CbbC,\ bC \to bc.$$

Нетрудно заметить, что слова, образованные теми буквами, которыми помечены вершины, находящиеся на одном и том же уровне в дереве на рис. 5.1, принадлежат выводу (5.3.2) и совокупность всех таких слов и образует этот вывод. □

Применим теперь к растянутому дереву вывода несколько раз операцию стягивания ребра. Стягивать будем те и только те ребра, которые удовлетворяют следующим двум условиям:

а) концы ребра помечены одной и той же буквой;

б) степень каждого конца ребра не превышает двух.

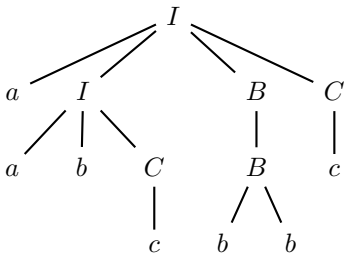


Рис. 5.2

Пример. После стягивания по приведенным правилам ребер растянутого дерева вывода из предыдущего примера получим **дерево вывода**, изображенное на рис. 5.2. □

В бесконтекстной грамматике дерево вывода может быть построено и непосредственно, без предварительного построения растянутого дерева вывода. Для этого целесообразно придерживаться следующих правил:

а) Корень дерева помечается символом, с которого начинается вывод. В случае полного вывода этим символом является буква I.

б) Добавляемые на очередном шаге вершины располагаются на горизонтальной линии и находятся ниже той вершины, с которой они соединяются ребрами.

в) Предположим, что на шаге k уже построено дерево G и теперь мы должны применить правило $A \to b_1 \dots b_s$. Это означает, что определенное вхождение символа A заменяется цепочкой $b_1 \dots b_s$. Ищем в графе G вершину, соответствующую данному вхождению символа

A, добавляем s вершин и соединяем их ребрами с найденной вершиной. Двигаясь слева направо, помечаем новые вершины символами $b_1, \ldots, b_s$. Переходим к шагу $k+1$.

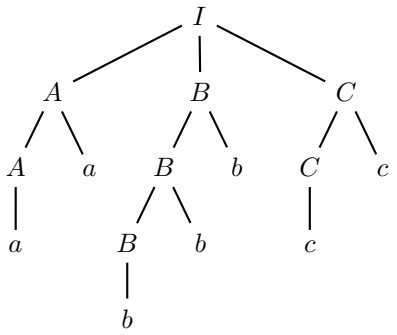


Рис. 5.3

Пример. Рассмотрим бесконтекстную грамматику со следующими правилами:

$$I \to ABC,\ A \to Aa,\ A \to a,\ B \to Bb,$$
$$B \to b,\ C \to Cc,\ C \to c.$$

Размеченный вывод

$$I,\ A*B*C,\ A*B*bC,\ A*B*bbC,$$
$$*A*bbbC*,\ *A*abbbC,\ aabbb*C*,$$
$$aabbb*C*c,\ aabbbcc$$

может быть представлен в виде дерева, изображенного на рис. 5.3. □

5.4. КОНЕЧНЫЕ АВТОМАТЫ

Конечные автоматы представляют собой математические модели устройств, преобразующих дискретную информацию. Схематически подобное устройство изображено на рис. 5.4. Информация в виде каких-то сигналов поступает на входы $x_1, \ldots, x_n$, обрабатывается процессором P, и результат подается на выходы $y_1, \ldots, y_k$. Время обработки не учитывается, поэтому можно считать, что сигналы на выходе появляются одновременно с поступлением входных сигналов. Входные сигналы дискретны и поступают на входы не непрерывно, а в некоторые моменты времени, называемые иногда **тактовыми**. Природа входных и

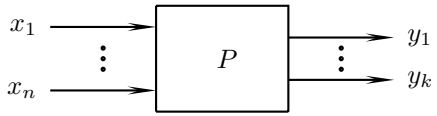


Рис. 5.4

выходных сигналов может быть совершенно произвольной. Это могут быть электрические импульсы, вспышки света и т. д. Для построения математической модели важно лишь уметь отличать каждый сигнал от любого другого. Мы будем обозначать сигналы, поступающие на входы $x_1, \ldots, x_n$, буквами некоторых алфавитов $A_1, \ldots, A_n$, а выходные сигналы буквами алфавитов $B_1, \ldots, B_n$. Все эти алфавиты должны быть конечными.

Если при подаче на входы в любые два момента времени одинаковых наборов букв на выходе также будем получать одинаковые наборы, то работа нашего устройства может быть описана совокупностью уравнений

$$\begin{aligned} y_1 &= f_1(x_1, \ldots, x_n), \\ &\ldots\ldots\ldots\ldots\ldots \\ y_k &= f_k(x_1, \ldots, x_n). \end{aligned}$$

Так как входные и выходные алфавиты конечны, каждую из функций можно задать таблицей. В этом случае мы говорим, что наше **устройство не имеет памяти**.

Более интересен случай, когда устройство имеет память, т. е. появление той или иной буквы на выходе зависит не только от букв, поданных в данный момент времени на входы, но и от того, какие буквы подавались на входы ранее. Состояние памяти в определенный момент времени мы назовем **состоянием** самого устройства. Состояния будем обозначать буквами некоторого конечного алфавита S. Поскольку все состояния памяти задаются при построении устройства и изменяться в процессе работы не могут, назовем ее **внутренней**.

Если не предполагается строить сложные схемы, соединяя выходы одних устройств со входами других, то можно ограничиться изучением устройств с одним входом и одним выходом. Например, пусть устройство имеет 3 входа и 2 выхода и на эти входы и выходы поступают нули и единицы. Иначе говоря, на вход поступают тройки, а на выход — двойки, состоящие из нулей и единиц. Эти тройки и двойки можно

обозначить буквами алфавитов $A = \{a_1, \ldots, a_8\}$ и $B = \{b_1, \ldots, b_4\}$ (табл. 5.1, 5.2).

Таблица 5.1

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
x_1	0	1	0	0	0	1	1	1
x_2	0	0	1	0	1	0	1	1
x_3	0	0	0	1	1	1	0	1

Таблица 5.2

	b_1	b_2	b_3	b_4
y_1	0	0	1	1
y_2	0	1	0	1

Теперь рассматриваемое устройство можно заменить устройством с одним входом и одним выходом, и подавать на его вход буквы алфавита A, получая на выходе буквы алфавита B.

Поскольку нас интересует не физическая реализация устройства, а лишь математическое описание его функционирования, с учетом сказанного выше мы приходим к такому определению.

5.4.1. *Система $\langle A, S, B, \varphi, \psi\rangle$, в которой A, S, B — конечные алфавиты, φ и ψ — функции, отображающие множество $S \times A$ соответственно в S и в B, называется* ***конечным автоматом****, или* ***автоматом Мили****. Множества A, S, B называются соответственно* ***входным алфавитом****,* ***множеством состояний*** *и* ***выходным алфавитом****. Отображения φ и ψ называются* ***функцией переходов*** *и* ***функцией выходов****.*

Отказавшись в определении 5.4.1 от требования конечности алфавитов A, S, B, получим определение **бесконечного автомата**. Отказ от конечности алфавитов приводит к существенному усложнению теории, поэтому мы их рассматривать не будем.

5.4.2. ***Командой автомата*** *$\langle A, S, B, \varphi, \psi\rangle$ называется четверка вида*

$$(s, a, \varphi(s, a), \psi(s, a)),$$

в которой $a \in A$, $s \in S$. Ее удобно записывать в виде

$$sa \to \varphi(s,a)\psi(s,a)\,.$$

Слово, стоящее в команде левее стрелки, назовем ***левой частью команды*** *автомата. Автомат не должен иметь двух различных команд с совпадающими левыми частями.*

Впредь мы не будем различать конечный автомат и устройство, в виде которого он реализован. Работа конечного автомата $\mathcal{A} = \langle A, S, B, \varphi, \psi\rangle$ происходит следующим образом. Имеются моменты времени, в которые он может воспринимать одну из букв на входе, находиться в одном из состояний и подавать на выход букву, принадлежащую выходному алфавиту. Эти моменты времени дискретны, т. е. разделены промежутками, и называются **тактовыми моментами**.

Обычно предполагается, что имеется момент, когда автомат начинает работу. Считая этот тактовый момент первым, остальные можно перенумеровать и говорить, например, о состоянии автомата в момент t или $t+k$, где $t, k \in \mathbb{N}$.

Предположим, что в один из тактовых моментов на вход автомата подается буква $a \in A$, $s \in S$ является его внутренним состоянием и

$$sa \to \varphi(s,a)\psi(s,a)$$

— команда автомата. В соответствии с этой командой в следующий тактовый момент автомат перейдет в состояние $\varphi(s,a)$, и на выходе появится буква $\psi(s,a)$. На вход подается следующая буква, автомат срабатывает, переходит в состояние, определяемое подходящей командой, и подает на выход соответствующую букву, и т. д.

Если на вход автомата буква за буквой подать некоторое слово, то на выходе появится также некоторое слово. Ввиду этого несколько упрощенно описанный автомат можно представить себе как устройство, определенным образом преобразующее слова во входном алфавите в слова в выходном алфавите.

5.4.3. *Слова во входном алфавите автомата называются* ***входными словами****, а слова в выходном алфавите —* ***выходными словами****. Через A^* и B^* будут обозначаться соответственно множество всех входных и множество всех выходных слов автомата $\langle A, S, B, \varphi, \psi\rangle$. Будем считать, что каждое из множеств A^* и B^* содержит пустое слово Λ.*

Нетрудно, однако, заметить, что, подав несколько раз на вход автомата некоторое слово, мы можем получить на выходе разные слова.

Такое явление объясняется тем, что появление каждой выходной буквы зависит не только от буквы на входе, но и от состояния автомата. Устранить указанную неоднозначность можно, если предполагать, что в начале работы автомат всегда находится в одном и том же состоянии.

5.4.4. *Автомат с выделенным начальным состоянием называется* ***инициальным****. Инициальный автомат $\mathcal{A}$ с начальным состоянием s_1 будет обозначаться либо через $\mathcal{A}_{s_1}$, либо через $(\mathcal{A}, s_1)$.*

Термин «инициальный» происходит от латинских слов *initialis* — первоначальный, *initiare* — начинать.

Если появление выходных букв автомата A зависит только от букв, поступающих на вход, и не зависит от его состояний, то, как уже говорилось, этому автомату можно сопоставить функцию, отображающую каждую букву входного алфавита в соответствующую выходную букву. Можно сказать, что эта функция описывает поведение автомата A. Функционирование автоматов, у которых буквы на выходе зависят как от букв на входе, так и от состояний, задается при помощи двух функций: функции переходов и функции выходов. Поведение автомата можно описать одной функцией, воспользовавшись приемом, указанным ниже.

5.4.5. *Доопределим функцию переходов и функцию выходов автомата $\mathcal{A}_{s_1} = \langle A, S, B, \varphi, \psi\rangle$ следующим образом: для любых $a \in A$, $s \in S$, пустого слова Λ и любого входного слова $\alpha \in A^*$ положим*

$$\varphi(s, \Lambda) = s, \quad \psi(s, \Lambda) = \Lambda,$$

$$\varphi(s, \alpha a) = \varphi(\varphi(s, \alpha), a), \quad \psi(s, \alpha a) = \psi(\varphi(s, \alpha), a).$$

Обозначим через $\pi_i(\alpha)$ подслово слова $\alpha \in A^*$, образованное первыми i буквами слова α. Пусть $\Psi_{\mathcal{A}_{s_1}}$ — двуместная функция, определенная на множестве $S \times A^*$ следующим образом: если $s \in S$, $\alpha \in A^*$ и $|\alpha| = n$, то

$$\Psi_{\mathcal{A}_{s_1}}(s, \alpha) = \psi(s, \pi_1(\alpha))\psi(s, \pi_2(\alpha)) \ldots \psi(s, \pi_n(\alpha)).$$

Определенная таким образом функция $\Psi_{\mathcal{A}_{s_1}}$, отображающая A^* в B^*, называется ***поведением инициального автомата*** *$\mathcal{A}_{s_1}$. Эту функцию называют также* ***функцией, вычислимой автоматом*** *$\mathcal{A}_{s_1}$.*

Очевидно, $\varphi(s, \alpha)$ — состояние, в котором окажется автомат $\mathcal{A}_{s_1}$ после обработки слова α, а $\psi(s, \alpha)$ — буква, которая будет на выходе после такой обработки при условии, что автомат начинал работу,

находясь в состоянии s. Функция $\Psi_{\mathcal{A}_{s_1}}$ отображает множество входных слов в множество выходных слов. Стоит отметить, что иногда поведением автомата считают множество значений функции $\Psi_{\mathcal{A}_{s_1}}$ или некоторые другие множества, связанные с автоматом $\mathcal{A}_{s_1}$.

5.5. СПОСОБЫ ЗАДАНИЯ КОНЕЧНЫХ АВТОМАТОВ

Автомат $\langle A, S, B, \varphi, \psi\rangle$ задан, если зафиксированы алфавиты A, S, B и определены функции φ и ψ. Поскольку указанные алфавиты конечны, конечными являются и области определения, и множества значений функций φ и ψ, что позволяет задавать эти функции матрицами (таблицами). Пусть $S = \{s_1, \ldots, s_m\}$ и $A = \{a_1, \ldots, a_n\}$. Тогда для $i = \overline{1, m}$ и $j = \overline{1, n}$ на пересечении i-й строки и j-го столбца в матрице, называемой **матрицей переходов**, стоит буква $\varphi(s_i, a_j)$, а в матрице, называемой **матрицей выходов**, — буква $\psi(s_i, a_j)$. Очевидно, эти две матрицы полностью задают функции φ и ψ.

Пример. Построим инициальный автомат, перерабатывающий слово «тетя» в слово «бука». Задаем входной алфавит $A = \{$т,е,я$\}$ и выходной $B = \{$б,у,к,а$\}$. Начальное состояние автомата обозначим через s_1. Очевидно, $\psi(s_1, \text{т}) = \text{б}$. Автомат оставим в прежнем состоянии, т. е. положим $\varphi(s_1, \text{т}) = s_1$. Следующая буква «е» должна превратиться в «у», поэтому $\psi(s_1, \text{е}) = \text{у}$. Сменим состояние автомата, так как очередная буква «т» уже встречалась ранее, а реагировать на нее надо по-новому. Положим

$$\varphi(s_1, \text{е}) = s_2, \quad \psi(s_2, \text{т}) = \text{к}, \quad \varphi(s_2, \text{т}) = s_2,$$

$$\psi(s_2, \text{я}) = \text{а}, \quad \varphi(s_2, \text{я}) = s_1.$$

Значения функций φ и ψ при остальных значениях аргументов можно выбрать произвольно, так как это не повлияет на выполнение автоматом поставленной задачи. Например, можно считать, что во всех остальных случаях значение функции φ равно s_1, а функции ψ — б. Получаем табл. 5.3, в которой левая часть содержит значения функции φ, а правая — ψ. □

Вместо выписывания матриц можно составить список всех команд автомата. Этот способ удобен при небольшом числе команд.

Пример. Автомат, описанный в предыдущем примере, имеет следующие команды:

$$s_1\text{т} \to s_1\text{б}, \quad s_2\text{т} \to s_2\text{к}, \quad s_1\text{я} \to s_1\text{б},$$
$$s_1\text{е} \to s_2\text{у}, \quad s_2\text{я} \to s_1\text{а}, \quad s_2\text{е} \to s_2\text{б}.$$

Таблица 5.3

φ	т	е	я		ψ	т	е	я
s_1	s_1	s_2	s_1		s_1	б	у	б
s_2	s_2	s_2	s_1		s_2	к	б	а

Для того чтобы автомат перерабатывал слово «тетя» в слово «бука», достаточно команд, содержащихся в первых двух столбцах. Дополнительные две команды превращают автомат во всюду определенный, т. е. такой, у которого для любого состояния s и любой буквы t из входного алфавита есть команда с левой частью st. □

Будем считать, что автомат начинает работу в момент времени $t = 1$ и, поскольку время предполагается дискретным, последующие тактовые моменты обозначим цифрами 2, 3, 4, ... Обозначим через $a(i)$ и $b(i)$ буквы, поступающие соответственно на вход и на выход автомата на такте i, а через $s(i)$ состояние автомата на том же такте. Поскольку автомат имеет конечную память, в некоторых случаях функционирование инициального автомата $\mathcal{A}_{s_1} = \langle A, S, B, \varphi, \psi\rangle$ может быть описано уравнениями

$$\begin{cases} s(1) & = s_1, \\ s(t+1) & = \varphi(s(t), a(t)), \\ b(1) & = \alpha_1, \\ \dots\dots\dots & \dots\dots\dots\dots\dots \\ b(k+1) & = \alpha_k, \\ b(t) & = \psi(s(t), a(t-k), a(t-k+1), \dots, a(t)). \end{cases} \tag{5.5.1}$$

5.5.1. *Равенства* (5.5.1) *называются* ***каноническими уравнениями*** *инициального автомата.*

Зная канонические уравнения, по всякому входному слову легко вычислить выходное слово.

Пример. Рассмотрим автомат, определяемый каноническими уравнениями

$$\begin{aligned} s(1) &= 0, \\ b(t) &= a(t) + 2s(t) + 3, \\ s(t+1) &= a(t) + s(t) (\mathrm{mod}\ 3), \end{aligned}$$

со входным алфавитом $\{0,1,2\}$, множеством состояний $\{0,1\}$ и выходным алфавитом $\{3,4,5,7\}$. Подадим на вход автомата слово 0102 и проследим 4 такта в его работе:

$$
\begin{aligned}
&s(1)=0,\ a(1)=0,\ a(2)=1,\ a(3)=0,\ a(4)=2,\\
&b(1)=a(1)+2s(1)+3=0+0+3=3,\\
&s(2)=a(1)+s(1)(\mathrm{mod}\ 3)=0+0(\mathrm{mod}\ 3)=0,\\
&b(2)=a(2)+2s(2)+3=1+0+3=4,\\
&s(3)=a(2)+s(2)(\mathrm{mod}\ 3)=1+0(\mathrm{mod}\ 3)=1(\mathrm{mod}\ 3),\\
&b(3)=a(3)+2s(3)+3=0+2+3=5,\\
&s(4)=a(3)+s(3)(\mathrm{mod}\ 3)=0+1(\mathrm{mod}\ 3)=1,\\
&b(4)=a(4)+2s(4)+3=2+2+3=7.
\end{aligned}
$$

Видим, что входное слово 0102 он превратит в выходное слово 3457. Перекодируем входной и выходной алфавиты следующим образом:

$$0=\text{т},\quad 1=\text{е},\quad 2=\text{я},\quad 3=\text{б},\quad 4=\text{у},\quad 5=\text{к},\quad 7=\text{а}.$$

Слово 0102 превращается в «тетя», слово 3457 — в «бука», следовательно, этот автомат решает ту же задачу, что и автомат, описанный в последнем примере. □

Задание автомата при помощи графа (диаграммы) является одним из наиболее наглядных способов. Далее в этой главе везде под словом «граф» мы будем подразумевать ориентированный псевдограф (см. 3.1.5, 3.1.7).

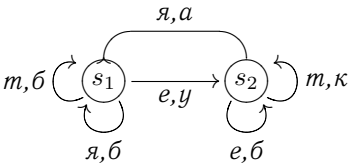


Рис. 5.5

5.5.2. *Пусть задан конечный автомат $\mathcal{A}=\langle A,S,B,\varphi,\psi\rangle$. **Диаграммой автомата** $\mathcal{A}$ называется орграф $G_{\mathcal{A}}$, удовлетворяющий следующим условиям:*

1. *Вершины графа $G_{\mathcal{A}}$ помечены буквами алфавита S.*
2. *Дуги графа помечены парами $(a,b)\in A\times B$.*

3. *Дуга с пометкой* (a, b) *имеет начало в вершине* s_i *и конец в вершине* s_j *тогда и только тогда, когда* $\varphi(s_i, a) = s_j$ *и* $\psi(s_i, a) = b$.

Пример. На рис. 5.5 изображена диаграмма конечного автомата, построенного в предпоследнем примере. □

5.5.3. *Пусть* A, S, B *— конечные алфавиты,* G *— граф, вершины которого взаимно однозначно сопоставлены буквам из* S*, а дуги помечены парами, принадлежащими множеству* $A \times B$*. Граф* G *является диаграммой некоторого детерминированного конечного автомата лишь при выполнении следующих условий:*

1) *Дуги, помеченные парами* (a, b_1) *и* (a, b_2)*, содержащими одну и ту же букву* $a \in A$*, не должны выходить из одной вершины.*

2) *Для каждой буквы* $a \in A$ *и каждой вершины* s *должна существовать дуга с началом в* s*, помеченная парой, левой буквой которой является* a.

Первое из этих условий называется ***условием однозначности****, второе —* ***условием полной определенности***. □

Пример. Диаграмма на рис. 5.6 слева задает конечный автомат. Правая диаграмма на том же рисунке не соответствует никакому ко-

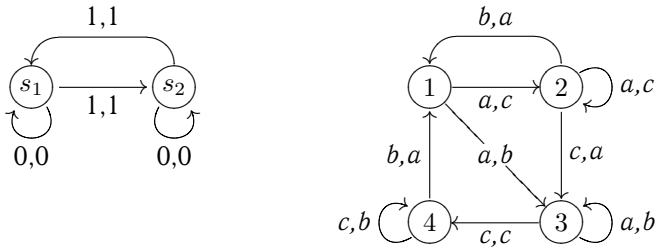


Рис. 5.6

нечному автомату Мили, так как две дуги, исходящие из вершины 1, помечены парами с левой буквой a (не выполняется условие однозначности) и нет дуги, исходящей из той же вершины, помеченной парой, начинающейся на букву c (нарушено условие полной определенности). □

Хотя диаграмма на рис. 5.6 справа не отвечает определению конечного автомата Мили, подавая на вход слова в алфавите $\{a, b, c\}$, в некоторых случаях мы будем получать выходные слова. Таким образом,

нарушение условий однозначности и полной определенности не означает невозможности вычислений, и можно расширить класс изучаемых автоматов, введя следующее определение.

5.5.4. *Детерминированным называется инициальный конечный автомат, диаграмма которого удовлетворяет условиям однозначности, полной определенности, а также следующему* ***условию связности****: для каждой вершины s существует путь из начальной вершины в вершину s.*

Название происходит от латинского слова *determinare* — определять. Условие связности гарантирует, что в автомате нет заведомо лишних состояний.

Пример. Автоматы, изображенные на рис. 5.5 и слева на рис. 5.6, детерминированные. □

5.5.5. *Будем говорить, что путь в диаграмме автомата из начальной вершины s_1 в вершину s_{k+1}, проходящий по дугам, помеченным парами*

$$(a_{i_1}, b_{j_1}),\ \ldots,\ (a_{i_k}, b_{j_k}),$$

несет входное слово $a_{i_1} \ldots a_{i_k}$ *и* ***выходное слово*** $b_{j_1} \ldots b_{j_k}$.

5.5.6. *Для каждого входного слова в графе детерминированного автомата существует один и только один путь с началом в начальной вершине, несущий это слово.*

Доказательство. Утверждение непосредственно следует из 5.5.2–5.5.5. □

5.6. НЕКОТОРЫЕ ВАРИАНТЫ АВТОМАТОВ

Добавляя в определение 5.4.1 различного рода ограничения или же ослабляя уже наложенные ограничения, можно определять те или иные классы автоматов. Примером служит следующее определение.

5.6.1. *Конечный автомат $\langle A, S, B, \varphi, \psi \rangle$, у которого функция выходов ψ не зависит от букв входного алфавита A и отображает множество S в множество B, называется* ***автоматом Мура****.*

Будем считать, что такт работы автомата Мура выглядит следующим образом: находясь в некотором состоянии s_i, автомат воспринимает входной символ a, переходит в состояние $\varphi(s_i, a)$ и на выходе появляется буква $\psi(\varphi(s_i, a))$.

Может показаться, что вычислительные возможности автоматов Мура слабее, чем у автоматов Мили. На самом деле это не так. Чтобы

доказать это, надо уметь сравнивать возможности различных автоматов. Прежде всего изменим определение 5.4.5 таким образом, чтобы оно было согласовано с определением 5.6.1.

5.6.2. ***Поведением автомата Мура*** $\mathcal{A}_{s_1} = \langle A, S, B, \varphi, \psi\rangle$ *называется функция, определенная для каждого состояния* $s \in S$ *и каждого входного слова* α *из* n *букв следующим образом:*

$$\Psi_{\mathcal{A}_{s_1}}(s, \alpha) = \psi(\varphi(s, \pi_1(\alpha)))\psi(\varphi(s, \pi_2(\alpha))) \dots \psi(\varphi(s, \pi_n(\alpha)))\,.$$

Говорят также, что автомат $\mathcal{A}_{s_1}$ ***вычисляет функцию*** $\Psi_{\mathcal{A}_{s_1}}$.

5.6.3. *Инициальные автоматы* $(\mathcal{A}_1, s_{11})$ *и* $(\mathcal{A}_2, s_{21})$ *называются* ***эквивалентными****, если они вычисляют одну и ту же функцию:* $\Psi_{(\mathcal{A}_1, s_{11})} = \Psi_{(\mathcal{A}_2, s_{21})}$.

5.6.4. *Для любого инициального автомата Мили существует эквивалентный ему инициальный автомат Мура.*

Доказательство. Пусть задан инициальный автомат Мили

$$\mathcal{A}_{s_1} = \langle A, S, B, \varphi, \psi\rangle, \quad A = \{a_1, \dots, a_m\},$$
$$B = \{b_1, \dots, b_k\}, \quad S = \{s_1, \dots, s_n\}.$$

Определим автомат Мура $\mathcal{M}_{s_1} = \langle A_1, S_1, B_1, \varphi_1, \psi_1\rangle$ следующим образом. Положим $A_1 = A$ и $B_1 = B$. Каждой паре $s_i a_j$ из множества $S \times A$ сопоставим букву $s_{ij} \notin S$ таким образом, чтобы разным парам соответствовали различные буквы. В алфавит S_1 включим все буквы из алфавита S и буквы s_{ij}, $i = \overline{1,n}$, $j = \overline{1,m}$.

Функцию φ_1 определим следующим образом:

$$\varphi_1(s_i, a_k) = s_{ik}, \quad \varphi_1(s_{ij}, a_k) = \varphi_1(\varphi(s_i, a_j), a_k).$$

Это означает, что если $\varphi(s_i, a_j) = s_q$, то $\varphi_1(s_{ij}, a_k) = s_{qk}$.

Функция ψ_1 на каждом такте должна зависеть только от состояния автомата и не зависеть от воспринимаемого символа, поэтому ее определим так:

$$\psi_1(s_{ij}) = \psi(s_i, a_j), \quad \psi_1(s_i) = b_1.$$

Начальным состоянием автомата будем считать s_1.

Предположим, что на вход автомата Мили $\mathcal{A}_{s_1}$ подана буква a_k. Автомат сработает, перейдет в некоторое состояние s_t и выдаст какую-то букву b_z. Это означает, что $\varphi(s_1, a_k) = s_t$ и $\psi(s_1, a_k) = b_z$. Подадим

ту же букву на вход автомата Мура $\mathcal{M}_{s_1}$. Согласно сказанному выше $\varphi_1(s_1, a_k) = s_{1k}$. Автомат Мура сработал и перешел в состояние s_{1k}. На выходе появляется буква $\psi_1(s_{1k}) = \psi(s_1, a_1) = b_z$. Поведение обоих автоматов на однобуквенных словах одинаково.

Предположим теперь, что при подаче на вход каждого из рассматриваемых автоматов слова, длина которого не превосходит $p > 0$, выходные слова у обоих автоматов совпадают. Подадим на входы слово $a_{k_1}a_{k_2}\ldots a_{k_{p+1}}$. Если автомат Мили подслово $a_{k_1}a_{k_2}\ldots a_{k_p}$ перевел в слово $b_{z_1}b_{z_2}\ldots b_{z_p}$, то, согласно индуктивному предположению, это же слово появится на выходе автомата Мура. При обработке слова $a_{k_1}a_{k_2}\ldots a_{k_p}$ автомат Мили последовательно находился в состояниях

$$\varphi(s_1, a_{k_1}) = s_{u_1},\ \ \varphi(s_{u_1}, a_{k_2}) = s_{u_2},\ \ \ldots$$
$$\ldots,\ \varphi(s_{u_{p-2}}, a_{k_{p-1}}) = s_{u_{p-1}},\ \ \varphi(s_{u_{p-1}}, a_{k_p}) = s_{u_p},$$

а автомат Мура — в состояниях

$$\varphi_1(s_1, a_{k_1}) = s_{1k_1},$$
$$\varphi_1(s_{1k_1}, a_{k_2})\varphi_1(\varphi(s_1, a_{k_1}), a_{k_2})\varphi_1(s_{u_1}, a_{k_2}) = s_{u_1k_2},$$
$$\varphi_1(s_{u_1k_2}, a_{k_3})\varphi_1(\varphi(s_{u_1}, a_{k_2}), a_{k_3})\varphi_1(s_{u_2}, a_{k_3}) = s_{u_2k_3},$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$\varphi_1(s_{u_{p-2}k_{p-1}}, a_{k_p})\varphi_1(\varphi(s_{u_{p-2}}, a_{k_{p-1}}), a_{k_p}) =$$
$$= \varphi_1(s_{u_{p-1}}, a_{k_p}) = s_{u_{p-1}k_p}.$$

Видим, что после обработки указанного слова автомат Мили находится в состоянии s_{u_p}, а автомат Мура — в состоянии $s_{u_{p-1}k_p}$. Оба автомата воспринимают букву $a_{k_{p+1}}$. Пусть в следующем такте на выходе автомата Мили появится буква $b_{z_{p+1}}$. Это означает, что $\psi(s_{u_p}, a_{k_{p+1}}) = b_{z_{p+1}}$. Но тогда

$$\varphi_1(s_{u_{p-1}k_p}, a_{k_{p+1}})\varphi_1(\varphi(s_{u_{p-1}}, a_{k_p}), a_{k_{p+1}})\varphi_1(s_{u_p}, a_{k_p})s_{u_pk_{p+1}},$$
$$\psi_1(s_{u_pk_{p+1}}) = \psi(s_{u_p}, a_{k_{p+1}}) = b_{z_{p+1}}.$$

Мы убедились в том, что выходные слова обоих автоматов совпадают. □

Пример. В разделе 5.5 функция входов и функция выходов автомата Мили $\mathcal{A} = \langle A, S, B, \varphi, \psi\rangle$, перерабатывающего слово «тетя» в слово «бука», были заданы табл. 5.4. Построим эквивалентный автомату $\mathcal{A}$ автомат Мура $\mathcal{A}_1 = \langle A_1, S_1, B_1, \varphi_1, \psi_1\rangle$. Входной и выходной алфавиты этих автоматов совпадают:

$$A_1 = \{\text{т,е,я}\},\quad B_1 = \{\text{б,у,к,а}\}.$$

Таблица 5.4

φ	т	е	я		ψ	т	е	я
s_1	s_1	s_2	s_1		s_1	б	у	б
s_2	s_2	s_2	s_1		s_2	к	б	а

Состояния автомата $\mathcal{A}_1$ обозначим следующим образом:

$$s_1,\ s_2,\ s_{1\text{т}},\ s_{1\text{е}},\ s_{1\text{я}},\ s_{2\text{т}},\ s_{2\text{е}},\ s_{2\text{я}}.$$

В соответствии с приведенными выше таблицами и правилами, указанными в доказательстве теоремы 5.6.4, зададим функцию переходов φ_1. Начальные шаги:

$$\begin{aligned}
&\varphi_1(s_1, \text{т}) = s_{1\text{т}},\ \varphi_1(s_1, \text{е}) = s_{1\text{е}},\ \varphi_1(s_1, \text{я}) = s_{1\text{я}},\\
&\varphi_1(s_2, \text{т}) = s_{2\text{т}},\ \varphi_1(s_2, \text{е}) = s_{2\text{е}},\ \varphi_1(s_2, \text{я}) = s_{2\text{я}},\\
&\varphi_1(s_{1\text{т}}, \text{т}) = \varphi_1(\varphi(s_1, \text{т}), \text{т}) = \varphi_1(s_1, \text{т}) = s_{1\text{т}},\\
&\varphi_1(s_{1\text{т}}, \text{е}) = \varphi_1(\varphi(s_1, \text{т}), \text{е}) = \varphi_1(s_1, \text{е}) = s_{1\text{е}},\\
&\varphi_1(s_{1\text{т}}, \text{я}) = \varphi_1(\varphi(s_1, \text{т}), \text{я}) = \varphi_1(s_1, \text{я}) = s_{1\text{я}}.
\end{aligned}$$

Значения функции φ_1, которые потребуются для превращения слова «тетя» в слово «бука»:

$$\begin{aligned}
&\varphi_1(s_{1\text{т}}, \text{е}) = \varphi_1(\varphi(s_1, \text{т}), \text{е}) = \varphi_1(s_1, \text{е}) = s_{1\text{е}},\\
&\varphi_1(s_{1\text{е}}, \text{т}) = \varphi_1(\varphi(s_1, \text{е}), \text{е}) = \varphi_1(s_2, \text{е}) = s_{2\text{е}},\\
&\varphi_1(s_{2\text{е}}, \text{я}) = \varphi_1(\varphi(s_2, \text{е}), \text{я}) = \varphi_1(s_2, \text{я}) = s_{2\text{я}}.
\end{aligned}$$

Остальные значения функции φ_1 находятся аналогично.

Руководствуясь правилами, указанными в доказательстве теоремы 5.6.4, задаем одноместную функцию выходов ψ_1:

$$\begin{aligned}
&\psi_1(s_1) = \text{б},\ \ \psi_1(s_2) = \text{б},\\
&\psi_1(s_{1\text{т}}) = \psi(s_1, \text{т}) = \text{б},\ \ \psi_1(s_{1\text{е}}) = \psi(s_1, \text{е}) = \text{у},\\
&\psi_1(s_{1\text{я}}) = \psi(s_1, \text{я}) = \text{б},\ \ \psi_1(s_{2\text{т}}) = \psi(s_2, \text{т}) = \text{к},\\
&\psi_1(s_{2\text{е}}) = \psi(s_2, \text{е}) = \text{б},\ \ \psi_1(s_{2\text{я}}) = \psi(s_2, \text{я}) = \text{а}.
\end{aligned}$$

(Для $\psi_1(s_1)$ и $\psi_1(s_2)$ значения назначаются произвольно. Они нужны только для того, чтобы функция ψ_1 была всюду определенной, и на выполнение автоматом $\mathcal{A}_1$ поставленной задачи не влияют.)

Проверим, будет ли построенный автомат решать поставленную задачу. Подадим на вход автомата $\mathcal{A}_1$ слово «тетя». Действия автомата отражены в следующей таблице.

Текущее состояние	Считываемая буква	Новое состояние	Буква на выходе
s_1	т	$s_{1\text{т}}$	б
$s_{1\text{т}}$	е	$s_{1\text{е}}$	у
$s_{1\text{е}}$	т	$s_{2\text{т}}$	к
$s_{2\text{т}}$	я	$s_{2\text{я}}$	а

Автомат работает правильно. □

5.6.5. *Система $\langle A, S, \varphi\rangle$, в которой A — входной алфавит, S — множество состояний и φ — функция переходов, отображающая множество $S \times A$ в S, называется* ***автоматом без выхода****.*

Автомат без выхода реагирует на входные слова не выдачей выходных слов, а своими состояниями. Поскольку на выходе автомата никакие символы не появляются, в его диаграмме ребра помечаются только входными буквами.

Пример. Начав работу, изображенный на рис. 5.7 автомат без выхода попадает в состояние s_1 тогда и только тогда, когда число единиц во входном слове четно. □

5.6.6. *Система $\langle A, S, B, \varphi, \psi\rangle$, в которой A — входной алфавит, S — множество состояний, B — выходной алфавит, φ — отношение на множестве $S \times A \times S$, ψ — отношение на множестве $S \times A \times B$, называется* ***недетерминированным автоматом****.*

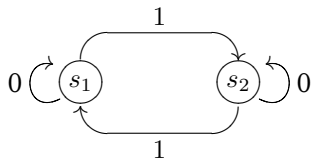


Рис. 5.7

Большинство понятий, введенных ранее для детерминированных автоматов, без труда переносится на случай недетерминированных автоматов. Каждому недетерминированному автомату можно сопоставить его диаграмму способом, аналогичным примененному в 5.5.2 для детерминированных автоматов. Эта диаграмма может не удовлетворять требованиям из 5.5.3. Например, может оказаться, что при подаче на вход автомата, находящегося в состоянии s, некоторой буквы a имеется несколько команд с левой частью sa или же, наоборот, нет команды с такой левой

частью. Логично предполагать, что в последнем случае автомат блокируется. Это, однако, еще не означает, что автомат не способен прочитать все входное слово, содержащее эту букву. О том, как автомат это может делать, будет сказано позднее.

Стоит обратить внимание на то, что отношения φ и ψ можно рассматривать и как функции, определенные на множестве $S \times A$, значениями которых являются соответственно конечные подмножества множеств S и B. Ввиду этого мы будем по-прежнему называть φ **функцией переходов**, а ψ — **функцией выходов**.

Далее мы ограничимся рассмотрением недетерминированных автоматов без выхода.

5.7. АВТОМАТЫ И ЯЗЫКИ

Выделим в множестве всех состояний конечного автомата без выхода некоторое подмножество состояний, называемых **заключительными**. Попадание автомата в заключительное состояние не является сигналом к прекращению работы, он может продолжать перерабатывать входное слово.

5.7.1. *Инициальный автомат без выхода с выделенным множеством заключительных состояний называется* ***настроенным автоматом****.*

5.7.2. *Будем говорить, что* ***входное слово воспринимается настроенным автоматом****, если после прочтения всех букв этого слова автомат перейдет в одно из заключительных состояний.*

5.7.3. *Множество всех входных слов, переводящих начальное состояние настроенного автомата в одно из заключительных состояний, называется* ***языком, представимым*** *этим* ***автоматом****.*

5.7.4. *Поскольку у недетерминированного автомата могут иметься различные команды с совпадающими левыми частями, одному и тому же входному слову в его диаграмме могут отвечать различные пути.* ***Слово воспринимается недетерминированным автоматом****, если хотя бы один из этих путей заканчивается в вершине, помеченной символом заключительного состояния.*

Это определение можно сформулировать и без упоминания о графе автомата. У детерминированного автомата каждому входному слову однозначно соответствует цепочка состояний, в которых он последовательно пребывает при чтении букв этого слова. Прочитав входную букву, недетерминированный автомат может, вообще говоря, перейти

в разные состояния, поэтому у него каждому входному слову соответствует некоторое множество цепочек состояний. Слово воспринимается недетерминированным автоматом, если последнее состояние хотя бы в одной из этих цепочек является заключительным.

Настроенный автомат делит все слова во входном алфавите на два класса: воспринимаемые и не воспринимаемые автоматом. После прочтения воспринимаемого слова автомат находится в заключительном состоянии. Если входное слово не является воспринимаемым, то после его прочтения автомат оказывается в состоянии, которое не является заключительным.

Пример. Язык $\{a^m,\ b^n \mid m, n = 1, 2, \ldots\}$ в алфавите $\{a, b\}$ представим автоматом с диаграммой, изображенной на рис. 5.8. Состояние 1 является начальным, состояния 2 и 3 — заключительными. □

Далее будут рассматриваться только настроенные конечные автоматы.

5.7.5. *Автоматные языки, и только они, представимы недетерминированными конечными автоматами.*

Доказательство. «⇒» Пусть Γ — автоматная грамматика с начальным символом I. Согласно 5.2.9 каждое ее правило имеет либо вид $A \to xB$, либо вид $A \to x$, где A и B — вспомогательные символы, x — один из основных символов. Построим диаграмму недетерминированного конечного автомата $\mathcal{A}$, представляющего язык, порождаемый грамматикой Γ. Для этого каждому вспомогательному символу из Γ сопоставим вершину графа и пометим эту вершину тем же символом. Добавим еще одну вершину и пометим ее некоторым символом, не являющимся основным или вспомогательным в Γ. Для определенности будем считать, что этим символом является Q.

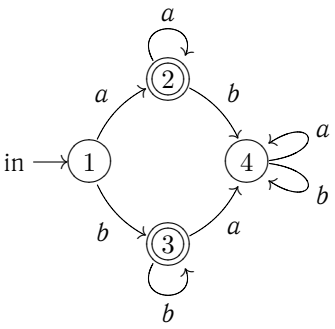


Рис. 5.8

Если правило $A \to xB$ принадлежит грамматике, то из вершины A проведем дугу в вершину B. Дугу пометим символом x. Если правило грамматики имеет вид $C \to y$, проведем дугу из вершины C в вершину Q и пометим ее символом y. Настроим автомат, назвав состояние I начальным, а Q — заключительным.

Докажем, что любое входное слово воспринимается автоматом $\mathcal{A}$ тогда и только тогда, когда это слово принадлежит языку $L(\Gamma)$,

порождаемому грамматикой Г. Пусть $\sigma \in L(\Gamma)$. Это означает, что существует полный вывод, т. е. последовательность слов $\alpha_0, \alpha_1, \ldots, \alpha_n$, в которой $\alpha_0 = I$, $\alpha_n = \sigma$ и, начиная со второго, каждое слово непосредственно выводимо из предыдущего (см. 5.2.3, 5.2.4). Воспользуемся свойствами выводов в автоматных грамматиках, отмеченными в 5.2.10. Слова $\alpha_1, \ldots, \alpha_{n-1}$ получены из предыдущих применением правил вида $A \to xB$, и лишь слово α_n получено из α_{n-1} применением правила вида $A \to x$ (свойство 3). Таким образом, слово α_1 получено из I применением некоторого правила $I \to x_1A_1$, и потому $\alpha_1 = x_1A_1$. Символ x_1 является самым левым в слове α_n (свойство 2). Если слово α_2 получено из α_1 применением правила $A_1 \to x_2A_2$, то $\alpha_2 = x_1x_2A_2$ и символ x_2 является вторым в слове α_n, и т. д. Ввиду этого можно считать, что $\alpha_{n-1} = x_1x_2 \ldots x_{n-1}A_{n-1}$ и что слово $\alpha_n = x_1x_2 \ldots x_n$ получено из α_{n-1} применением правила $A_{n-1} \to x_n$.

Подадим слово $x_1x_2 \ldots x_n$ на вход автомата $\mathcal{A}$. Автомат находится в состоянии I. Правило $I \to x_1A_1$ принадлежит грамматике Г, поэтому диаграмма автомата содержит дугу (I, A_1), помеченную буквой x_1. Автомат перейдет в состояние A_1. Правило $A_1 \to x_2A_2$ также принадлежит грамматике Г, поэтому после прочтения буквы x_2 автомат перейдет в состояние A_2, и т. д. Прочитав букву x_{n-1}, автомат перейдет в состояние A_{n-1}. Далее автомат прочитает букву x_n и в соответствии с правилом $A_{n-1} \to x_n$ перейдет в состояние Q. Так как состояние Q заключительное, слово $x_1x_2 \ldots x_n$ воспринимается автоматом $\mathcal{A}$.

Каждому слову $y_1y_2 \ldots y_m$, воспринимаемому автоматом $\mathcal{A}$, отвечает некоторый маршрут в графе этого автомата, начинающийся в вершине I и заканчивающийся в вершине Q. Пусть

$$Iy_1B_1y_2B_2 \ldots y_{m-1}B_{m-1}y_mQ$$

— пройденный маршрут. Существование этого маршрута означает, что графу автомата $\mathcal{A}$ принадлежат дуги

$$(I, B_1), \quad (B_1, B_2), \quad (B_2, B_3), \ \ldots, \ (B_{m-2}, B_{m-1}), \quad (B_{m-1}, Q),$$

помеченные соответственно буквами $y_1, y_2, \ldots, y_m$. Но в таком случае грамматика Г содержит правила

$$I \to y_1B_1, \quad B_1 \to y_2B_2, \ \ldots, \ B_{m-2} \to y_{m-1}B_{m-1}, \quad B_{m-1} \to y_m.$$

Применяя эти правила в указанном порядке, получим полный вывод слова $y_1y_2 \ldots y_m$.

«$\Leftarrow$» Теперь докажем, что любой язык, представимый недетерминированным конечным автоматом $\mathcal{B}$, может быть порожден подходящей

автоматной грамматикой. Пусть имеется некоторый недетерминированный конечный автомат $\mathcal{B}$, заданный диаграммой. Сопоставим ему грамматику Δ следующим образом. В качестве основного алфавита грамматики возьмем входной алфавит автомата $\mathcal{B}$. Во вспомогательный алфавит включим все буквы, которыми помечены вершины диаграммы автомата $\mathcal{B}$. Напомним, что метками вершин диаграммы автомата являются его состояния. В качестве начального символа выберем букву, которой обозначено начальное состояние автомата $\mathcal{B}$. Дугам графа сопоставим правила грамматики Δ, соблюдая следующие условия:

а) Каждой дуге (s_i, s_j), помеченной символом a_p, сопоставим правило $s_i \to a_p s_j$.

б) Если вершина s_r является заключительной, то каждой дуге (s_u, s_r), помеченной символом a_q, дополнительно сопоставим правило $s_u \to a_q$.

Осталось доказать, что язык, представимый автоматом $\mathcal{B}$, совпадает с языком, порождаемым грамматикой Δ. Требуемые для этого рассуждения аналогичны проведенным выше, и мы их опустим. □

5.7.6. *Назовем недетерминированный автомат $\mathcal{A}$* ***эквивалентным*** *детерминированному автомату $\mathcal{B}$, если языки, представимые этими автоматами, совпадают.*

5.7.7. *Для любого недетерминированного конечного автомата без выхода существует эквивалентный ему детерминированный конечный автомат.*

Доказательство. Рассмотрим недетерминированный конечный автомат $\mathcal{A} = \langle A, S, \varrho \rangle$ с множеством состояний $S = \{s_1, \ldots, s_n\}$, входным алфавитом $A = \{a_1, \ldots, a_m\}$, начальным состоянием s_1 и отношением $\varrho \subseteq A \times S$ (см. 5.6.6), заданный диаграммой G. Построим граф D детерминированного конечного автомата $\mathcal{B}$.

Шаг 1. В D включим $n + 1$ вершин, пометив их буквами $u_0, u_1, \ldots, u_n$. Проведем из каждой вершины m дуг, по одной для каждой буквы входного алфавита, и пометим их этими буквами, руководствуясь следующими правилами:

а) Каждая дуга, выходящая из вершины u_0, должна оканчиваться в этой же вершине, т. е. являться петлей.

б) Если из вершины s_i графа G, где $i \in \{1, \ldots, n\}$, выходит только одна дуга, помеченная буквой a_1, и оканчивается в вершине s_j, то дуга графа D, выходящая из вершины u_i и помеченная той же буквой, должна оканчиваться в u_j. Если из вершины s_1 не выходит ни одной дуги с пометкой a_1, то u_i соединим дугой с u_0, пометив эту дугу

буквой a_1. В случае, когда в графе G из вершины s_i выходит несколько дуг с пометкой a_1, оканчивающихся в вершинах $s_{j_1}, \ldots, s_{j_k}$, добавим к D новую вершину и пометим ее буквой $u_{j_1 \ldots j_k}$. Соединим u_i с новой вершиной дугой, помеченной буквой a_1.

Аналогичные построения проведем для каждой буквы входного алфавита и для каждой вершины u_i, $i = \overline{1, \ldots, n}$. Если окажется, что в какой-то момент к графу D требуется добавить новую вершину с некоторой пометкой, а такая вершина уже есть, то добавлять вершину не надо, дугу следует проводить к имеющейся вершине.

Шаг 2. Проведем дуги из тех вершин графа D, метки которых отличны от $u_0, u_1, \ldots, u_n$. Пусть $u_{j_1 \ldots j_k}$ — одна из них и $a_i \in A$. Предположим, что исходящие из s_{j_1} дуги с пометкой a_i оканчиваются в вершинах $s_{j_{11}}, \ldots, s_{j_{1p_1}}$, ..., дуги с той же пометкой, начинающиеся в s_{j_k}, оканчиваются в вершинах $s_{j_{k1}}, \ldots, s_{j_{kp_k}}$. Объединим индексы меток концов дуг:

$$\{j_{11}, \ldots, j_{1p_1}\} \cup \ldots \cup \{j_{k1}, \ldots, j_{kp_k}\} = \{l_1, \ldots, l_q\}.$$

Добавим к графу D вершину с пометкой $u_{l_1, \ldots, l_q}$, если такой вершины в нем еще нет, и проведем дугу из $u_{j_1 \ldots j_k}$ в эту вершину, пометив ее буквой a_i.

Может случиться, что мы выбрали такую вершину $u_{j_1 \ldots j_k}$, что ни одна из вершин $s_{j_1}, \ldots, s_{j_k}$ не является началом дуги с пометкой a_i. В этом случае из $u_{j_1 \ldots j_k}$ проведем дугу с пометкой a_i в вершину u_0. Аналогичные построения проведем для каждой буквы входного алфавита и для каждой из добавленных на первом шаге вершин.

На шаге 3 проведем такие же построения для вершин, добавленных на втором шаге, и т. д. Построение графа закончится, когда на очередном шаге не будет добавлено ни одной вершины. Такой момент обязательно наступит, так как индексами добавляемых вершин служат подмножества множества $\{1, 2, \ldots, n\}$, а таких подмножеств конечное число.

Настроим автомат $\mathcal{D}$. Если s_1 — начальное состояние автомата $\mathcal{A}$, то u_1 назовем начальным состоянием автомата $\mathcal{D}$. Состояние $u_{i_1, \ldots, i_k}$ назовем заключительным, если хотя бы одно из состояний $s_{i_1}, \ldots, s_{i_k}$ автомата $\mathcal{A}$ является заключительным.

Очевидно, автомат $\mathcal{D}$ детерминированный. Докажем, что он эквивалентен автомату $\mathcal{A}$. Пусть $a_{z_1} \ldots a_{z_q}$ — слово, воспринимаемое автоматом $\mathcal{A}$. Воспринимаемость слова означает существование такой последовательности $s_1, s_{i_1}, \ldots, s_{i_q}$ состояний автомата $\mathcal{A}$, что

$$(s_1, a_{z_1}, s_{i_1}) \in \varrho, \quad (s_{i_1}, a_{z_2}, s_{i_2}) \in \varrho, \ \ldots, \ (s_{i_{q-1}}, a_{z_q}, s_{i_q}) \in \varrho \qquad (5.7.1)$$

и состояние s_{i_q} является заключительным. (Напомним, что ϱ — функция переходов автомата $\mathcal{A}$.)

Подадим на вход автомата $\mathcal{D}$ слово $a_{z_1} \dots a_{z_q}$. Считывая буквы этого слова, автомат последовательно переходит в состояния

$$\varphi(u_1, a_{z_1}) = u_{I_1}, \quad \varphi(u_{I_1}, a_{z_2}) = u_{I_2}, \ \dots, \ \varphi(u_{I_{q-1}}, a_{z_q}) = u_{I_q}.$$

(Здесь через I_j обозначено множество индексов $\{k_{j_1}, \dots, k_{j_t}\}$.) Ввиду определения функции φ для каждого состояния s_{i_j} из последовательности (5.7.1) индекс i_j принадлежит множеству I_j. В частности, $i_q \in I_q$, поэтому состояние u_{I_q} автомата $\mathcal{D}$ является заключительным и слово $a_{z_1} \dots a_{z_q}$ воспринимается этим автоматом.

Теперь докажем, что каждое слово $a_{y_1} \dots a_{y_p}$, воспринимаемое автоматом $\mathcal{D}$, воспринимается также автоматом $\mathcal{A}$. Читая последовательно буквы этого слова, автомат $\mathcal{D}$ будет находиться в состояниях

$$u_1, \quad u_{V_1}, \quad \dots, \quad u_{V_p},$$

причем состояние u_{V_p} является заключительным. Последнее означает, что для некоторого индекса $i_1 \in V_p$ состояние s_{i_1} автомата $\mathcal{A}$ также является заключительным. Из $i_1 \in V_p$ и способа построения графа D следует, что множеству V_{p-1} принадлежит такой индекс i_2, что из вершины s_{i_2} графа G в вершину s_{i_1} проходит дуга, помеченная буквой a_{y_p}.

Множеству V_{p-2} принадлежит такое число i_3, что из вершины s_{i_3} графа G в вершину s_{i_2} проходит дуга, помеченная буквой $a_{y_{p-1}}$, и т. д. Видим, что в графе G имеется маршрут от вершины s_1 до вершины s_{i_1}, ребра которого помечены буквами $a_{y_1}, \dots, a_{y_p}$. Это означает, что слово $a_{y_1} \dots a_{y_p}$ воспринимается автоматом $\mathcal{A}$ (см. 5.7.4). □

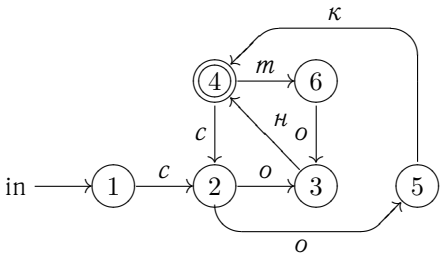


Рис. 5.9
Автомат $\mathcal{A}$

Пример. На рис. 5.9 приведена диаграмма недетерминированного конечного автомата $\mathcal{A}$. Нетрудно убедиться в том, что все слова, принадлежащие языку, представимому этим автоматом, имеют длину $3k$, $k \in \mathbb{N}$, и являются комбинациями подслов «сон», «сок», «тон», причем подслово «тон» не может быть начальным.

Тот же язык представим детерминированным конечным автоматом $\mathcal{D}$, диаграмма которого приведена на рисунке 5.10. Согласно 5.7.6 автоматы $\mathcal{A}$ и $\mathcal{D}$ эквивалентны. □

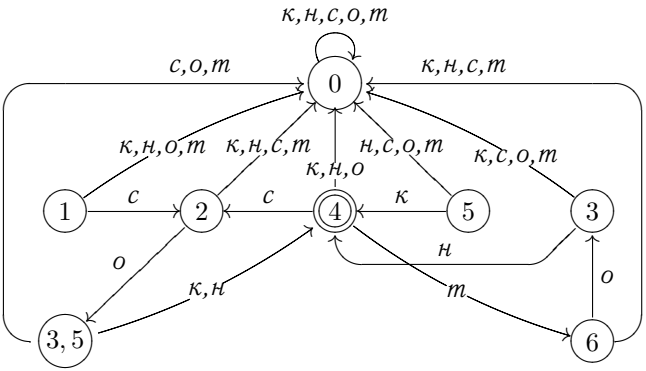


Рис. 5.10
Автомат $\mathcal{D}$

5.7.8. *Автоматные языки, и только они, представимы детерминированными конечными автоматами.*

Доказательство. Утверждение непосредственно следует из 5.7.5–5.7.7. □

Глава 6

ТЕОРИЯ АЛГОРИТМОВ

6.1. О ПОНЯТИИ АЛГОРИТМА

Слово «алгоритм» прочно вошло в русский язык и часто употребляется. Но что же оно означает? Согласно словарю русского языка (М.: Русский язык, 1981) **алгоритмом** называется система вычислений по строго определенным правилам, которая после их выполнения приводит к решению поставленной задачи. По-видимому, более правильно назвать **алгоритмом** совокупность правил, соблюдение которых при вычислениях приводит к решению поставленной задачи. Именно в таком смысле этот термин используется в повседневной речи, и не только применительно к каким-либо вычислениям, но и вообще в тех случаях, когда имеется описание некоторой деятельности в виде конечной совокупности отдельных элементарных действий, выполнение которых в заданном порядке приводит к желаемому результату. Например, спросив у прохожего, как попасть в определенный магазин, вы можете получить примерно такой ответ: «Идите прямо два квартала, поверните налево, на следующем перекрестке поверните направо и через 100 метров справа увидите магазин». Не вызывает сомнений, что совокупность этих указаний является алгоритмом, следуя которому, вы попадете в нужное место.

Мы будем рассматривать только те алгоритмы, которые встречаются в математике. При изучении математики в школе ученики сталкиваются с различными алгоритмами буквально с первых шагов, только чаще эти алгоритмы называются **правилами**. Классическими примерами служат алгоритмы сложения, вычитания, умножения и деления с остатком десятичных чисел. Другим простым примером является правило дифференцирования многочлена от одной переменной. Для того чтобы найти производную от многочлена по этому правилу, нужно только знать все коэффициенты этого многочлена и вовсе не требуется знать, что такое производная. Приведенное в начале параграфа толкование

слова «алгоритм» является довольно расплывчатым. Анализ большого количества известных в математике алгоритмов позволяет выделить некоторые характерные свойства, присущие любому алгоритму.

а) **Дискретность алгоритма**. Преобразование начальных данных происходит по шагам. На каждом шаге из данных, имевшихся на предыдущем шаге, по предписанным правилам получается новая совокупность величин.

б) **Детерминированность**. На каждом шаге результат работы алгоритма однозначно определяется совокупностью данных, полученных на предыдущем шаге.

в) **Направленность алгоритма**. Имеется критерий, позволяющий определить, что является результатом работы алгоритма.

г) **Элементарность шагов алгоритма**. Описание действий, производимых на каждом шаге, должно быть достаточно простым.

д) **Наличие четкого описания**. Правила, по которым надо работать, должны быть описаны настолько подробно, чтобы действия по ним не вызывали затруднений и производились как бы механически.

Опираясь на указанные свойства, обычно можно решить, является ли алгоритмом предложенная процедура. Тем не менее их нельзя принять за определение алгоритма, так как использованные при их описании термины являются весьма расплывчатыми. Например, можно не сойтись во мнении, является ли достаточно простым и четким описание действий алгоритма на каком-то шаге.

Но так ли уж важно иметь точное определение алгоритма? Предположим, что мы хотим выяснить, существует ли алгоритм, позволяющий найти решение каждой из однотипных задач, число которых бесконечно. Указав такой алгоритм, мы дадим на вопрос положительный ответ. Но ведь в некоторых случаях алгоритма может не существовать. Доказать, что алгоритма нет, можно только тогда, когда имеется строгое определение этого понятия.

Само слово алгоритм происходит от имени арабского математика аль-Хорезми (т. е. Хорезмского, из Хорезма), написавшего в IX в. книгу по арифметике, в которой рассказывалось о десятичной системе счисления и излагались правила действий с десятичными числами. В средневековой Европе эта книга пользовалась популярностью, но имя автора при переводе приобрело новое звучание.

Как правило, алгоритм создается для обработки определенных исходных данных, и его использование в других случаях бессмысленно. Однако и при работе с подходящими начальными данными возможны

ситуации, когда применение алгоритма не дает результата, т. е. алгоритм к этим данным не применим.

6.1.1. *Множество объектов, к которым применим алгоритм, называется* ***областью его применимости.***

Далее мы будем рассматривать только такие функции, у которых аргументы изменяются на множестве целых неотрицательных чисел $\mathbb{N} \cup \{0\} = \{0, 1, 2, \ldots\}$ и значения принадлежат тому же множеству. При некоторых значениях аргументов значения этих функций могут быть не определены. Такие функции называются **частичными**.

Все указанные выше функции делятся на два класса. Для функций из одного класса существует алгоритм, позволяющий для каждого набора значений аргументов вычислить значение этой функции. Для функций из другого класса подобного алгоритма не существует.

6.1.2. *Функция, область определения которой совпадает с областью применимости некоторого алгоритма, а значение всегда совпадает с результатом применения этого алгоритма, называется* ***вычислимой.***

Множества также делятся на два типа: перечислимые и неперечислимые.

6.1.3. ***Алгоритм перечисляет множество*** A*, если областью его применимости является множество целых неотрицательных чисел, и* A *совпадает с множеством всех возможных результатов использования алгоритма.*

6.1.4. ***Множество называется перечислимым****, если либо оно пусто, либо существует алгоритм, перечисляющий это множество.*

Иногда возникает следующая задача: можно ли для каждого элемента из множества Q определить, принадлежит ли он также другому заданному множеству P. В одних случаях существует алгоритм, решающий эту проблему, а в других случаях такого алгоритма нет.

6.1.5. *Множество* P *называется* ***разрешимым относительно множества*** Q*, если существует алгоритм, применимый к каждому элементу* x *из* Q*, и результатом применения является 0, если* $x \notin Q \cap P$*, и 1, если* $x \in Q \cap P$*.*

6.2. МАШИНЫ ТЬЮРИНГА

Существует несколько равносильных формализаций понятия алгоритма. В этом параграфе мы рассмотрим один из вариантов. В его основе лежат следующие соображения. Каждый шаг алгоритма должен

быть элементарным, для его выполнения не требуется вникать в существо задачи. Такие действия может выполнять некоторая специальным образом сконструированная машина, а сам алгоритм превращается в программу, по которой работает эта машина. Впервые подобные машины были описаны независимо А. М. Тьюрингом и Э. Постом в первой трети прошлого века, и в дальнейшем было придумано много вариантов таких машин. В настоящее время их принято называть **машинами Тьюринга**. Опишем одну из версий, близкую к машине Поста.

Для ввода и вывода данных, а также для промежуточных записей машина Тьюринга имеет **ленту**, разделенную на **ячейки** (клетки). В каждый момент времени длина ленты конечна, но при необходимости к ней можно подклеивать новые куски, так что потенциально ее длина не ограничена. В каждую ячейку может быть записана буква некоторого алфавита $A = \{a_0, a_1, \ldots, a_m\}$. Удобно считать, что пустые ячейки на самом деле уже содержат какую-нибудь букву алфавита A. Для определенности будем полагать, что этой буквой является a_0. На ленте фиксированы направления вправо и влево, так что у каждой ячейки, за исключением крайних, имеется соседняя справа и соседняя слева. Некоторое устройство, называемое **головкой** или **кареткой**, может на каждом шаге воспринимать (считывать) букву в одной из ячеек, заменять эту букву любой буквой алфавита A, оставаться на месте или передвигаться на каждом шаге (такте) на одну ячейку вправо или влево. Головка находится в одном из **состояний**, которые будут обозначаться буквами алфавита $Q = \{q_0, q_1, \ldots, q_n\}$. Зная состояние головки и считываемый символ, можно предсказать, что произойдет с головкой на следующем шаге. Состояние q_0 назовем **заключительным**, а состояние q_1 — **начальным**. Состояния головки будем считать также и состояниями машины.

Опишем один шаг в работе машины. Предполагаем, что в рассматриваемый момент времени она находится в состоянии, не являющемся заключительным, и ее головка воспринимает букву, записанную в одной из ячеек. В следующий момент машина записывает в ту же ячейку одну из букв алфавита A, переходит в новое состояние или остается в старом, головка сдвигается на одну ячейку вправо или влево или остается на месте. На этом шаг в работе машины заканчивается. Конкретные действия машины на каждом шаге задаются **командами**. Каждая команда имеет вид

$$q_i a_j \to a_k S q_l\,, \tag{6.2.1}$$

где a_j и a_k — буквы из алфавита A, q_i и q_l — буквы из Q, $\to$ — особый символ, не входящий ни в A, ни в Q, S может принимать значения L,

R, N. Слова, записанные в команде слева и справа от стрелки, будем называть **левой частью** и **правой частью** команды. Конечное множество команд образует **программу**. Программа не должна содержать различных команд с совпадающими левыми частями. Также программа не может содержать команды, в левой части которой имеется символ q_0.

Предположим, что в некоторый момент машина находится в состоянии $q_i \neq q_0$ и воспринимает букву a_j. В этом случае машина должна выполнить команду (6.2.1). В соответствии с этой командой головка машины заменит в обозреваемой ячейке букву a_j на a_k, сдвинется на одну ячейку влево, если $S = L$; на одну ячейку вправо, если $S = R$, останется на месте при $S = N$ и перейдет в состояние q_l. Один шаг в работе машины проделан.

Если для каждой пары букв из множества $Q \times A$ программа содержит команду, левая часть которой совпадает с этой парой, то на каждом шаге действия машины определяются этой программой. При нарушении этого условия может оказаться, что в состоянии q_s машина воспринимает букву a_t, а программа не содержит команды с левой частью $q_s a_t$. Будем считать, что в этом случае машина остается в состоянии q_s и более никаких действий не производит.

Работа машины считается законченной только в том случае, если машина находится в состоянии q_0. Символ q_0 не может входить в левую часть какой-либо команды, поэтому согласно сказанному выше машина дальнейших действий не производит и остается в этом состоянии.

Поскольку поведение машины Тьюринга полностью определено ее программой, мы будем отождествлять машину с этой программой. Будем говорить, что рассматриваемая машина является **машиной Тьюринга в алфавите** A, чтобы подчеркнуть, что для записей на ленте используются буквы этого алфавита.

Записав на ленте какие-нибудь слова в алфавите A, приведя машину в некоторое состояние, отличное от q_0, и поместив ее головку над одной из ячеек ленты, можно машину запустить. Она начнет работать в соответствии с программой и либо на некотором шаге попадет в заключительное состояние q_0 и закончит работу, либо никогда не попадет в состояние q_0, и работа машины будет продолжаться «вечно». В последнем случае считается, что к исходным данным машина не применима.

Нетрудно заметить сходство в определениях и функционировании машины Тьюринга и конечного автомата. Фундаментальное различие состоит в том, что в процессе работы машина Тьюринга может менять

записи на ленте, т. е. обладает внешней памятью. Эта память бесконечна, так как количество записей ничем не ограничивается.

6.3. ВЫЧИСЛЕНИЯ НА МАШИНАХ ТЬЮРИНГА

6.3.1. *Конфигурацией машины Тьюринга в данный момент времени называется полная информация о внутреннем состоянии машины, заполнении ячеек ленты буквами и о ячейке, которую обозревает головка машины. Конфигурация может быть записана в виде слова xq_iy, удовлетворяющего следующим условиям:*

а) *xy — слово, полученное соединением слов x и y и записанное на ленте;*

б) *левее слова x и правее слова y все ячейки ленты содержат букву a_0;*

в) *головка машины обозревает ячейку, в которой записана первая буква слова y;*

г) *q_i — состояние машины в рассматриваемый момент времени.*

6.3.2. *Конфигурация машины называется начальной, если она содержит символ q_1, заключительной, если она содержит символ q_0.*

Следуя определению 6.3.1, конфигурацию машины в определенный момент времени можно зафиксировать различными словами, так как слова x и y можно изменять, добавляя к ним буквы a_0 соответственно слева и справа. Обычно это не вызывает недоразумений, поскольку в любом случае мы будем иметь полную информацию о содержимом ячеек, состоянии машины и о том, какая ячейка обозревается. Неоднозначности записи легко избежать, если предполагать, что слово xy содержит все буквы, записанные на ленте (по предположению в каждый момент времени длина ленты конечна).

6.3.3. *Чтобы показать, что за один такт в работе машины конфигурация $x_1q_iy_1$ переходит в конфигурацию $x_2q_jy_2$, будем использовать запись*

$$x_1q_iy_1 \models x_2q_jy_2\,.$$

Каждая из формул

$$x_1q_iy_1 \vdash x_2q_jy_2, \quad x_1q_iy_1 \Rightarrow x_2q_jy_2$$

означает, что из конфигурации $x_1q_iy_1$ машина переходит в конфигурацию $x_2q_jy_2$ через один или более тактов.

Из приведенного выше описания видно, что действия машины Тьюринга весьма разнообразны и зависят не только от программы, но и от начальной записи на ленте и начального положения головки. Любые действия машины можно называть вычислениями. Особый интерес представляют вычисления, описанные в следующем определении.

6.3.4. *Пусть $F(x_1, \ldots, x_n)$ — функция, определенная на множестве всех слов в алфавите A. Машина Тьюринга* ***правильно вычисляет функцию*** *F, если для любых слов $w_1, \ldots, w_n$ в том же алфавите, начав работу в конфигурации*

$$q_1 a_0 w_1 a_0 w_2 a_0 \ldots a_0 w_n \,,$$

машина закончит ее в конфигурации $q_0 a_0 F(w_1, \ldots, w_n)$:

$$q_1 a_0 w_1 a_0 w_2 a_0 \ldots a_0 w_n \Rightarrow q_0 a_0 F(w_1, \ldots, w_n) \,.$$

В состоянии q_0 головка должна остановиться над той же ячейкой, над которой она была в начальной конфигурации.

Примеры 1. Машина Тьюринга в алфавите {0,1}, заданная приведенной ниже программой, правильно вычисляет функцию $x + y$, определенную на множестве целых неотрицательных чисел. Величины x и y записываются на ленте как x единиц и y единиц, идущих подряд, а между ними стоит разделяющий нуль. Нуль считаем тем самым символом a_0, который заранее вписан во все ячейки ленты. Начальная конфигурация при $x = 2$, $y = 3$ выглядит так: $q_1 0110111$. При $x = 0$, $y = 3$ начальная конфигурация принимает вид $q_1 00111$. Это означает, что число нуль кодируется пустым словом. Программа работы машины:

$$\begin{array}{llll} q_1 0 \to 0 R q_2, & q_2 1 \to 1 R q_2, & q_2 0 \to 1 R q_3, & q_3 1 \to 1 R q_3, \\ q_3 0 \to 0 L q_4, & q_4 1 \to 0 L q_5, & q_5 1 \to 1 L q_5, & q_5 0 \to 0 N q_0. \end{array}$$

Чтобы лучше понять, как проходят вычисления, выпишем последовательно конфигурации машины от начала работы до завершения при $x = 2$, $y = 3$:

$$\begin{array}{l} q_1 01101110 \models 0 q_2 1101110 \models 01 q_2 101110 \models 011 q_2 01110 \models \\ 0111 q_3 1110 \models 01111 q_3 110 \models 011111 q_3 10 \models 0111111 q_3 0 \models \\ 011111 q_4 10 \models 01111 q_5 100 \models 0111 q_5 110 \models 011 q_5 1110 \models \\ 01 q_5 11110 \models 0 q_5 111110 \models q_5 111110 \models q_0 0111110. \end{array}$$

Видим, что вычисление начинается со сдвига головки вправо и перехода в состояние q_2 — состояние движения по единицам вправо. Как только головка обнаруживает нуль, она заменяет его единицей и сдвигается вправо, переходя в состояние q_3. В этом состоянии головка движется вправо, проходит через все единицы, образующие слагаемое y, на нуле переходит в состояние q_4 и сдвигается влево. В состоянии q_4 машина заменяет правую единицу нулем, сдвигает головку влево и переходит в состояние q_5 — состояние движения через единицы влево. Как только головка обнаруживает нуль, она останавливается, и машина переходит в состояние q_0, т. е. выключается.

Следует обратить особое внимание на то, что при попадании машины в состояние q_0 головка находится над той же ячейкой, над которой она находилась в начале работы, а также на то, что вычисления производятся правильно и в случаях, когда один или оба аргумента функции равны нулю.

2. Требуется построить машину Тьюринга в алфавите

$$\{a_0, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\},$$

правильно вычисляющую функцию, определенную на множестве целых неотрицательных чисел, сопоставляющую каждой десятичной записи $x_1x_2\ldots x_k$ числа x десятичную запись числа $x+2$.

Очевидно, $q_1a_0x_1x_2\ldots x_k$ — начальная конфигурация машины. Сначала перебросим головку вправо командами

$$q_1a_0 \to a_0Rq_2, \quad q_2i \to iRq_2, \quad i = \overline{0,9}.$$

Как только в состоянии q_2 головка окажется над ячейкой с символом a_0, движение вправо должно закончиться, поэтому записываем команду

$$q_2a_0 \to a_0Lq_3\,.$$

Теперь головка находится над последней цифрой числа x и можно начинать сложение. Воспользуемся известным из курса арифметики правилом сложения целых неотрицательных чисел. Если x оканчивается на одну из цифр $0, 1, \ldots, 7$, то команды выглядят следующим образом:

$$q_3i \to i+2Lq_4, \quad i = \overline{0,7}.$$

После выполнения одной из этих команд головка должна пройти влево через все остальные цифры в записи числа x и остановиться левее первой из них:

$$q_4i \to iLq_4, \quad i = \overline{0,9}, \quad q_4a_0 \to a_0Nq_0.$$

В случаях, когда последняя цифра x_k числа x равна 8 или 9, действия машины должны быть другими. Так как $8 + 2 = 10$ и $9 + 2 = 11$, записываем команды

$$q_3 8 \to 0Lq_5, \quad q_3 9 \to 1Lq_5.$$

Согласно правилам сложения чисел в обоих случаях следует к x_{k-1} прибавить 1. Команды похожи на уже записанные:

$$q_5 i \to i + 1Lq_4, \quad i = \overline{0,8}, \quad q_5 9 \to 0Lq_5.$$

Если $x_{k-1} = 9$, то единица прибавляется к x_{k-2}. Это позволило организовать в программе цикл. Чтобы выйти из него, даем команду

$$q_5 a_0 \to 1Lq_0\,.$$

Построение машины закончено.

Для проверки выпишем все конфигурации машины, получаемые при вычислениях для $x = 298$:

$$q_1 a_0 298 a_0 \models a_0 q_2 298 a_0 \models a_0 2 q_2 98 a_0 \models a_0 29 q_2 8 a_0 \models a_0 298 q_2 a_0 \models$$
$$a_0 29 q_3 8 a_0 \models a_0 2 q_5 90 a_0 \models a_0 q_5 200 a_0 \models q_4 a_0 300 a_0 \models q_0 a_0 300 a_0. \qquad \square$$

Как и в случае конечных автоматов, машину Тьюринга можно задавать графом. Так, диаграмма машины Тьюринга, построенной в примере 1, приведена на рис. 6.1. Такой способ задания нагляден и особенно удобен при большом числе состояний.

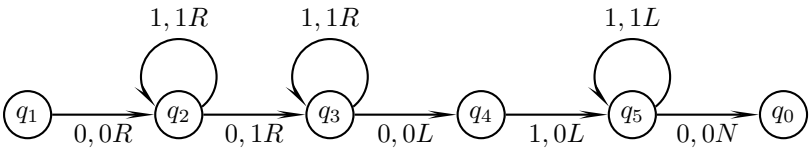


Рис. 6.1
Диаграмма машины Тьюринга

6.4. ТЕЗИС ТЬЮРИНГА

Как правило, используя некоторый алгоритм для решения какой-либо математической задачи, мы записываем начальные данные в виде слов в некотором алфавите, затем преобразуем эти слова в другие сло-

ва согласно предписаниям алгоритма и, наконец, получаем результат также в виде слова. Вследствие этого указанный алгоритм представляет собой совокупность правил, по которым последовательно преобразуются слова в некотором алфавите A. Подобные алгоритмы называются **алгоритмами в алфавите** A.

6.4.1. *Для любого алгоритма $\mathcal{A}$ в некотором алфавите существует такая машина Тьюринга $\mathcal{T}$ в том же алфавите, что всегда при одинаковых начальных данных результаты работы алгоритма $\mathcal{A}$ и машины $\mathcal{T}$ совпадают.*

Это утверждение, известное под названием **тезиса Тьюринга**, не может быть доказано, поскольку само понятие алгоритма является интуитивным, т. е. основанным на опыте. Однако оно выглядит очень убедительно, так как при своей работе машина Тьюринга в определенном смысле копирует действия человека, производящего вычисления по заданным правилам. Деятельность такого человека-вычислителя можно описать следующим образом. Человек

а) читает данные, записанные на бумаге;

б) ищет подходящее правило;

в) в соответствии с таким правилом что-то изменяет в исходных данных и полученное записывает на бумаге;

г) снова и снова производит действия а), б), в) до тех пор, пока не станет применимым правило, утверждающее, что получен нужный результат.

Прочитав данные, вычислитель запоминает их, т. е. приходит в определенное «состояние ума». В этом состоянии он ищет правило и запоминает его, т. е. переходит в новое «состояние ума». Далее он производит записи измененных данных и переходит в новое «состояние», чтобы прочитать и запомнить написанное.

Можно возразить, что машина Тьюринга на каждом шаге читает и изменяет одну букву, в то время как человек-вычислитель считывает, запоминает и записывает сразу большие объемы информации. Однако это не принципиально, поскольку процесс чтения и записи является побуквенным. Таковым можно упрощенно считать и процесс запоминания.

Справедливость тезиса Тьюринга подтверждается также тем, что

а) не найдено никаких алгоритмов, не реализуемых в виде подходящей машины Тьюринга;

б) известно несколько определений, уточняющих понятие алгоритма; в каждом случае было доказано, что при помощи алгоритмов,

основанных на этих определениях, вычислимыми являются те и только те функции, которые вычислимы на подходящей машине Тьюринга.

Таким образом, тезис Тьюринга является естественнонаучным фактом, который не доказывается, а подтверждается всем нашим опытом.

6.5. СУЩЕСТВОВАНИЕ НЕРАЗРЕШИМЫХ ПРОБЛЕМ

6.5.1. *Каждой машине Тьюринга можно сопоставить некоторое натуральное число таким образом, что разным машинам Тьюринга будут соответствовать разные числа.*

Доказательство. Чтобы задать машину Тьюринга, мы указываем внешний алфавит $\{a_0, a_1, \ldots, a_m\}$, внутренний алфавит $\{q_0, q_1, \ldots, q_n\}$ и программу, состоящую из команд вида

$$q_i a_j \to a_k S q_l, \quad S \in \{L, N, R\}.$$

Эту программу можно записать в виде одного слова, отбросив стрелки:

$$q_{i_1} a_{j_1} a_{k_1} S_1 q_{l_1} q_{i_2} a_{j_2} a_{k_2} S_2 q_{l_2} \ldots q_{i_p} a_{j_p} a_{k_p} S_p q_{l_p}. \tag{6.5.1}$$

Пример. Рассмотрим машину Тьюринга с внешним алфавитом $\{0, 1\}$, внутренним алфавитом $\{q_0, q_1, q_2, q_3\}$ и программой

$$\begin{aligned} &q_1 0 \to 0 R q_2, \quad && q_2 0 \to 0 L q_0, \quad && q_2 1 \to 1 R q_3, \\ &q_3 1 \to 0 R q_3, \quad && q_3 0 \to 0 L q_4, \quad && q_4 0 \to 0 L q_4, \quad q_4 1 \to 1 L q_0. \end{aligned} \tag{6.5.2}$$

Эта машина правильно вычисляет функцию

$$\operatorname{sg}(x) = \begin{cases} 0, & \text{если } x = 0, \\ 1, & \text{если } x > 0. \end{cases}$$

Ее программу можно записать в виде слова

$$q_1 00 R q_2 q_2 00 L q_0 q_2 11 R q_3 q_3 10 R q_3 q_3 00 L q_4 q_4 00 L q_4 q_4 11 L q_0\,. \qquad \square$$

Поскольку число символов во внешнем и внутреннем алфавитах произвольной машины Тьюринга может быть сколь угодно велико, зададим два счетных алфавита, не имеющих общих букв:

$$A_1 = \{a_0, a_1, a_2, \ldots\}, \quad A_2 = \{q_0, q_1, q_2, \ldots\}.$$

Можно считать, что алфавиты каждой конкретной машины Тьюринга получаются из A_1 и A_2 отбрасыванием лишних букв. Теперь зададим

функцию $n(x)$, сопоставляющую каждой букве x из алфавитов A_1 и A_2 число, полагая

$$n(a_i) = 10^{2i+4}, \quad n(q_i) = 10^{2i+5}, \quad i = 0, 1, 2, \ldots.$$

Будем также предполагать, что

$$n(L) = 10, \quad n(N) = 100, \quad n(R) = 1000.$$

Очевидно, значения $n(x_1)$ и $n(x_2)$ могут совпадать только тогда, когда $x_1 = x_2$.

Сопоставим теперь программе (6.5.1) слово

$$n(q_{i_1})n(a_{j_1})n(a_{k_1})n(S_1)n(q_{l_1})\ldots n(q_{i_p})n(a_{j_p})n(a_{k_p})n(S_p)n(q_{l_p})$$

и назовем его **шифром** машины Тьюринга. В этом слове значения функции n не перемножаются, а просто последовательно выписаны. Легко заметить, что каждой машине Тьюринга могут соответствовать разные шифры, так как последовательность команд в (6.5.1) можно менять. Но у разных машин Тьюринга шифры совпадать не могут, так как по каждому шифру программу машины можно однозначно восстановить.

Шифр машины состоит из нулей и единиц и всегда начинается с единицы. Его можно считать двоичной записью некоторого числа из множества $\mathbb{N}$. □

Пример. Машина Тьюринга, заданная программой (6.5.2) и вычисляющая функцию $\mathrm{sg}(x)$, имеет следующий шифр (из-за ограниченности места слово разбито на части):

$$\begin{aligned}
&n(q_1)n(0)n(0)n(R)n(q_2)n(q_2)n(0)n(0)n(L)n(q_0)\\
&n(q_2)n(1)n(1)n(R)n(q_3)n(q_3)n(1)n(0)n(R)n(q_3)\\
&n(q_3)n(0)n(0)n(L)n(q_4)n(q_4)n(0)n(0)n(L)n(q_4)\\
&n(q_4)n(1)n(1)n(L)n(q_0).
\end{aligned}$$

Так как

$$\begin{aligned}
&n(L) = 10, \quad n(N) = 10^2, \quad n(R) = 10^3,\\
&n(0) = n(a_0) = 10^4, \quad n(1) = n(a_1) = 10^6,\\
&n(q_0) = 10^5, \quad n(q_1) = 10^7, \quad n(q_2) = 10^9,\\
&n(q_3) = 10^{11}, \quad n(q_4) = 10^{13},
\end{aligned}$$

в числах этот шифр выглядит так:

$$\begin{aligned}
&10^7 10^4 10^4 10^3 10^9\ 10^9 10^4 10^4 10^1 10^5\ 10^9 10^6 10^6 10^3 10^{11}\ 10^{11} 10^6 10^4\\
&10^3 10^{11} 10^{11} 10^4 10^4 10^1 10^{13}\ 10^{13} 10^4 10^4 10^1 10^{13}\ 10^{13} 10^6 10^6 10^1 10^5.
\end{aligned}$$

□

6.5.2. *Предположим, что внешний алфавит некоторой машины Тьюринга T содержит символы $a_0, 0, 1$, на ленте записан шифр этой машины, головка находится над ячейкой левее первой единицы шифра, а машина находится в состоянии q_1. Машина T называется* ***самоприменимой****, если после начала работы в указанной конфигурации она через конечное число тактов попадет в состояние q_0, в противном случае машина T называется* ***несамоприменимой****.*

Пример. Машина

$$q_1 1 \to 1 N q_0, \quad q_1 0 \to 0 N q_0$$

является самоприменимой, так как после одного такта в работе она попадает в состояние q_0 независимо от того, что было записано на ленте.

Программа машины

$$q_1 1 \to 1 N q_1, \quad q_1 0 \to 0 N q_1$$

не содержит состояния q_0, поэтому она в это состояние попасть не может и, следовательно, является несамоприменимой. □

Рассмотрим теперь **проблему самоприменимости**: существует ли алгоритм, позволяющий по каждому шифру определить, является ли машина с этим шифром самоприменимой? Поскольку каждый алгоритм мы теперь отождествляем с некоторой машиной Тьюринга, указанную задачу можно сформулировать более конкретно.

6.5.3 (проблема самоприменимости)**.** *Пусть $p(x)$ — функция, равная единице, если x является шифром самоприменимой машины, и равная нулю, если x является шифром несамоприменимой машины. Существует ли машина Тьюринга T в алфавите $\{a_0, 0, 1\}$, правильно вычисляющая функцию $p(x)$?*

6.5.4. *Говорят, что проблема самоприменимости* ***алгоритмически разрешима****, если указанная в* 6.5.3 *машина Тьюринга существует. В противном случае говорят, что эта проблема* ***алгоритмически не разрешима****.*

Замечание. Машина T должна сигнализировать нам о том, является ли слово на ленте шифром самоприменимой или несамоприменимой машины. Она может это сделать различными способами. Самым простым является выдача в качестве результата некоторой буквы в одном случае и иной буквы в другом случае. Мы взяли в качестве таких букв символы $1, 0$.

6.5.5. *Проблема самоприменимости алгоритмически не разрешима.*

Доказательство. Предположим, что указанная в 6.5.3 машина Тьюринга T, правильно вычисляющая функцию $p(x)$, существует. Если в ее начальной конфигурации на ленте был записан шифр какой-нибудь самоприменимой машины, то через некоторое время машина T попадет в состояние q_0, а на ленте останется только одна единица. Если же на ленте был записан шифр какой-нибудь несамоприменимой машины, то машина T также попадет в состояние q_0, но на ленте останутся только нули.

Пусть $\{q_0, q_1, \ldots, q_m\}$ — внутренний алфавит машины T, т. е. список всех состояний, в которых эта машина может находиться. Изменим программу машины T следующим образом. В командах, содержащих символ q_0, заменим q_0 на q_{m+1}. Добавим к программе команды

$$q_{m+1}0 \to 0Nq_{m+1}, \quad q_{m+1}1 \to 1Nq_0. \tag{6.5.3}$$

В результате получим новую машину Тьюринга, которую обозначим через T_1.

Предположим, что в начальный момент на лентах машин T и T_1 был записан шифр какой-то самоприменимой машины Тьюринга и обе машины начинают работу. Действия машин будут идентичны до тех пор, пока машина T не попадет в состояние q_0. На этом такте машина T_1 выполнит другую команду и перейдет в состояние q_{m+1}.

Так как машина T правильно вычисляет введенную в 6.5.3 функцию $p(x)$, после ее остановки на ленте будет находиться одна единица и головка в состоянии q_0 будет находиться левее этой единицы, над ячейкой, в которой записана буква 0. В тот же тактовый момент конфигурация машины T_1 будет отличаться от конфигурации машины T только тем, что она попадет в состояние q_{m+1}. Машина T_1 продолжит работу, выполняя команды (6.5.3). В состояние q_0 машина T_1 попасть не сможет.

Если в начальный момент на лентах машин T и T_1 был записан шифр некоторой несамоприменимой машины Тьюринга, то каждая из них попадет в заключительное состояние, выдав в качестве результата число 0.

Очевидно, что каждая машина Тьюринга в алфавите $\{a_0, 0, 1\}$ является либо самоприменимой, либо несамоприменимой. Попробуем выяснить, к какому из этих двух классов принадлежит машина T_1. Для этого на ленте запишем шифр этой машины и запустим ее.

Если машина T_1 самоприменимая, то ее шифр является шифром самоприменимой машины и машина будет действовать так, как описано

выше. Она никогда не попадет в заключительное состояние, следовательно, самоприменимой быть не может.

Предположим, что машина T_1 несамоприменимая и ее шифр является шифром несамоприменимой машины. Как уже говорилось ранее, через конечное число тактов она попадет в состояние q_0. Согласно 6.5.2 это означает, что машина T_1 самоприменимая, что противоречит нашему предположению.

Мы убедились в том, что машина T_1 не является ни самоприменимой, ни несамоприменимой. Поскольку это невозможно, неверным является наше предположение о том, что существует машина Тьюринга T, распознающая шифры самоприменимых и несамоприменимых машин. □

6.6. РЕКУРСИВНЫЕ ФУНКЦИИ

Другой подход к формализации понятия вычислимой функции основывается на следующих соображениях. Определяется некоторый класс K настолько простых функций, что в вычислимости их невозможно сомневаться. Указывается совокупность P таких правил получения из вычислимых функций новых функций, что при использовании любого из них получается также вычислимая функция. Если удачно выбрать K и P, то вычислимыми окажутся те и только те функции, которые могут быть получены из функций, принадлежащих классу K, применением конечного числа правил из множества P.

6.6.1. *Все рассматриваемые в этом разделе функции определены на множестве $\mathbb{N}_0$ целых неотрицательных чисел, и их значения также принадлежат этому множеству.*

6.6.2. ***Простейшими*** *называются следующие функции:*

$$s(x) = x + 1, \quad o^n(x_1, \ldots, x_n) = 0, \quad e^n_m(x_1, \ldots, x_n) = x_m.$$

Вычислимость значений любой простейшей функции очевидна. Функции e^n_m обычно называются **проектирующими** или **проекциями**.

Перейдем к описанию правил получения из заданных вычислимых функций новых вычислимых функций.

6.6.3. *Пусть заданы k-местные функции*

$$g_1(x_1, \ldots, x_k), \ \ldots, \ g_n(x_1, \ldots, x_k),$$

среди которых могут быть совпадающие, и n-местная функция $f(x_1, \ldots, x_n)$. Говорят, что функция

$$h(x_1, \ldots, x_k) = f(g_1(x_1, \ldots, x_k), \ldots, g_n(x_1, \ldots, x_k))$$

получена из функций $f, g_1, \ldots, g_n$ ***операцией суперпозиции (подстановки)***.

Пример 1. Любая функция $f(x_1, \ldots, x_n)$, принимающая только одно значение a, может быть получена суперпозицией из простейших функций. Действительно,

$$f(x_1, \ldots, x_n) = \underbrace{s(s(\ldots s}_{a \text{ раз}}(o^n(x_1, \ldots, x_n))\ldots))\,. \qquad \square$$

Пример 2. Функция $(x+1)^2(x+2)$ получена суперпозицией функций $xyz,\ x+1,\ x+2$. $\square$

6.6.4. *Пусть при* $n > 0$ *заданы* n*-местная функция* g *и* $(n+2)$*-местная функция* h. *Функция* f, *зависящая от* $n+1$ *переменных, получена из* g *и* h ***примитивной рекурсией***, *если для любых* $x_1, x_2, \ldots, x_n, y$

$$f(x_1, \ldots, x_n, 0) = g(x_1, \ldots, x_n)\,,$$

$$f(x_1, \ldots, x_n, y+1) = h(x_1, \ldots, x_n, y, f(x_1, \ldots, x_n, y))\,.$$

Одноместная функция p *получена примитивной рекурсией из константы* a *и функции* q, *зависящей от двух аргументов, если*

$$p(0) = a, \quad p(x+1) = q(x, p(x))\,.$$

Латинское слово *recursio* означает *возвращение*. Для нахождения неизвестного значения функции f при определенном ненулевом значении последнего аргумента используются значения этой же функции при меньших значениях того же аргумента, т. е. происходит «возвращение» к найденным ранее величинам.

Пример. Пусть

$$f(x_1, x_2) = x_1 \cdot x_2, \quad g(x_1) = o(x_1), \quad h(x_1, x_2, x_3) = x_1 + x_3,$$

тогда

$$\begin{aligned}
&f(x_1, 0) = 0 = o(x_1),\\
&f(x_1, 1) = x_1 = x_1 + 0 = h(x_1, 0, f(x_1, 0)),\\
&f(x_1, 2) = 2x_1 = x_1 + x_1 = h(x_1, 1, f(x_1, 1)),\\
&f(x_1, 3) = 3x_1 = x_1 + 2x_1 = h(x_1, 2, f(x_1, 2)),\\
&\qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots
\end{aligned}$$

Это доказывает, что функция $x_1 \cdot x_2$ получается примитивной рекурсией из функций $o(x_1)$ и $h(x_1, x_2, x_3) = x_1 + x_3$. $\square$

6.6.5. *Примитивно рекурсивной называется функция, которая либо является простейшей, либо может быть получена из простейших и констант при помощи конечного числа операций суперпозиции и примитивной рекурсии.*

Пример 1. В примере к определению 6.6.4 показано, что бинарную функцию умножения можно получить примитивной рекурсией из бинарной функции сложения и простейшей функции $o(x)$. Докажем, что функцию $u(x,y) = x + y$ можно получить операциями суперпозиции и примитивной рекурсии из функций $e_1^1,\ e_3^3,\ s$. Действительно,

$$\begin{aligned} u(x,0) &= x = e_1^1(x), \\ u(x,y+1) &= x + (y+1) = (x+y) + 1 = \\ &= (e_3^3(x,y,(x+y))) + 1 = \\ &= s(e_3^3(x,y,(x+y))) = s(e_3^3(x,y,u(x,y))). \end{aligned}$$

Согласно 6.6.4 это означает, что функция $u(x,y)$ получается примитивной рекурсией из функций $g(x) = e_1^1(x)$ и

$$h(x,y,z) = s(e_3^3(x,y,z)) = z + 1.$$

Функция $h(x,y,z)$ примитивно рекурсивная, так как получена суперпозицией из функций $s(x)$ и $e_3^3(x,y,z)$, поэтому примитивно рекурсивной является также функция $u(x,y) = x + y$, а вместе с ней и функция $f(x_1,x_2) = x_1 \cdot x_2$. □

Пример 2. Докажем, что примитивно рекурсивной является функция

$$f(x,y) = x \dot{-} y = \begin{cases} x - y, & \text{если } x \geqslant y, \\ 0, & \text{если } x < y, \end{cases}$$

называемая **усеченной** (или **арифметической**) разностью. Действительно, одноместная функция $g(x) = x \dot{-} 1$ получается примитивной рекурсией из числа 0 и функции e_1^2:

$$g(0) = 0 \dot{-} 1 = 0, \quad g(x+1) = (x+1) \dot{-} 1 = x = e_1^2(x, g(x)).$$

Функция $h(x,y,z) = z \dot{-} 1 = x \dot{-} 1$ является примитивно рекурсивной, так как возникает при суперпозиции примитивно рекурсивных функций:

$$h(x,y,z) = g(e_3^3(x,y,z)).$$

Функция $f(x,y) = x \dot{-} y$ может быть получена из функций e_1^1 и $h(x,y,z)$ примитивной рекурсией:

$$\begin{aligned} f(x,0) &= x \dot{-} 0 = x = e_1^1(x), \\ f(x,y+1) &= x \dot{-} (y+1) = (x \dot{-} y) \dot{-} 1 = \\ &= f(x,y) \dot{-} 1 h(x,y,f(x,y))\,. \end{aligned}$$ □

Предположим, что нам надо найти одно решение уравнения $g(x) = y$, в котором g — вычислимая функция. Поскольку эта функция определена на множестве целых неотрицательных чисел, процесс решения может быть организован следующим образом.

Вычисляем $g(0)$ и сравниваем с y. Если $g(0) \neq y$, то вычисляем $g(1)$ и сравниваем с y, и т. д. Если при всех рассматриваемых значениях аргумента функция g определена и уравнение имеет решение, то мы обязательно найдем такое $k \in \mathbb{N}_0$, что $g(k) = y$. Если же уравнение решения не имеет, то наш процесс будет длиться бесконечно и ни к какому результату не приведет. Также мы не получим результата и в том случае, когда уравнение имеет решение, но для некоторого n для всех $x < n$ значения $g(x)$ определены, но не равны y, а значение $g(n)$ не определено.

Используя указанную процедуру, мы можем из функции g получить новую функцию f, являющуюся аналогом функции, обратной к g. Если уравнение $g(x) = y$ имеет решение, значения $g(0)$, $g(1)$, …, $g(k-1)$ определены и не равны y, а значение $g(k)$ определено и равно y, то полагаем $f(y) = k$. Во всех остальных случаях считаем, что $f(y)$ имеет неопределенное значение.

Аналогичным образом можно получать новые функции из функций с любым числом аргументов. Пусть нам известна некоторая функция $g(x_1, x_2, \ldots, x_n)$, значения которой там, где она определена, мы умеем вычислять. Мы хотим определить новую вычислимую функцию f, зависящую от того же числа аргументов. Задаем набор чисел $x_1, x_2, \ldots, x_n$. Вычисляем $g(x_1, x_2, \ldots, x_{n-1}, 0)$. Если это значение не определено, то и значение $f(x_1, x_2, \ldots, x_n)$ не определено. Если же $g(x_1, x_2, \ldots, x_{n-1}, 0)$ определено, то сравниваем это число с x_n. При

$$g(x_1, x_2, \ldots, x_{n-1}, 0) = x_n \tag{6.6.1}$$

считаем, что $f(x_1, x_2, \ldots, x_n) = 0$. Если же равенство (6.6.1) неверно, вычисляем $g(x_1, x_2, \ldots, x_{n-1}, 1)$ и сравниваем с x_n, и т. д.

6.6.6. *Говорят, что функция f, имеющая n аргументов, получается из n-местной функции g* ***операцией минимизации****, если для любых чисел*

$x_1, x_2, \ldots, x_n, k$ *из* $\mathbb{N}_0$ *равенство* $f(x_1, \ldots, x_n) = k$ *выполняется тогда и только тогда, когда для всех* $i < k$ *значения* $g(x_1, \ldots, x_{n-1}, i)$ *определены и отличны от* x_n*, а значение* $g(x_1, \ldots, x_{n-1}, k)$ *определено и равно* x_n*.*

6.6.7. *Утверждение о том, что функция* f *получена из* g *операцией минимизации может быть записано в виде формулы*

$$f(x_1, \ldots, x_n) = \mu_y(g(x_1, \ldots, x_{n-1}, y) = x_n)\,.$$

Знак μ_y читается так: «Наименьшее y такое, что».

Пример. Пусть

$$g(y, z) = y + z, \quad f(x, y) = \mu_z(g(y, z) = x)\,.$$

Чтобы найти значение $f(x, y)$, мы должны последовательно вычислять значения $g(y, 0)$, $g(y, 1)$, ... и сравнивать с x до тех пор, пока очередное значение окажется неопределенным, либо совпадет с x. Однако в нашем случае все вычисляемые значения определены, так как функция $+$ всюду определенная. Очевидно, равенство $y + k = x$ выполняется только тогда, когда $k = x - y$. Это означает, что $f(x, y) = x - y$. □

6.6.8. *Функция называется* ***частично рекурсивной****, если она может быть получена из простейших функций конечным числом операций суперпозиции, примитивной рекурсии и минимизации.*

Отметим, что примитивно рекурсивные функции являются всюду определенными и их значения определены при всех значениях аргументов. Однако при использовании операции минимизации из них могут получаться *частичные* функции, значения которых на некоторых наборах значений аргументов не определены. Так, функция $x - y$ из последнего примера не определена при $y > x$. (Напомним, что $x, y \in \mathbb{N}_0$.)

6.6.9. *Всюду определенные частично рекурсивные функции называются* ***общерекурсивными****.*

6.6.10 (тезис Черча)**.** *Класс вычислимых функций, определенных на множестве* $\mathbb{N}_0$*, совпадает с классом всех частично рекурсивных функций.*

Иными словами, каждая вычислимая в интуитивном смысле функция является частично рекурсивной, и любая частично рекурсивная функция является вычислимой. Как и в случае тезиса Тьюринга, тезис

Черча нельзя доказать; он является гипотезой, подтверждаемой всеми известными примерами.

Мы познакомились с двумя различными уточнениями понятия алгоритма — с частично рекурсивными функциями и с машинами Тьюринга. Из тезисов Черча и Тьюринга вытекает, что *класс функций, вычислимых на машинах Тьюринга, совпадает с классом частично рекурсивных функций*. Доказательство этого важного утверждения ввиду большого объема опустим.

Глава 7

ЭЛЕМЕНТЫ ТЕОРИИ ЧИСЕЛ

7.1. ДЕЛИМОСТЬ И ДЕЛИТЕЛИ

В этой главе, говоря о числах, мы подразумеваем числа целые, образующие множество

$$\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}.$$

Изучением свойств таких чисел занимается теория чисел.

7.1.1. *Пусть $a, b \in \mathbb{Z}$. Число a* ***делится*** *на b и b* ***делит*** *a, если найдется такое $c \in \mathbb{Z}$, что $a = bc$. Если a делится на b, то b называется* ***делителем*** *числа a и говорят, что a* ***кратно*** *b.*

Поскольку мы рассматриваем только целые числа, модуль любого делителя произвольного числа a не может быть больше модуля этого числа, поэтому число делителей у a конечно.

Примеры. Так как

$$30 = 6 \cdot 5, \quad -56 = -7 \cdot 8, \quad 12 = 4 \cdot 3,$$

можно сказать, что 30 кратно 5, 8 является делителем числа -56, число 12 делится на 3, 6 делит число 30. □

Для любой пары целых чисел (a, b) либо a делится на b, либо не делится, поэтому определение 7.1.1 задает на множестве $\mathbb{Z}$ бинарное отношение делимости. Это отношение транзитивно.

7.1.2. *Если a делится на b, а b делится на c, то и a делится на c.*

Доказательство. Согласно 7.1.1 найдутся такие числа p и q, что $a = pb$ и $b = qc$. Но тогда $a = pqc$. □

7.1.3. *Если числа $a_1, a_2, \ldots, a_{n-1}$ делятся на b и*

$$a_1 + a_2 + \ldots + a_n = 0, \tag{7.1.1}$$

то и a_n делится на b.

Доказательство. По условию существуют такие числа $c_1, c_2, \ldots, c_{n-1}$, что

$$a_1 = c_1 b,\ a_2 = c_2 b,\ \ldots,\ a_{n-1} = c_{n-1} b.$$

Подставляя правые части в (7.1.1) и перенося a_n влево, получим равенство

$$a_n = -(c_1 b + c_2 b + \ldots + c_{n-1} b) = -b(c_1 + c_2 + \ldots + c_{n-1}),$$

показывающее, что a_n делится на b. □

7.1.4. *Если числа $a_1, \ldots, a_n$ делятся на b, то b называется их* ***общим делителем****.*

7.1.5. *Наибольший из общих делителей чисел $a_1, \ldots, a_n$ называется их* ***наибольшим общим делителем****, обозначается через* НОД$(a_1, \ldots, a_n)$*, сокращенно* НОД.

7.1.6. *Если наибольший общий делитель чисел $a_1, \ldots, a_n$ равен единице, то эти числа называются* ***взаимно простыми****.*

Замечание. Обычно рассматриваются только положительные общие делители.

Пример. Числа 15, 45, 60 имеют следующие общие делители: 1, 3, 5, 15. Число 15 является наибольшим общим делителем.

Числа 15 и 28 взаимно просты. □

Каждое не равное единице число имеет по крайней мере два делителя: оно делится на себя и на единицу. Особую роль играют числа, имеющие ровно два положительных делителя.

7.1.7. *Положительное число, имеющее ровно два положительных делителя, называется* ***простым****. Положительное число, имеющее более двух положительных делителей, называется* ***составным****. Поскольку число 1 имеет только один положительный делитель, оно не является ни простым, ни составным.*

Пример. Числа 2, 3, 5, 7, 11, 13 простые, а числа 4, 6, 8, 9, 10, 12 составные. □

7.1.8. *Простых чисел бесконечно много.*

Доказательство. Предположим, что имеется лишь конечное число простых чисел и перечислим их:

$$p_1, p_2, \ldots, p_n. \tag{7.1.2}$$

Рассмотрим число $q = p_1 \cdot p_2 \cdot \ldots \cdot p_n + 1$. Если это число простое, то оно должно содержаться в последовательности (7.1.2). Однако при делении числа q на p_i, $i = \overline{1,n}$, остаток равен 1, поэтому в последовательности (7.1.2) числа q нет.

Приходим к выводу, что либо список (7.1.2) неполон, либо число q составное и потому равно произведению нескольких простых чисел. Предположим последнее, и пусть r — один из простых сомножителей. Если он имеется в списке (7.1.2), то при делении q на r получим остаток 1, что невозможно. Видим, что и в этом случае имеется простое число, не принадлежащее последовательности (7.1.2). Это означает, что предположение о конечности числа простых чисел неверно. □

7.1.9. *Для любых двух положительных чисел a и b существует единственная пара неотрицательных чисел p и q такая, что $q < b$ и*

$$a = bp + q. \tag{7.1.3}$$

Доказательство. Сначала докажем существование чисел p и q. Если $b > a$, то условиям теоремы удовлетворяют числа $p = 0$, $q = a$. При $b = a$ подходят числа $p = 1$, $q = 0$.

Пусть $b < a$. Рассмотрим последовательность

$$1b,\ 2b,\ 3b, \ldots .$$

Поскольку она возрастает, найдется такое s, что

$$sb \le a, \quad (s+1)b > a.$$

Положим $p = s$, $q = a - sb$. Очевидно, равенство (7.1.3) будет выполнено. Так как $(s+1)b = sb + b > a$, справедливо неравенство $q < b$.

Теперь докажем, что условиям теоремы удовлетворяет единственная пара чисел p и q. Пусть имеется еще одна такая пара чисел p_1 и q_1, что $q < b$ и

$$a = bp_1 + q_1. \tag{7.1.4}$$

Из двух чисел q и q_1 одно не превосходит другое. Предположим, что $q \le q_1$. Вычтем равенство (7.1.4) из (7.1.3) и перенесем член $q - q_1$ влево:

$$0 = (p - p_1)b + (q - q_1), \quad q_1 - q = (p - p_1)b.$$

По условию $b > 0$. Если $q_1 - q = 0$, то и $p - p_1 = 0$ и потому

$$q = q_1, \quad p = p_1.$$

При $q_1 - q > 0$ справедливо неравенство $p - p_1 > 0$, так как произведение $(p - p_1)b$ положительное. Из $q < b$ и $q_1 < b$ следует, что левая часть равенства (7.1.4) меньше b. В правой части того же равенства b умножается на число $p - p_1 \geq 1$, поэтому эта часть не меньше b и равенство (7.1.4) неверно. □

7.1.10. *Пусть a и b — положительные числа и a представлено в виде* (7.1.3)*, где $q < b$. Числа a, b, p и q называются соответственно* ***делимым, делителем, (неполным) частным*** *и* ***остатком от деления*** *a* ***на*** *b.*

Пример. Поскольку $17 = 3 \cdot 5 + 2$ и $2 < 3$, частное от деления числа 17 на 3 равно 5, а остаток равен 2. □

Как известно, найти наибольший общий делитель двух чисел можно следующим образом. Разложить каждое из этих чисел в произведение простых сомножителей. Выбрать все сомножители, общие для этих разложений, и перемножить. Получим НОД рассматриваемых чисел.

Этот способ, удобный в случаях, когда разложения легко находятся, в общем случае мало пригоден. Более практичный метод нахождения наибольшего общего делителя двух чисел основывается на равенстве (7.1.4).

7.1.11 (алгоритм Евклида)**.** *Ищем наибольший общий делитель положительных чисел a и b.*

1. *Разделим a на b. Получим частное p_1 и остаток q_1. Если $q_1 = 0$, то b=*НОД(a, b).

2. *Если $q_1 \neq 0$, то разделим b на q_1. Получим частное p_2 и остаток q_2. Если $q_2 = 0$, то q_1=*НОД(a, b).

3. *Если $q_2 \neq 0$, то разделим q_1 на q_2, и т. д.*

Доказательство. В соответствии с алгоритмом получаем цепочку равенств:

$$\begin{aligned}
a &= bp_1 + q_1, && 0 < q_1 < b,\\
b &= q_1p_2 + q_2, && 0 < q_2 < q_1,\\
p_2 &= q_2p_3 + q_3, && 0 < q_3 < q_2,\\
&\ldots\ldots\ldots\ldots\ldots\ldots && \ldots\ldots\ldots\ldots\ldots\\
p_{n-2} &= q_{n-2}p_{n-1} + q_{n-1}, && 0 < q_{n-1} < q_{n-2},\\
p_{n-1} &= q_{n-1}p_n.
\end{aligned}$$

Процесс деления оборвется, и на каком-то шаге остаток равен нулю, так как числа $q_1, \ldots, q_{n-1}$ неотрицательные и $q_1 > q_2 > \ldots > q_{n-1}$.

Воспользуемся утверждением 7.1.1. Первое из равенств показывает, что если c — общий делитель чисел a и b, то q_1 кратно c. Так как c делит b и q_1, то оно делит q_2, и т. д. Выполнение предпоследнего равенства означает, что b делит q_{n-1}.

Видим, что все общие делители чисел a и b являются делителями числа q_{n-1}. Среди них находится и НОД чисел a и b.

Из последнего равенства видно, что q_{n-1} является делителем числа p_{n-1}. Ввиду 7.1.1 предпоследнее равенство показывает, что q_{n-1} делит число p_{n-2}, и т. д. Из первого равенства следует, что q_{n-1} — делитель числа a. Это означает, что q_{n-1} — НОД чисел a и b. □

Пример. Найдем НОД чисел 420 и 1029. Делим с остатком 1029 на 420:

$$1029 = 420 \cdot 2 + 189.$$

Делим с остатком 420 на 189:

$$420 = 189 \cdot 2 + 42.$$

Делим с остатком 189 на 42:

$$189 = 42 \cdot 4 + 21.$$

Последнее деление:

$$42 = 21 \cdot 2.$$

НОД чисел 420 и 1029 равен 21. □

7.1.12. *Наибольший общий делитель d положительных чисел a и b всегда можно представить в виде*

$$d = au + bv,$$

где u и v — целые числа.

Доказательство. Обозначим через S множество всех положительных чисел вида $ai + bj$. В этом множестве есть наименьшее число m. Пусть $m = ak_1 + bk_2$.

Поскольку $a = 1 \cdot a + 0$, число a принадлежит множеству S, а потому $m \leq a$. Разделив a на m с остатком, найдем такие $q > 0$ и $r < m$, что $a = mq + r$. Так как $m = ak_1 + bk_2$, получаем равенство

$$a = (ak_1 + bk_2)q + r,$$

откуда

$$r = a - (ak_1 + bk_2)q = a(1 - k_1q) + b(-k_2q).$$

Видим, что если $r \neq 0$, то $r \in S$. Однако $r < m$, а m — наименьшее из чисел, принадлежащих S. Это означает, что $r = 0$, т. е. что a делится на m.

Аналогичным образом можно доказать, что b делится на m. Таким образом, m — общий делитель чисел a и b, а потому делит и НОД этих чисел d. Получаем равенства

$$d = tm = t(ak_1 + bk_2) = a(tk_1) + b(tk_2) = au + bv. \qquad \square$$

7.1.13 (следствие)**.** *Положительные числа a и b взаимно просты тогда и только тогда, когда*

$$au + bv = 1 \tag{7.1.5}$$

для подходящих чисел u и v.

Доказательство. «⇒» Если a и b взаимно просты, то число 1 является наибольшим их общим делителем, поэтому множители u и v существуют (см. 7.1.12).

«⇐» Пусть существуют такие числа u и v, что равенство (7.1.5) выполняется. Любой общий делитель чисел a и b должен быть делителем числа 1 и потому равен либо 1, либо -1. Это означает, что числа a и b взаимно просты. $\square$

7.1.14. *Если $a > 0$ взаимно просто с каждым из положительных чисел $b_1, \ldots, b_n$, то a взаимно просто и с их произведением $b_1 \ldots b_n$.*

Доказательство. Поскольку НОД$(a, b_i) = 1$, $i = \overline{1,n}$, то согласно 7.1.12 существуют такие числа $u_1, \ldots, u_n$, $v_1, \ldots, v_n$, что

$$1 = au_1 + b_1v_1, \ \ldots, \ 1 = au_n + b_nv_n.$$

Перемножим эти равенства:

$$1 = (au_1 + b_1v_1) \cdot \ldots \cdot (au_n + b_nv_n)\,. \tag{7.1.6}$$

После перемножения в правой части возникнет сумма, в которой все слагаемые содержат сомножитель a, за исключением произведения

$$b_1 \cdot \ldots \cdot b_nv_1 \cdot \ldots \cdot v_n\,.$$

Вынесем a из содержащих его слагаемых за скобки, а число в скобках обозначим через u. Если обозначить произведение $v_1 \cdot \ldots \cdot v_n$ через v, то равенство (7.1.6) превратится в

$$1 = au + (b_1 \cdot \ldots \cdot b_n)v\,,$$

означающее согласно 7.1.5, что числа a и $b_1 \cdot \ldots \cdot b_n$ взаимно просты. $\square$

7.2. СРАВНЕНИЯ И КЛАССЫ ВЫЧЕТОВ

7.2.1. *Пусть m, n, k — целые числа и $k > 0$. Говорят, что числа m и n **сравнимы по модулю** k, если остатки от деления этих чисел на k совпадают. Обозначение:* $m \equiv n (\mathrm{mod}\ k)$.

Пример. Остатки от деления чисел 5 и 7 на 3 равны 2 и 1, поэтому по модулю 3 они не сравнимы. При делении этих же чисел на 2 в остатке получаем единицу, следовательно, $5 \equiv 7 (\mathrm{mod}\ 2)$. □

Нетрудно убедиться в том, что введенное в 7.2.1 отношение на множестве $\mathbb{Z}$ является эквивалентностью (см. 1.2.10).

7.2.2. *Числа m и n сравнимы по модулю k тогда и только тогда, когда число $m - n$ делится на k.*

Доказательство. «⇒» Пусть остаток от деления чисел m и n на k равен r, тогда

$$m = pk + r, \quad n = qk + r$$

для подходящих целых чисел p и q. Разность

$$m - n = pk - qk = (p - q)k$$

делится на k.

«⇐» Пусть

$$m = pk + r_1, \quad n = qk + r_2.$$

Если $m - n = sk$ для какого-то $s \in \mathbb{Z}$, то

$$m - n = (pk - qk) + (r_1 - r_2),$$

$$0 = (m - n) - sk = (pk - qk) - sk + (r_1 - r_2),$$

$$r_1 - r_2 = sk + (qk - pk) = (s + q - p)k.$$

Так как $0 \leq r_1 < k$ и $0 \leq r_2 < k$, то $|r_1 - r_2| < k$ и потому

$$s + q - p = 0, \quad r_1 - r_2 = 0, \quad r_1 = r_2.$$ □

7.2.3 (следствие)**.** *Числа a и b сравнимы по модулю k тогда и только тогда, когда найдется такое $m \in \mathbb{Z}$, что $a = b + mk$.* □

7.2.4. *Если $a \equiv b (\mathrm{mod}\ k)$ и $c \equiv d (\mathrm{mod}\ k)$, то $a + c \equiv b + d (\mathrm{mod}\ k)$.*

Доказательство. Воспользуемся утверждением 7.2.2. Разности $a - b$ и $c - d$ делятся на k нацело, поэтому нацело на k делится и разность

$$(a + c) - (b + d) = (a - b) + (c - d).$$ □

7.2.5. *Множество всех целых чисел, дающих при делении на $k \in \mathbb{Z}$ один и тот же остаток, называется* ***классом вычетов по модулю*** *k, а числа из этого множества называются* ***вычетами****. Класс, содержащий все целые числа, дающие при делении на k остаток i, обозначим через $[i]$. Обозначим через Z_k совокупность всех классов вычетов по модулю k:*

$$Z_k = \{[0], \ldots, [k-1]\}\,.$$

Пример. По модулю 4 имеются следующие классы вычетов:

$$\begin{aligned}
[0] &= \{\ldots, -8, -4, 0, 4, 8, \ldots\},\\
[1] &= \{\ldots, -7, -3, 1, 5, 9, \ldots\},\\
[2] &= \{\ldots, -6, -2, 2, 6, 10, \ldots\},\\
[3] &= \{\ldots, -5, -1, 3, 7, 11, \ldots\}.
\end{aligned}$$ □

Поскольку при делении целого числа на $k \in \mathbb{Z}$ в остатке могут быть лишь числа 0, 1, ..., $k-1$, множество $\mathbb{Z}$ представляет собой объединение k классов вычетов по модулю k, при этом никакие два класса не пересекаются. Иначе говоря, мы разбиваем множество $\mathbb{Z}$ на k подмножеств (см. 2.3.1).

7.2.6. *Пусть $a_1, a_2 \in [i]$, $b_1, b_2 \in [j]$. Если $a_1 + b_1 \in [u]$, то и $a_2 + b_2 \in [u]$.*

Доказательство. По условию $a_1 \equiv a_2 (\mathrm{mod}\ k)$ и $b_1 \equiv b_2 (\mathrm{mod}\ k)$. Согласно 7.2.4 $a_1 + b_1 \equiv a_2 + b_2 (\mathrm{mod}\ k)$. Последнее означает, что числа $a_1 + b_1$ и $a_2 + b_2$ принадлежат одному и тому же классу вычетов. □

7.2.7 (следствие)**.** *Сравнения по одному и тому же модулю можно почленно складывать.* □

Каждый класс вычетов содержит как положительные, так и отрицательные числа, поэтому утверждение 7.2.6 справедливо и в случае замены символа сложения вычитанием. Это позволяет ввести операции сложения и вычитания классов вычетов следующим образом.

7.2.8. *Пусть $a \in [i]$, $b \in [j]$. Положим*

$$[i] + [j] = [u] \iff a + b \in [u]\,, \quad [i] - [j] = [v] \iff a - b \in [v]\,.$$

Пример. По модулю три операция сложения задается табл. 7.1. □

7.2.9. *Пусть $a_1, a_2 \in [i]$, $b_1, b_2 \in [j]$, где $[i]$ и $[j]$ — классы вычетов по модулю k. Если $a_1 \cdot b_1 \in [u]$, то и $a_2 \cdot b_2 \in [u]$.*

Таблица 7.1

$+$	$[0]$	$[1]$	$[2]$
$[0]$	$[0]$	$[1]$	$[2]$
$[1]$	$[1]$	$[2]$	$[0]$
$[2]$	$[2]$	$[0]$	$[1]$

Доказательство. Согласно 7.2.3

$$a_1 = b_1 + pk, \quad a_2 = b_2 + qk$$

для подходящих $p, q \in \mathbb{Z}$. Перемножим соответствующие части этих равенств:

$$a_1 a_2 = b_1 b_2 + (pqk + b_1 q + b_2 p)k\,.$$

Из 7.2.3 следует, что числа $a_1 a_2$ и $b_1 b_2$ сравнимы по k, т. е. принадлежат одному и тому же классу вычетов. □

7.2.10 (следствие)**.** *Сравнения по одному и тому же модулю можно почленно умножать.* □

Определим теперь операцию умножения классов вычетов способом, аналогичным введению сложения.

7.2.11. *Пусть $a \in Z(i)$, $b \in Z(j)$. Положим*

$$[i][j] = [u] \iff ab \in [u]\,.$$

Пример. По модулю 3 операция умножения задается табл. 7.2. □

Таблица 7.2

$\cdot$	$[0]$	$[1]$	$[2]$
$[0]$	$[0]$	$[0]$	$[0]$
$[1]$	$[0]$	$[1]$	$[2]$
$[2]$	$[0]$	$[2]$	$[1]$

Чтобы задать класс вычетов по какому-нибудь модулю, достаточно указать любое принадлежащее этому классу число. Обычно таким

числом служит либо наименьшее по модулю, либо наименьшее положительное. Задав по одному вычету из каждого класса, мы укажем все классы вычетов по данному модулю.

7.2.12. *Система вычетов по фиксированному модулю называется* ***полной****, если она содержит в точности по одному представителю из каждого класса вычетов.*

Вычет, равный остатку от деления его на модуль, называется ***наименьшим неотрицательным****.*

Принадлежащее классу вычетов число, модуль которого не превосходит модуля любого другого вычета из данного класса, называется ***абсолютно наименьшим вычетом****.*

Пример. Числа

$$21, \quad 15, \quad 9, \quad 10, \quad -3, \quad -9, \quad 20$$

образуют полную систему вычетов по модулю 7, так как при делении на 7 дают соответственно остатки

$$0, \quad 1, \quad 2, \quad 3, \quad 4, \quad 5, \quad 6.$$

Система, образованная этими остатками, также является полной и образована наименьшими неотрицательными вычетами.

Следующие числа образуют полную систему абсолютно наименьших вычетов по тому же модулю:

$$-3, \quad -2, \quad -1, \quad 0, \quad 1, \quad 2, \quad 3. \qquad \square$$

7.3. НЕКОТОРЫЕ СВОЙСТВА СРАВНЕНИЙ

7.3.1. *Если числа m и k взаимно просты, то из истинности сравнения $am \equiv bm (\mathrm{mod}\ k)$ следует истинность сравнения $a \equiv b (\mathrm{mod}\ k)$.*

Доказательство. Согласно 7.2.3 из $am \equiv bm (\mathrm{mod}\ k)$ следует, что

$$(a-b)m = am - bm = kp$$

для некоторого целого числа p. Поскольку НОД чисел m и k равен 1, все простые множители числа m не могут быть делителями числа k и потому делят p. Это означает, что p делится на m и $a - b = ks$ для какого-то целого s. Последнее равенство означает, что $a \equiv b (\mathrm{mod}\ k)$. $\square$

Таким образом, разделив обе части истинного сравнения на общий множитель, взаимно простой с модулем, мы получим истинное сравнение по тому же модулю. Если же делитель не взаимно прост с модулем, то результат может быть иным.

Пример. Числа 28 и 42 делятся на 14, поэтому $28 \equiv 42 (\mathrm{mod}\ 14)$. Разделив обе части на 2, получим неверное сравнение $14 \equiv 21 (\mathrm{mod}\ 14)$. □

7.3.2. *Разделив обе части истинного сравнения и модуль на общий множитель, получим истинное сравнение.*

Доказательство. Пусть $a \equiv b (\mathrm{mod}\ k)$. Согласно 7.2.3 существует такое m, что $a = b + mk$. Если d — общий множитель чисел $a,\ b,\ k$, то

$$a = a_1 d, \quad b = b_1 d, \quad k = k_1 d$$

для подходящих чисел $a_1,\ b_1,\ k_1$. Получаем равенства

$$a_1 d = b_1 d + m k_1 d, \quad a_1 d = b_1 d + m k_1 d.$$

Последнее из них означает, что $a_1 \equiv b_1 (\mathrm{mod}\ k_1)$ (см. 7.2.3). □

7.3.3. *Если p — простое число, то*

$$(a+b)^p \equiv a^p + b^p (\mathrm{mod}\ p) \tag{7.3.1}$$

для любых целых чисел a и b.

Доказательство. Вспомним формулу бинома Ньютона (2.2.4):

$$(a+b)^p = \sum_{k=0}^{p} C_p^k a^{p-k} b^k \,. \tag{7.3.2}$$

Так как p — простое число и биномиальные коэффициенты

$$C_p^k = \frac{p(p-1)\ldots(p-k+1)}{k!}$$

являются целыми числами, при $k \notin \{0, p\}$ каждый сомножитель в знаменателе является делителем некоторого сомножителя в числителе, отличного от p. После выполнения деления получится число, делящееся на p. Таким образом, в правой части равенства (7.3.2) все слагаемые, кроме первого и последнего, делятся на p, и равенство можно переписать в виде

$$(a+b)^p = C_p^0 a^p + C_p^p b^p + hp = a^p + b^p + hp \,,$$

где h — некоторое целое число. Отсюда следует истинность доказываемого сравнения. □

7.3.4 (малая теорема Ферма)**.** *Если p — простое число и число a — целое, то $a^p \equiv a(\mathrm{mod}\ p)$.*

Доказательство. Воспользуемся индукцией и докажем теорему для неотрицательных значений a. При $a = 0$ и $a = 1$ утверждение справедливо. Пусть оно справедливо, если $a = m < p - 1$. Докажем, что тогда оно справедливо и для $a = m + 1$. Действительно, по утверждению 7.3.3

$$(m+1)^p \equiv m^p + 1^p (\mathrm{mod}\ p)\,,$$

а по индуктивному предположению $m^p \equiv m(\mathrm{mod}\ p)$. Видим, что

$$(m+1)^p \equiv m + 1(\mathrm{mod}\ p),$$

т. е. что $a^p \equiv a(\mathrm{mod}\ p)$.

Числа $0, 1, \ldots, p-1$ образуют полную систему вычетов, и поскольку они неотрицательные, для них доказываемое утверждение справедливо. Каждое отрицательное число a сравнимо по модулю p с каким-нибудь числом s, принадлежащим данной системе, и для него найдется такое k, что

$$a + pk \equiv s(\mathrm{mod}\ p)\,.$$

Но тогда

$$(a+pk)^p \equiv a^p + (pk)^p \equiv a^p \equiv s^p \equiv s(\mathrm{mod}\ p)\,. \quad \square$$

7.3.5. *Определенная на множестве $\mathbb{N}$ функция $\varphi(n)$, значение которой равно числу положительных целых чисел, меньших n и взаимно простых с n, называется* ***функцией Эйлера****.*

Пример. Так как $8 = 2 \cdot 2 \cdot 2$, то числами, меньшими 8 и взаимно простыми с 8, являются 1, 3, 5, 7 и потому $\varphi(8) = 4$.

Число 13 простое, поэтому с ним взаимно просты все числа от 1 до 12, следовательно, $\varphi(13) = 12$. $\square$

7.3.6. *Если $\varphi(n)$ — функция Эйлера и числа m и k взаимно просты, то*

$$\varphi(mk) = \varphi(m)\varphi(k)\,.$$

Доказательство. Изучим последовательности

$$\begin{array}{ccccc} 1, & 1+m, & 1+2m, & \ldots, & 1+(k-1)m, \\ 2, & 2+m, & 2+2m, & \ldots, & 2+(k-1)m, \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ m, & m+m, & m+2m, & \ldots, & m+(k-1)m. \end{array} \qquad (7.3.3)$$

Вместе они содержат все числа от 1 до mk без повторений, поэтому число $\varphi(mk)$ в точности равно количеству различных чисел, содержащихся в этих последовательностях и взаимно простых с mk. Если в некоторой строке первое число не является взаимно простым с m, то и остальные числа этой строки не взаимно просты с m и все числа строки можно во внимание не принимать. Таким образом, число $\varphi(mk)$ равно количеству различных чисел, взаимно простых с mk и содержащихся в строках, у которых первое число взаимно просто с m. Очевидно, число таких строк равно $\varphi(m)$.

Пусть положительное число a меньше m и взаимно просто с m. Рассмотрим последовательность

$$a, \quad a+m, \quad a+2m, \quad \ldots, \quad a+(k-1)m. \tag{7.3.4}$$

Предположим, что два члена этой последовательности сравнимы по модулю k:

$$a+im \equiv a+jm (\mathrm{mod}\ k)\,.$$

Согласно 7.2.2 число $(a+im)-(a+jm) = im-jm = (i-j)m$ делится на k. Поскольку m и k взаимно просты, k является делителем числа $i-j$. Это невозможно, так как модуль числа $i-j$ меньше k.

Видим, что последовательность (7.3.4) содержит k членов и никакие два из них не сравнимы по модулю k. Это означает, что члены последовательности образуют полную систему вычетов по модулю k. Полную систему вычетов по модулю k образуют также числа

$$0, \quad 1, \quad 2, \quad \ldots, \quad k-1, \tag{7.3.5}$$

поэтому каждый член последовательности (7.3.4) сравним по модулю k с одним из членов последовательности (7.3.5).

Если числа i и j сравнимы по модулю k, то $i = j+tk$ для некоторого t. Так как $i-j-tk = 0$, ввиду утверждения 7.1.3 общий делитель чисел i и k должен делить также число j. Это означает, что если некоторый вычет взаимно прост с модулем, то и остальные числа из того же класса вычетов взаимно просты с модулем. Отсюда следует, что число чисел, взаимно простых с k, в последовательностях (7.3.4) и (7.3.5) одинаково. Очевидно, в (7.3.5) число таких чисел равно $\varphi(k)$.

Мы подсчитали количество чисел, содержащихся в строке списка (7.3.3) и взаимно простых с k. Всего имеется $\varphi(m)$ таких строк, поэтому общее количество подобных чисел в (7.3.3) равно $\varphi(m)\varphi(k)$. Получаем нужное равенство $\varphi(mk) = \varphi(m)\varphi(k)$. $\square$

7.3.7. *Если $k \in \mathbb{N}$ и число p простое, то*

$$\varphi(p^k) = p^{k-1} \cdot (p-1)\,.$$

Доказательство. Поскольку число p простое, оно имеет ровно два делителя: p и 1. Отсюда следует, что любое число не взаимно просто с p тогда и только тогда, когда оно делится на p. Не превосходят p^k и делятся на p следующие натуральные числа:

$$p,\quad 2p,\quad 3p,\quad \ldots,\quad p^{k-1}p.$$

В этом списке p^{k-1} чисел. Остальные натуральные числа, не превосходящие p^k, взаимно просты с p. Таких чисел $p^k - p^{k-1}$. Ввиду 7.1.14 они также взаимно просты с p^k, поэтому

$$\varphi(p^k) = p^k - p^{k-1} = p^{k-1}(p-1)\,. \qquad \square$$

7.3.8. *Пусть $n = p_1^{k_1} \cdot p_2^{k_2} \cdot \ldots \cdot p_s^{k_s}$, где $k_i \in \mathbb{N}$ и число p_i простое для любого i. Если числа $p_1, \ldots, p_s$ попарно различны, то*

$$\varphi(n) = p_1^{k-1} \cdot p_2^{k-2} \cdot \ldots \cdot p_s^{k-s}(p_1-1) \cdot (p_2-1) \cdot \ldots \cdot (p_s-1)\,.$$

Доказательство. Используя утверждения 7.3.6 и 7.3.7, получаем

$$\begin{aligned}
\varphi(n) =& \varphi(p_1^{k_1} \cdot p_2^{k_2} \cdot \ldots \cdot p_s^{k_s}) = \\
=& \varphi(p_1^{k_1}) \cdot \varphi(p_2^{k_2}) \cdot \ldots \cdot \varphi(p_s^{k_s}) = \\
=& p_1^{k-1} \cdot (p_1-1) \cdot p_1^{k-1}(p_2-1) \cdot \cdot \ldots \cdot p_s^{k-s} \cdot (p_s-1) = \\
=& p_1^{k-1} \cdot p_2^{k-2} \cdot \ldots \cdot p_s^{k-s}(p_1-1) \cdot (p_2-1) \cdot \ldots \cdot (p_s-1)\,. \qquad \square
\end{aligned}$$

Для не слишком больших n это утверждение позволяет легко находить значения функции $\varphi(n)$.

Пример. Так как $700 = 2^2 \cdot 5^2 \cdot 7$, число чисел, не превосходящих 700 и взаимно простых с ним, совпадает с числом

$$\varphi(2^2 \cdot 5^2 \cdot 7) = 2^1 \cdot 5^1 \cdot 7^0 \cdot (2-1) \cdot (5-1) \cdot (7-1) = 240\,. \qquad \square$$

7.3.9. *Если $a \equiv b (\mathrm{mod}\ k)$, то* НОД$(a,k)$=НОД$(b,k)$.

Доказательство. Согласно 7.2.3 существует такое m, что $a = b + mk$, поэтому $a - mk - b = 0$. Пусть c является общим делителем чисел a и k. Поскольку c делит a и mk, то c делит и b. Иначе говоря, каждый общий делитель чисел a и k, в том числе и НОД(a,k), является общим делителем чисел b и k. Аналогично доказывается, что НОД(b,k) является общим делителем чисел a и k. Это означает, что НОД(a,k) = НОД(b,k). $\square$

7.3.10 (следствие)**.** *Числа, принадлежащие одному и тому же классу вычетов, имеют одинаковые наибольшие общие делители с модулем.* □

7.3.11. *Удалив из полной системы вычетов числа, имеющие с модулем общие не равные* 1 *делители, получим систему вычетов, называемую* ***приведенной****.*

Пример. Числа $0, 1, 2, 3, 4, 5$ образуют по модулю 6 полную систему вычетов, а числа $1, 5$ — приведенную систему. □

7.3.12. *Если число a взаимно просто с k и числа $b_1, \ldots, b_m$ образуют приведенную систему вычетов по модулю k, то числа $ab_1, \ldots, ab_m$ также образуют приведенную систему вычетов.*

Доказательство. Числа в системе $ab_1, \ldots, ab_m$ попарно не сравнимы. Действительно, если $ab_i \equiv ab_j (\mathrm{mod}\ k)$, то и $b_i \equiv b_j (\mathrm{mod}\ k)$, так как a и k взаимно просты (см. 7.3.1). Это невозможно, так как в приведенной системе вычетов числа попарно не сравнимы.

Поскольку a взаимно просто с k и каждое b_j, $j = \overline{1, m}$, взаимно просто с k, принадлежащие системе $ab_1, \ldots, ab_m$ числа взаимно просты с модулем. Их столько же, сколько чисел в системе $b_1, \ldots, b_m$, поэтому $ab_1, \ldots, ab_m$ — приведенная система вычетов. □

7.3.13 (теорема Эйлера)**.** *Если $k > 1$ и число a взаимно просто с k, то*

$$a^{\varphi(k)} \equiv 1 (\mathrm{mod}\ k)\,.$$

Доказательство. Рассмотрим приведенную систему $b_1, \ldots, b_m$ наименьших неотрицательных вычетов по модулю k. Обозначим через c_j, $j = \overline{1, m}$, наименьшее неотрицательное число, сравнимое с ab_j по модулю k. Получаем систему сравнений

$$ab_1 \equiv c_1 (\mathrm{mod}\ k),\ \ldots,\ ab_m \equiv c_m (\mathrm{mod}\ k)\,.$$

Сравнения по одному и тому же модулю можно перемножать (см. 7.2.10), поэтому

$$ab_1 ab_2 \cdots ab_m \equiv c_1 c_2 \cdots c_m (\mathrm{mod}\ k)\,,$$
$$a^m b_1 b_2 \cdots b_m \equiv c_1 c_2 \cdots c_m (\mathrm{mod}\ k)\,.$$

Согласно 7.3.12 числа $ab_1, \ldots, ab_m$ образуют приведенную систему вычетов, следовательно, в ней $\varphi(k)$ элементов. Так как $m = \varphi(k)$, получаем сравнение

$$a^{\varphi(k)} b_1 b_2 \cdots b_m \equiv c_1 c_2 \cdots c_m (\mathrm{mod}\ k)\,. \qquad (7.3.6)$$

Приведенную систему вычетов образуют также и числа $c_1, \ldots, c_m$. Поскольку системы

$$b_1, \ldots, b_m, \quad c_1, \ldots, c_m$$

являются приведенными и состоят из наименьших неотрицательных вычетов, они образованы одинаковыми числами и могут отличаться только расположением этих чисел. Отсюда следует, что

$$b_1 b_2 \cdots b_m = c_1 c_2 \cdots c_m \,.$$

Отметим, что эти произведения отличны от нуля, так как среди сомножителей нуль отсутствует.

Принадлежащие приведенной системе вычетов числа взаимно просты с модулем, поэтому их произведение также взаимно просто с модулем и обе части сравнения (7.3.6) можно на него разделить (см. 7.3.1). В результате получаем нужное сравнение

$$a^{\varphi(k)} \equiv 1 (\mathrm{mod}\ k). \qquad \square$$

Глава 8

АЛГЕБРАИЧЕСКИЕ СИСТЕМЫ

8.1. АЛГЕБРЫ С ОДНОЙ БИНАРНОЙ ОПЕРАЦИЕЙ

Раздел математики, называемый алгеброй, посвящен изучению **алгебраических систем**, состоящих из какого-то множества и некоторой совокупности определенных на этом множестве операций и предикатов. Обычно алгебраическая система обозначается следующим образом:

$$\mathbf{A} = \langle A;\ \{f_i \mid i \in I\},\ \{P_j \mid j \in J\}\rangle\,; \tag{8.1.1}$$

здесь f_i — определенные на A операции, а P_j — предикаты. Множество A называется **основным**, или **носителем** алгебраической системы.

Алгебраическая система, в которой отсутствуют предикаты, называется **(универсальной) алгеброй**. Слово «универсальная» подчеркивает общий характер объекта. Операции f_i в (8.1.1) называются **основными операциями** алгебры $\mathbf{A}$. Суперпозициями из них и из проекций можно получать новые операции, называемые **производными**. Универсальные алгебры называют также **абстрактными алгебрами** и **общими алгебрами**.

Пример. Алгебрами являются системы

$$\langle \mathbb{N};\ f^2\,\rangle, \quad \langle \mathbb{N};\ f^2,\ g^2\,\rangle,$$

в которых f^2 и g^2 — обычные операции сложения и умножения чисел. Упоминая эти алгебры, в записях чаще используют обычные обозначения операций, если это не вызывает путаницы:

$$\langle \mathbb{N};\ +\,\rangle, \quad \langle \mathbb{N};\ +,\ \cdot\,\rangle.$$

Граф можно рассматривать как алгебраическую систему $\langle V;\ E\,\rangle$, в которой множество вершин является носителем, а E — бинарный

предикат, истинный на тех парах вершин, которые соединены ребром. □

Дискретная математика не включает в себя теорию алгебраических систем несмотря на то, что используемые методы в разработке этих двух разделов математики часто совпадают. С одной стороны, теория алгебраических систем очень обширна. С другой, носителями таких систем могут быть непрерывные множества. Однако конечные универсальные алгебры находят важные приложения в различных разделах дискретной математики, поэтому стоит хотя бы очень бегло познакомиться с начальными понятиями этой теории.

Все универсальные алгебры делятся на классы. В основу классификации положены вид и определенные свойства основных операций этих алгебр. Наиболее простыми следует считать алгебры, у которых все основные операции одноместные.

8.1.1. *Универсальная алгебра, все операции которой одноместные, называется* ***унарной*** *или* ***уноидом****.*

Пример. Алгебра $\langle \mathbb{N};\ x+1, 2x \rangle$ является уноидом. □

8.1.2. *Унарная алгебра с единственной основной операцией называется* ***моноунарной*** *или* ***унаром****.*

Пример. Алгебра $\langle \mathbb{N};\ x+1 \rangle$ является унаром. □

8.1.3. *Универсальная алгебра с одной бинарной операцией называется* ***группоидом****.*

Основную операцию группоида обычно называют либо умножением, либо сложением. В первом случае группоид называется **мультипликативным**, во втором — **аддитивным**.

Примеры. 1. Группоидом является алгебра $\langle \mathbb{R};\ f^2 \rangle$, где f^2 — сложение.

2. Алгебра $\langle \{0,1\};\ \supset \rangle$, где $\supset$ — импликация, также является группоидом. □

8.1.4. *Универсальная алгебра называется* ***полугруппой****, если она имеет одну основную бинарную операцию и эта операция ассоциативна.*

Иначе говоря, полугруппа — это ассоциативный группоид, и если $*$ — операция полугруппы, то

$$x * (y * z) = (x * y) * z\,.$$

Примеры. 1. Алгебры

$$\langle \mathbb{N};\ + \rangle, \quad \langle \mathbb{N};\ \cdot \rangle$$

где $+$ и $\cdot$ — обычные операции сложения и умножения, являются полугруппами.

2. Пусть H_A — совокупность всех подмножеств непустого множества A. Система $\langle H_A;\ \cap \rangle$, где $\cap$ — операция пересечения множеств, является полугруппой.

3. Так как умножение матриц ассоциативно, относительно умножения полугруппу образует множество всех квадратных матриц с действительными элементами, имеющих один и тот же порядок.

4. Полугруппой является множество всех непрерывных функций, определенных на отрезке $[0, 1]$, с операцией подстановки.

5. Очевидно, полугруппами являются системы

$$\langle \{0, 1\};\ \& \rangle, \quad \langle \{0, 1\};\ \vee \rangle.$$

6. Рассмотрим систему, основное множество которой образуют всевозможные слова конечной длины в непустом алфавите, а основной операцией является соединение двух слов в новое слово. Поскольку эта операция ассоциативная, получим полугруппу. □

Иногда требуется, чтобы алгебраическая система содержала некоторые элементы, обладающие определенными свойствами. Эти элементы можно считать нульарными операциями, т. е. константами. Чаще всего это аналоги нуля и единицы в множестве действительных чисел.

8.1.5. *Элемент e мультипликативного группоида* $\mathbf{H}$ *с основной операцией $*$ называется* ***правой (левой) единицей****, если для любого элемента a справедливо равенство*

$$e * a = a \quad (a * e = a).$$

Нейтральным элементом, *или* ***единицей,*** *группоида называется элемент, являющийся одновременно и правой, и левой единицей.*

Аналогичным образом определяется **нуль** аддитивного группоида.

Пример. В полугруппе $\langle \mathbf{N};\ \cdot \rangle$ единица есть, но ее нет в полугруппе $\langle \{3, 9, 27, \ldots\};\ \cdot \rangle$. □

Как и раньше, константы можно считать нульарными операциями. Включая их в число основных операций, мы получаем особые алгебры, как показывает следующее определение.

8.1.6. *Полугруппа* $\mathbf{G} = \langle A;\ f^0, g^2 \rangle$, *в которой* f^0 *— единица, называется* ***полугруппой с единицей*** *или* ***моноидом****.*

Можно сказать и иначе: моноидом является полугруппа с выделенной единицей. Важно только заметить, что полугруппа, обладающая единицей, может и не быть моноидом, так как эта единица может быть не выделенной, т. е. не входить в число основных нульарных операций. Обычно единица моноида обозначается символами 1 или e.

Примеры. 1. Полугруппа $\langle \{1, 2, 4, \ldots\};\ 1, \cdot \rangle$ является моноидом.

2. Не является моноидом полугруппа $\langle \{1, 2, 4, \ldots\};\ \cdot \rangle$.

3. Множество всех квадратных матриц одного порядка с действительными элементами относительно умножения образует моноид, если в число основных операций добавить нульарную с постоянным значением E, где E — единичная матрица.

4. Множество всех слов конечной длины относительно бинарной операции соединения слов образует моноид, если в качестве единицы выделить пустое слово. □

8.1.7. *Пусть* $\mathbf{H}$ *— мультипликативная полугруппа с единицей и* $a \in \mathbf{H}$. *Элемент* $b \in \mathbf{H}$ *называется* ***обратным к*** a, *если* $ab = ba = e$.

Пример. В полугруппе $\langle \mathbb{Q};\ \cdot \rangle$, где $\mathbb{Q}$ — множество рациональных чисел, т. е. чисел вида $\frac{m}{n}$, где m и n — любые целые числа, исключая $n = 0$, элемент 2 является обратным к элементу $\frac{1}{2}$. □

8.1.8. *Если основная операция полугруппы называется сложением, то термины* ***единица полугруппы*** *и* ***обратный элемент*** *меняют на* ***нуль полугруппы*** *и* ***противоположный элемент****.*

Пример. В полугруппе $\langle \{0, \pm 1, \pm 2, \ldots\};\ + \rangle$ числа 5 и -5 взаимно противоположны, так как $5 + (-5) = 0$. □

8.2. ГРУППЫ

8.2.1. ***Группой*** *называется алгебра* $\mathbf{G}$ *с одной основной бинарной операцией, удовлетворяющая следующим трем условиям:*

1. Операция ассоциативна.

2. В G *есть единица.*

3. Для каждого элемента a *из* $\mathbf{G}$ *в* $\mathbf{G}$ *есть такой элемент* x, *что* $ax = xa = e$.

8.2.2. *Принадлежащий группе* G *элемент* x *называется* ***обратным к элементу*** $a \in G$ *и обозначается через* a^{-1}, *если он удовлетворяет равенствам* $ax = xa = e$.

Если $\mathbf{G} = \langle A;\ * \rangle$ — мультипликативная полугруппа, то указанные условия можно, как обычно, записать формулами:

1) $\forall a, b, c \in A\,(a * (b * c) = (a * b) * c)$,

2) $\exists e \in A\,\forall a \in A\,(a * e = e * a = e)$,

3) $\forall a \in A\,\exists x \in A\,(a * x = x * a = e)$.

Можно сказать и так: группой называется полугруппа, в которой есть единица, а основная операция обратима.

Если a и b — действительные числа, то $ab = ba$. Это означает, что операция умножения действительных чисел **коммутативна**. Хотя в группе основная операция и называется умножением, она может быть некоммутативной.

8.2.3. *Если операция в группе G коммутативна, т. е. если $ab = ba$ для любых a и b из G, то группа G называется* ***коммутативной*** *или* ***абелевой****.*

Примеры. 1. Группу образует множество целых чисел $\mathbb{Z}$ с операцией сложения. Действительно, для любых чисел $a, b, c \in \mathbb{Z}$ справедливо равенство

$$(a + b) + c = a + (b + c)\,.$$

Роль единицы в этой группе играет число 0:

$$\forall a \in \mathbb{Z}\ (a + 0 = 0 + a = a)\,.$$

Выполняется и третья аксиома:

$$\forall a \in \mathbb{Z}\ (-a \in \mathbb{Z}\ \&\ a + (-a) = 0)\,.$$

Поскольку операцией в этой группе является сложение, ее нейтральный элемент, т. е. число 0, называют **нулем**, а число $-a$ — **противоположным** к a. Эта группа абелева, так как $a + b = b + a$ для любых чисел $a, b \in \mathbb{Z}$.

2. Множества отличных от нуля рациональных чисел $\mathbb{Q}$ и отличных от нуля действительных чисел $\mathbb{R}$ с операцией умножения также являются абелевыми группами.

3. Множество всех неособенных (невырожденных) квадратных матриц с действительными элементами относительно операции умножения образует группу. Единицей в этом случае служит единичная матрица E. Кроме того, умножение матриц ассоциативно, и для каждой неособенной матрицы имеется к ней обратная. Эта группа некоммутативная. Действительно, пусть, например,

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}.$$

Матрицы A и B неособенные, так как их определители отличны от нуля. Легко убедиться в том, что при перестановке сомножителей произведение меняется:

$$AB = \begin{pmatrix} 2 & 1 \\ 3 & 2 \end{pmatrix}, \quad BA = \begin{pmatrix} 1 & 1 \\ 2 & 3 \end{pmatrix}.$$

4. Аддитивную группу образуют прямоугольные матрицы одинакового размера, все элементы которых действительны, если в качестве основной операции взять сложение матриц. Нулем служит матрица, у которой все элементы равны нулю, противоположной к произвольной матрице A является матрица $(-1) \cdot A$.

5. Зафиксируем какую-либо точку плоскости. Обозначим через a_φ поворот этой плоскости вокруг данной точки на угол φ. Будем считать, что направления поворотов плоскости с углами φ и ψ совпадают, если знаки у φ и ψ одинаковы, направления противоположны, если знаки разные. Последовательное выполнение двух поворотов сначала на угол φ, а затем на угол ψ назовем произведением. Очевидно, введенная операция коммутативна и ассоциативна:

$$a_\varphi a_\psi = a_\psi a_\varphi, \quad (a_\varphi a_\psi)a_\gamma = a_\varphi(a_\psi a_\gamma)\,.$$

Поворот на угол 0, т. е. отсутствие поворота, обозначим через e. Обратным к повороту a_φ назовем поворот на угол $-\varphi$. Совокупность всех поворотов плоскости с введенной операцией умножения поворотов образует абелеву группу. □

8.2.4. *В группе имеется только одна единица.*

Доказательство. Предположим, что e_1 и e_2 — две единицы, принадлежащие некоторой группе G. Это означает, что

$$a * e_1 = e_1 * a = e_1, \quad b * e_2 = e_2 * b = e_2$$

для любых $a, b \in G$. Положим $a = e_2$ и $b = e_1$:

$$e_2 * e_1 = e_1 * e_2 = e_1, \quad e_1 * e_2 = e_2 * e_1 = e_2.$$

Видим, что $e_1 = e_2$. □

8.2.5. *Для любых элементов a и b группы G уравнения*

$$ax = b, \quad ya = b \tag{8.2.1}$$

имеют решения, и эти решения единственны.

Доказательство. Из определения группы следует, что для a в G имеется обратный элемент a^{-1}. Умножим на него слева обе части первого из уравнений (8.2.1):

$$a^{-1}ax = a^{-1}b, \quad ex = a^{-1}b, \quad x = a^{-1}b.$$

Умножая обе части второго уравнения в (8.2.1) справа, аналогичным образом выясним, что $y = ba^{-1}$.

Пусть u и v — два решения уравнения $ax = b$. Это означает, что

$$au = b, \quad av = b.$$

Умножая обе части каждого равенства слева на a^{-1}, видим, что $u = ba^{-1} = v$.

Единственность решения уравнения $xa = b$ доказывается аналогично. □

8.2.6. *Пусть a и b — элементы группы G. Решения уравнений*

$$ax = b, \quad ya = b$$

называются ***левым*** *и* ***правым частными от деления*** b ***на*** a.

В абелевой группе левое частное совпадает с правым, и говорят просто о **частном от деления двух элементов**.

8.2.7. *Если группа содержит конечное число элементов, то она называется* ***конечной****, в противном случае группа называется* ***бесконечной****. Число элементов конечной группы называется ее* ***порядком****.*

8.2.8. *Умножая какой-нибудь элемент a группы G на себя n раз, мы получим некоторый элемент той же группы. Этот элемент называется* ***n-й степенью элемента*** *a и обозначается через a^n.*

8.2.9. *Нулевую и целые отрицательные степени элементов группы G вводим следующим образом: для любого $a \in G$*

$$a^0 = e, \quad a^{-n} = (a^n)^{-1}.$$

Поскольку умножение в группе является ассоциативной операцией, для любого элемента a из группы и любых целых чисел m и n справедливы равенства

$$a^m a^n = a^{m+n}, \quad (a^m)^n = a^{mn}.$$

8.2.10. *Пусть e — единица группы, $a \in G$ и $a \neq e$. Наименьшая положительная степень n, для которой $a^n = e$, называется* ***порядком элемента*** *a. Если равенство $a^n = e$ не выполняется ни для какого натурального n, то a называется* ***элементом бесконечного порядка****.*

8.2.11. *Если в группе G есть такой элемент u, что все остальные ее элементы являются степенями элемента u, то группа G называется* ***циклической****, а u называется* ***порождающим элементом группы*** *G.*

Если циклическая группа порождается элементом a и имеет конечный порядок n, то все ее элементы можно перечислить следующим образом:

$$e,\ a,\ a^2,\ a^3,\ \ldots,\ a^{n-1}.$$

Поскольку $a^n = e$, порядок элемента a также равен n.

Примеры. 1. Циклической является аддитивная группа целых чисел $\langle \mathbb{Z};\ + \rangle$. Она порождается числом 1.

2. Ранее была рассмотрена группа поворотов плоскости вокруг некоторой фиксированной точки. Рассмотрим только те повороты, для которых угол имеет вид $0,5 \cdot k\pi$. Обозначим поворот на такой угол через a_k. Так как после поворота на угол $0,5 \cdot 4\pi$ все точки плоскости возвращаются в исходное положение, группа таких поворотов является циклической и содержит только четыре элемента: $a_0 = e$, a_1, a_2, a_3.

3. Пусть $i = \sqrt{-1}$, тогда

$$i \cdot i = \sqrt{-1} \cdot \sqrt{-1} = -1, \quad -1 \cdot i = -\sqrt{-1} = -i,$$
$$-i \cdot i = (-\sqrt{-1}) \cdot \sqrt{-1} = 1, \quad -i \cdot -i = (-\sqrt{-1}) \cdot (-\sqrt{-1}) = 1,$$

поэтому система $\langle \{1, -1, i, -i\};\ \cdot \rangle$ является группой четвертого порядка. Эта группа циклическая, порождается элементом i. □

В разделе, посвященном теории графов, мы уже встречались с понятием изоморфизма. Напомним, что алгебраические свойства изоморфных объектов полностью совпадают. Для групп определение изоморфизма выглядит следующим образом.

8.2.12. *Группы G_1 и G_2 называются* ***изоморфными****, если существует функция ψ, взаимно однозначно отображающая множество элементов группы G_1 на множество элементов группы G_2 и сохраняющая операцию умножения: для любых элементов a, b, c из G_1 если $ab = c$, то в G_2 должно выполняться равенство*

$$\psi(a)\psi(b) = \psi(c).$$

При выполнении этих условий функция ψ называется ***изоморфизмом****.*

Пример. Мультипликативная группа $G_1 = \langle \mathbb{R}^+;\ \cdot \rangle$, где $\mathbb{R}^+$ — множество положительных действительных чисел, изоморфна аддитивной группе $G_2 = \langle \mathbb{R};\ + \rangle$. Изоморфное отображение группы G_1 на группу G_2 всем знакомо из курса элементарной математики: $\psi(x) = \log x$. □

8.2.13. *Если все отличные от нейтрального элементы группы имеют бесконечный порядок, то она называется* ***группой без кручения****.*

Пример. В группе $\langle \mathbb{Z};\ + \rangle$, состоящей из целых чисел и операции сложения, все отличные от нейтрального элементы группы имеют бесконечный порядок. □

8.2.14. *Группа называется* ***периодической****, если порядок каждого ее элемента конечен.*

Пример. Периодической является группа $\langle \{-1, 1\};\ \cdot \rangle$. □

8.2.15. *Периодическая группа, у которой порядок каждого элемента является степенью фиксированного простого числа p, называется* ***p-группой****.*

Пример. Пусть p — простое число и $S = \{0, 1, \ldots, p^n - 1\}$ — полная система наименьших неотрицательных вычетов по модулю p^n. Система $\langle S;\ + \rangle$ является аддитивной p-группой. Действительно, если порядок элемента $m \in S$ равен k, то

$$\underbrace{m + \ldots + m}_{k \text{ раз}} \equiv 0 (\mathrm{mod}\ p^n).$$

Из $km \equiv 0 (\mathrm{mod}\ p^n)$ следует, что числа k и m являются степенями числа p. □

Значение теории групп в математике в значительной степени определяется тем, что группу можно интерпретировать как группу взаимно однозначных отображений множеств, а если необходимо, то множеств с заданными свойствами. Мы рассмотрим лишь конечные множества. Пусть $f(x)$ — взаимно однозначное отображение множества $M = \{1, \ldots, n\}$ на себя. Его можно записать в виде матрицы

$$\sigma_f = \begin{pmatrix} 1 & 2 & \ldots & n \\ f(1) & f(2) & \ldots & f(n) \end{pmatrix}. \qquad (8.2.2)$$

Переставляя столбцы этой матрицы, мы получим другую матрицу, однако она будет соответствовать той же функции.

8.2.16. *Взаимно однозначное отображение $f(x)$ множества $\{1,\dots,n\}$ на себя называется* ***подстановкой n-й степени****. Такая подстановка может быть записана в виде*

$$\sigma_f = \begin{pmatrix} i_1 & i_2 & \dots & i_n \\ f(i_1) & f(i_2) & \dots & f(i_n) \end{pmatrix}, \quad \{i_1,\dots,i_n\} = \{1,\dots,n\}.$$

8.2.17. *Если $f(x)$ и $g(x)$ — две подстановки степени n, то функция $g(f(x))$ также является подстановкой той же степени, называется* ***произведением*** *подстановки f на подстановку g и обозначается через fg.*

Переходя к матричным записям, заметим, что если подстановка σ_f определена равенством (8.2.2) и

$$\sigma_g = \begin{pmatrix} f(1) & f(2) & \dots & f(n) \\ g(f(1)) & g(f(2)) & \dots & g(f(n)) \end{pmatrix},$$

то

$$\sigma_{fg} = \begin{pmatrix} 1 & 2 & \dots & n \\ g(f(1)) & g(f(2)) & \dots & g(f(n)) \end{pmatrix}.$$

Пример. Произведением перестановок

$$\sigma_f = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 3 & 1 \end{pmatrix}, \quad \sigma_g = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 2 & 1 \end{pmatrix} \tag{8.2.3}$$

является перестановка

$$\sigma_{fg} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 1 & 2 & 3 \end{pmatrix}. \qquad \square$$

8.2.18. *Умножение подстановок степени n ассоциативно и при $n \geq 3$ некоммутативно.*

Доказательство. Рассмотрим произведения

$$f(gh), \quad (fg)h$$

трех подстановок f, g и h множества $\{1,\dots,n\}$. Для каждого принадлежащего этому множеству элемента i из определения умножения подстановок следует справедливость равенств

$$(f(gh))(i) = (gh)(f(i)) = h(g(f(i))),$$

$$((fg)h)(i) = h((fg)(i)) = h(g(f(i))).$$

Это доказывает справедливость первого утверждения.

Чтобы доказать второе утверждение, достаточно привести пример двух подстановок, у которых перемена мест сомножителей влияет на произведение. Выше было вычислено произведение fg перестановок (8.2.3). Нетрудно проверить, что

$$\sigma_{gf} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 1 & 4 & 2 \end{pmatrix}$$

и потому $fg \neq gf$. □

8.2.19. *Множество всех подстановок степени n относительно умножения образует группу.*

Доказательство. Уже доказано, что умножение подстановок ассоциативно. Единицей при умножении служит тождественная подстановка

$$\sigma_e = \begin{pmatrix} 1 & 2 & \dots & n \\ 1 & 2 & \dots & n \end{pmatrix}.$$

Обратной к подстановке вида (8.2.2) является подстановка

$$\sigma_{f^{-1}} = \begin{pmatrix} f(1) & f(2) & \dots & f(n) \\ 1 & 2 & \dots & n \end{pmatrix},$$

поскольку $\sigma_f \sigma_{f^{-1}} = \sigma_{f^{-1}} \sigma_f = \sigma_e$. □

8.2.20. *Система, состоящая из всех подстановок n-й степени и операции умножения, называется* ***симметрической группой n-й степени****.*

8.2.21. *Подмножество H множества элементов группы G называется* ***подгруппой*** *этой группы, если относительно операции, определенной в G, оно само является группой. Обозначение: $H \leq G$.*

Пример. Множество всех четных чисел с операцией сложения является подгруппой группы всех целых чисел с той же операцией. □

8.2.22 (теорема Кэлли)**.** *Для любой конечной группы G найдется такое множество M, что G изоморфна некоторой подгруппе группы всех подстановок множества M.*

Доказательство. Перенумеруем элементы группы G, сопоставив каждому одно из чисел $1, \dots, n$, где n — порядок группы. Через g_i

обозначим элемент с номером i, а через g_{ji} — элемент группы G, представимый в виде произведения $g_j g_i$. Из $j \neq k$ следует, что $g_j g_i \neq g_k g_i$. Действительно, если $g_j g_i = g_k g_i$, то $g_j g_i g_i^{-1} = g_k g_i g_i^{-1}$ и потому $g_j = g_k$. Это означает, что множество $\{g_{1i}, \ldots, g_{ni}\}$ содержит все элементы группы G.

В качестве M возьмем множество $\{g_1, \ldots, g_n\}$. Для каждого $i \in \{1, \ldots, n\}$ сопоставим элементу g_i подстановку

$$\sigma^{g_i} = \begin{pmatrix} g_1 & g_2 & \ldots & g_n \\ g_{1i} & g_{2i} & \ldots & g_{ni} \end{pmatrix}.$$

Из сказанного ранее следует, что все элементы второй строки различны. Множество подстановок $\{\sigma^{g_1}, \ldots, \sigma^{g_n}\}$ обозначим через H.

Обозначим через φ отображение группы G на H, задаваемое формулами $\varphi(g_i) = \sigma^{g_i}$, $i = \overline{1,n}$. Докажем, что φ — изоморфизм. Отметим сначала, что разные элементы группы G отображаются на разные подстановки. Действительно, если $i \neq j$, то рассуждения, аналогичные проведенным ранее, показывают, что

$$g_{1i} = g_1 g_i \neq g_1 g_j = g_{1j}.$$

Это означает, что первые столбцы подстановок σ^{g_i}, σ^{g_j}, равные соответственно

$$\begin{pmatrix} g_1 \\ g_{1i} \end{pmatrix}, \quad \begin{pmatrix} g_1 \\ g_{1j} \end{pmatrix},$$

различны, и потому $\sigma^{g_i} \neq \sigma^{g_j}$.

Теперь докажем, что

$$\varphi(g_i g_j) = \varphi(g_i)\varphi(g_j) \tag{8.2.4}$$

для любых элементов g_i, g_j группы G. Пусть подстановка σ^{g_i} определена равенством

$$\sigma^{g_i} = \begin{pmatrix} g_1 & g_2 & \ldots & g_n \\ g_{1i} & g_{2i} & \ldots & g_{ni} \end{pmatrix},$$

а подстановка σ^{g_j} — равенствами

$$\sigma^{g_j} = \begin{pmatrix} g_1 & g_2 & \ldots & g_n \\ g_{1j} & g_{2j} & \ldots & g_{nj} \end{pmatrix} = \begin{pmatrix} g_1 & g_2 & \ldots & g_n \\ g_1 g_j & g_2 g_j & \ldots & g_n g_j \end{pmatrix}.$$

Так как множество $\{g_{1i}, g_{2i}, \ldots, g_{ni}\}$ и множество $\{g_1, g_2, \ldots, g_n\}$ всех элементов группы G совпадают, подстановку σ^{g_j} можно записать также в виде

$$\sigma^{g_j} = \begin{pmatrix} g_{1i} & g_{2i} & \ldots & g_{ni} \\ g_{1i} g_j & g_{2i} g_j & \ldots & g_{ni} g_j \end{pmatrix}.$$

Согласно 8.2.17 произведение $\sigma^{g_i}\sigma^{g_j}$ этих подстановок равно

$$\begin{pmatrix} g_1 & g_2 & \cdots & g_n \\ g_{1i}g_j & g_{2i}g_j & \cdots & g_{ni}g_j \end{pmatrix} = \begin{pmatrix} g_1 & g_2 & \cdots & g_n \\ g_1g_ig_j & g_2g_ig_j & \cdots & g_ng_ig_j \end{pmatrix}.$$

Видим, что оно совпадает с $\varphi(g_ig_j)$ и равенство (8.2.4) справедливо. □

8.3. КОЛЬЦА И ПОЛЯ

В этом разделе мы рассмотрим некоторые алгебры с двумя основными бинарными операциями.

8.3.1. *Кольцом* называется непустое множество R, на котором заданы две бинарные операции, обычно обозначаемые символами $+$ и $\cdot$ и называемые ***сложением*** и ***умножением***, удовлетворяющие следующим условиям:

R1) $\langle R;\ + \rangle$ — *абелева группа;*

R2) $\langle R;\ \cdot \rangle$ — *полугруппа;*

R3) *сложение и умножение связаны дистрибутивными законами:*

$$(\alpha+\beta)\cdot\gamma = \alpha\cdot\gamma + \beta\cdot\gamma, \quad \gamma\cdot(\alpha+\beta) = \gamma\cdot\alpha + \gamma\cdot\beta$$

для любых α, β, γ *из* R.

Замечание. Символ умножения в формулах часто не пишут, а лишь подразумевают.

Примеры. 1. Множество $\mathbb{Z}$ всех целых чисел вместе с операциями сложения и умножения образует кольцо.

2. Множество всех четных чисел вместе с операциями сложения и умножения образует кольцо.

3. Кольцо образует множество всех многочленов, имеющих действительные коэффициенты, относительно операций сложения и умножения многочленов.

4. Вводя обычные операции сложения и умножения действительных квадратных матриц одинакового размера, мы также получаем кольцо. □

Выполнение аксиомы R2 из определения 8.3.1 означает, что операция умножения ассоциативна. Ввиду этого системы, удовлетворяющие требованиям определения 8.3.1, часто называют **ассоциативными кольцами**. Удаляя аксиому R2, получим следующее определение.

8.3.2. *Система, состоящая из непустого множества* R *и двух бинарных операций* $+$ *и* $\cdot$*, называемых* ***сложением*** *и* ***умножением***, *называется*

***неассоциативным кольцом**, если она удовлетворяет следующим аксиомам:*

R1) $\langle R;\ + \rangle$ *— абелева группа;*

R2) *сложение и умножение связаны дистрибутивными законами:*

$$(\alpha+\beta)\cdot\gamma=\alpha\cdot\gamma+\beta\cdot\gamma,\quad \gamma\cdot(\alpha+\beta)=\gamma\cdot\alpha+\gamma\cdot\beta$$

для любых $\alpha,\beta,\gamma\in R$

Как известно, если произведение двух действительных чисел равно нулю, то равен нулю хотя бы один из сомножителей. Для некоторых колец это правило оказывается неверным.

Пример. Рассмотрим кольцо, образованное действительными квадратными матрицами второго порядка с обычными операциями сложения и умножения. Принадлежащие ему матрицы

$$\begin{pmatrix}0&0\\0&0\end{pmatrix},\quad \begin{pmatrix}1&0\\0&1\end{pmatrix}$$

играют роль нуля и единицы. Легко проверить, что

$$\begin{pmatrix}1&2\\2&4\end{pmatrix}\begin{pmatrix}-2&4\\1&-2\end{pmatrix}=\begin{pmatrix}0&0\\0&0\end{pmatrix}.$$ □

8.3.3. *Если* a *и* b *— отличные от нуля элементы некоторого кольца и* $ab=0$*, то* a *называется **левым делителем нуля**, а* b *— **правым делителем нуля**.*

В коммутативном кольце любой левый делитель нуля является и правым делителем и, наоборот, правый делитель нуля является левым делителем, поэтому говорят просто о **делителях нуля**.

8.3.4. *Если в кольце* $\mathbf{R}$ *равенство* $ab=0$ *справедливо только тогда, когда один из сомножителей равен нулю, то* $\mathbf{R}$ *называется **кольцом без делителей нуля**. Коммутативное кольцо без делителей нуля с отличной от нуля единицей называется **целостным** или **областью целостности**.*

8.3.5. *Система, образованная множеством* F*, содержащим более одного элемента, и двумя бинарными операциями, обычно обозначаемыми символами* $+$ *и* $\cdot$ *и называемыми **сложением** и **умножением**, называется **полем**, если она удовлетворяет следующим условиям:*

F1) $\langle F;\ + \rangle$ *— абелева группа;*

F2) *если* 0 *— нейтральный элемент относительно сложения, то* $\langle F\setminus\{0\};\ \cdot\ \rangle$ *— абелева группа;*

F3) *сложение и умножение связаны дистрибутивными законами:*

$$(\alpha+\beta)\cdot\gamma=\alpha\cdot\gamma+\beta\cdot\gamma,\quad \gamma\cdot(\alpha+\beta)=\gamma\cdot\alpha+\gamma\cdot\beta$$

для любых α,β,γ *из* F.

Очевидно, любое поле является кольцом. Для того чтобы ассоциативное кольцо было полем, необходимо, чтобы относительно умножения ненулевые элементы кольца образовывали абелеву группу. Из определений 8.2.1, 8.2.3 следует, что достаточно проверить, есть ли в кольце единица и верно ли, что для каждого отличного от нуля элемента есть к нему обратный.

Примеры. 1. Из определения 8.3.5 следует, что множество действительных чисел $\mathbb{R}$ в совокупности с обычными операциями сложения и умножения является полем.

2. Множество $\mathbb{Q}$ рациональных чисел с теми же операциями сложения и умножения также образует поле.

3. Операции сложения и умножения комплексных чисел удовлетворяют всем аксиомам из определения 8.3.5, поэтому система $\langle\mathbb{C};\ +,\cdot\rangle$, где $\mathbb{C}$ — множество всех комплексных чисел, является полем. □

Системы $\langle F;\ +\rangle$ и $\langle F\setminus\{0\};\ \cdot\rangle$ называются **аддитивной** и **мультипликативной группами поля**. В первой из них есть нуль, а во второй — единица. Эти элементы (единственные, как было доказано ранее) называются **нулем** и **единицей** поля.

8.3.6. *Множество* F_1*, элементы которого принадлежат полю* F*, называется* ***подполем поля*** F*, если оно само является полем относительно операций, определенных в* F.

8.3.7. *Если* F_1 *является подполем поля* F*, то* F *называется* ***расширением подполя*** F_1.

Пример. Множество $\mathbb{Q}$ рациональных чисел является подполем поля $\mathbb{R}$ действительных чисел, а поле $\mathbb{R}$ — расширением поля $\mathbb{Q}$. □

Система $\mathbf{G}_k=\langle Z_k;\ +\rangle$, состоящая из классов вычетов по модулю k и введенной в 7.2.8 операции сложения, представляет собой конечную аддитивную абелеву группу. Поскольку множество Z_k конечное, его элементы обычно обозначают через $0,\ldots,k-1$.

8.3.8. *Система* $\mathbf{Z}_k=\langle Z_k;\ +,\ \cdot\rangle$*, состоящая из классов вычетов по модулю* k *и введенных выше операций сложения и умножения, является кольцом, называемым* ***кольцом классов вычетов по модулю*** k. □

Таблица 8.1

$+$	0	1	2	3	$\cdot$	0	1	2	3
0	0	1	2	3	0	0	0	0	0
1	1	2	3	0	1	0	1	2	3
2	2	3	0	1	2	0	2	0	2
3	3	0	1	2	3	0	3	2	1

Пример. В кольце $\mathbf{Z}_4$ операции сложения и умножения задаются табл. 8.1. □

8.3.9. *Кольцо классов вычетов $\mathbf{Z}_k$ является полем тогда и только тогда, когда k — простое число.*

Доказательство. «⇒» Если число k не простое, то найдутся такие числа $p > 1$ и $q > 1$, что $k = pq$. Переходя к классам вычетов, получаем равенство $[p][q] = [0]$, означающее, что в кольце $\mathbf{Z}_k$ имеются делители нуля, следовательно, оно не может быть полем.

«⇐» Пусть теперь число k простое. Так как $[p][1] = [p]$ для любого $p \in \mathbb{Z}$, класс $[1]$ играет в кольце $\mathbf{Z}_k$ роль единицы. Убедимся в том, что для любого элемента $[q]$, не делящегося на k, в этом кольце есть к нему обратный.

Так как число k простое, числа

$$1q, \quad 2q, \quad \ldots, \quad (k-1)q \tag{8.3.1}$$

не делятся на k, и потому классы

$$[1q], \quad [2q], \quad \ldots, \quad [(k-1)q] \tag{8.3.2}$$

отличны от класса $[0]$. Эти классы попарно различны. Действительно, пусть

$$0 < i < j < k, \quad [iq] = [jq],$$

тогда

$$[jq] - [iq] = [0].$$

Отсюда следует, что принадлежащее последовательности (8.3.1) число $(j-i)q$ делится на k, что неверно.

Поскольку в последовательности (8.3.2) все классы попарно различны, она содержит все элементы кольца $\mathbf{Z}_k$, и в частности элемент $[1]$. Это означает, что $[iq] = [1]$ для некоторого $i < k$. Поскольку $[iq] = [i][q]$, элемент $[i]$ является обратным к $[q]$. □

8.4. ИТЕРАТИВНЫЕ АЛГЕБРЫ. КЛОНЫ

Обозначим через E_k множество чисел $\{0, 1, \ldots, k-1\}$, а через P_k — множество всех функций, определенных на E_k, значения которых также принадлежат E_k. Пусть функции $f, g \in P_k$ зависят соответственно от n и m переменных. Из этих функций можно получать новые функции, принадлежащие P_k, при помощи следующих операций:

$$\begin{aligned}
(\zeta f)(x_1, \ldots, x_n) &= f(x_2, x_3, \ldots, x_n, x_1),\\
(\tau f)(x_1, \ldots, x_n) &= f(x_2, x_1, x_3, \ldots, x_n),\\
(\Delta f)(x_1, \ldots, x_{n-1}) &= f(x_1, x_1, x_2, \ldots, x_{n-1}),\\
(\nabla f)(x_1, \ldots, x_n) &= f(x_2, x_3, \ldots, x_{n+1}),\\
(f * g)(x_1, \ldots, x_{n+m-1}) &= \\
&= f(g(x_1, \ldots, x_m), x_{m+1}, \ldots, x_{m+n-1}).
\end{aligned}$$

Если функция f унарная, то $\zeta f = \tau f = \Delta f = f$.

Каждая из операций ζ, τ, Δ, ∇ меняет индексы аргументов функций. При применении операции ζ к функции $f(x_1, \ldots, x_n)$ индекс n превращается в 1, а к остальным индексам прибавляется единица. При использовании операции τ индекс 1 меняется на 2, а индекс 2 заменяется единицей. Остальные индексы не меняются. Операция ∇ прибавляет единицу к индексу каждой переменной. Операция Δ оставляет неизменным индекс 1, а от остальных индексов отнимается единица. При применении операции $*$ к функциям $f(x_1, \ldots, x_n)$ и $g(x_1, \ldots, x_m)$ функция $g(x_1, \ldots, x_m)$ подставляется в f вместо переменной x_1, а к индексам всех остальных переменных функции f прибавляется число $m - 1$.

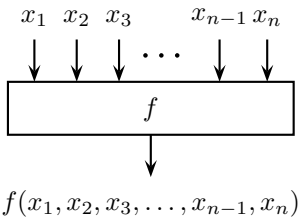


Рис. 8.1
Элемент $f(x_1, \ldots, x_n)$

На выбор введенных операций над функциями значительное влияние оказала простота и естественность их трактовки в некоторых

приложениях. Предположим, что имеется некоторое устройство с n входами и одним выходом, которое при подаче на первый, второй, ..., последний вход соответственно букв $x_1, x_2, \ldots, x_n$ выдает на выходе букву $f(x_1, \ldots, x_n)$. Назовем это устройство **функциональным элементом**, реализующим функцию $f(x_1, \ldots, x_n)$, или, для краткости, **элементом** $f(x_1, \ldots, x_n)$. Изобразим его на рис. 8.1.

Операции ζ, τ, Δ, ∇ позволяют получать из этого функционального элемента новые элементы путем коммутации входов (рис. 8.2–8.5).

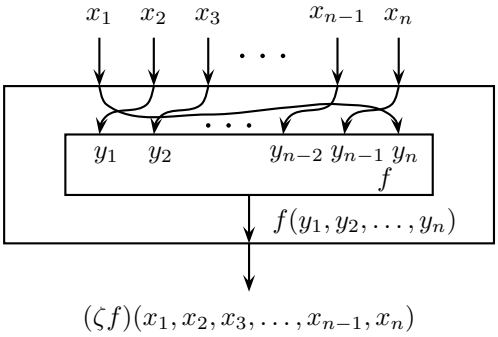


Рис. 8.2
Элемент ζf

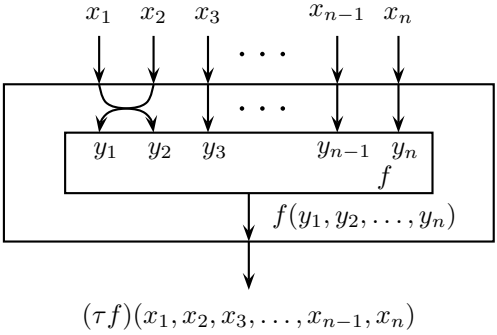


Рис. 8.3
Элемент τf

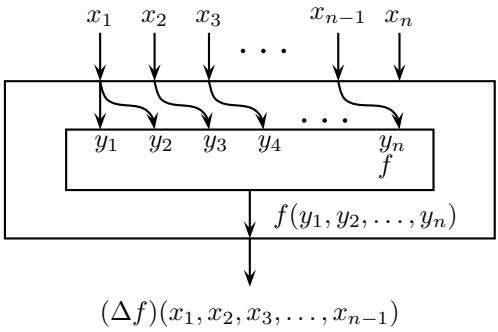


Рис. 8.4
Элемент Δf

Соединяя выход элемента $g(x_1, \ldots, x_m)$ с первым входом элемента $f(x_1, \ldots, x_n)$ и подавая на остальные входы элемента f значения переменных $x_{m+1}, \ldots, x_{m+n-1}$, получаем новый элемент $f * g$ (рис. 8.6).

Введем следующее обозначение:

$$\zeta^k f := \underbrace{\zeta \ldots \zeta}_{k \text{ раз}} f.$$

Аналогичный смысл будут иметь обозначения τ^k, Δ^k, ∇^k.

8.4.1. *Аргументы каждой функции $f(x_1, \ldots, x_n)$ можно переставить в любом порядке при помощи операций ζ и τ.*

Доказательство. Сначала поменяем местами переменные x_i и x_{i+1}:

$$(\zeta^{n-i} f)(x_1, \ldots, x_n) = f(\underbrace{x_{n-(i-1)}, \ldots, x_n}_{i-1 \text{ мест}}, x_1, x_2, \ldots, x_{n-i}),$$
$$(\tau\zeta^{i-1} f)(x_1, \ldots, x_n) = f(x_{n-(i-1)}, \ldots, x_n, x_2, x_1, \ldots, x_{n-i}),$$
$$(\zeta^{i-1}\tau\zeta^{n-i} f)(x_1, \ldots, x_n) = f(x_1, \ldots, x_{i-1}, x_{i+1}, x_i, x_{i+2}, \ldots, x_n).$$

Поменять местами аргументы x_i и x_j можно следующим образом. Предположим, что x_j находится справа от x_i. Сначала последовательно меняем местами x_j с соседними переменными, находящимися слева, до тех пор, пока аргументы x_i и x_j не окажутся рядом, причем x_j должен находиться слева от x_i. Затем последовательно меняем местами x_i с соседними переменными справа до тех пор, пока он не окажется на том месте, где сначала был x_j.

Очевидно, имея возможность менять местами любые два аргумента функции, можно переставить аргументы этой функции в любом порядке. □

Пример. Операциями ζ и τ превратим функцию $f(x_1, x_2, x_3, x_4)$ в функцию $f(x_3, x_1, x_4, x_2)$:

$$f_1(x_1, x_2, x_3, x_4) = (\zeta^3 f)(x_1, x_2, x_3, x_4) = f(x_4, x_1, x_2, x_3),$$
$$f_2(x_1, x_2, x_3, x_4) = (\tau f_1)(x_1, x_2, x_3, x_4) = f(x_4, x_2, x_1, x_3),$$
$$f_3(x_1, x_2, x_3, x_4) = (\zeta^3 f_2)(x_1, x_2, x_3, x_4) = f(x_3, x_1, x_4, x_2). \quad \square$$

Операции ζ, τ позволяют осуществить любую перестановку аргументов функции, операции Δ, ∇ уменьшают и увеличивают число аргументов функции.

Пример. Преобразуем функцию $f(x_1, x_2, x_3, x_4)$ в $f(x_1, x_2, x_1, x_2)$:

$$f_1(x_1, x_2, x_3, x_4) = \tau f(x_1, x_2, x_3, x_4) = f(x_2, x_1, x_3, x_4),$$
$$f_2(x_1, x_2, x_3, x_4) = \zeta^3 f_1(x_1, x_2, x_3, x_4) = f(x_1, x_4, x_2, x_3),$$
$$f_3(x_1, x_2, x_3, x_4) = \Delta f_2(x_1, x_2, x_3, x_4) = f(x_1, x_3, x_1, x_2),$$
$$f_4(x_1, x_2, x_3, x_4) = \zeta^2 f_3(x_1, x_2, x_3, x_4) = f(x_3, x_2, x_3, x_1),$$
$$f_5(x_1, x_2, x_3, x_4) = \Delta f_4(x_1, x_2, x_3, x_4) = f(x_2, x_1, x_2, x_1),$$
$$f_6(x_1, x_2, x_3, x_4) = \zeta f_5(x_1, x_2, x_3, x_4) = f(x_1, x_2, x_1, x_2). \quad \square$$

8.4.2. *Множество P_k с определенными на нем операциями ζ, τ, $\Delta, \nabla, *$ называется k-**значной логикой** или **итеративной алгеброй Поста ранга** k. Обозначение: $\mathcal{P}_k$.*

Латинское слово *iteratio* означает *повторение*. В математике при использовании серии в чем-то схожих операций **итерацией** называют результат применения отдельной операции.

8.4.3. *Подмножество A функций из P_k называется **замкнутым**, если при применении операций ζ, τ, Δ, ∇, $*$ к функциям из A всегда получаются функции, также принадлежащие A. Замкнутые подмножества множества P_k называются **подалгебрами** (алгебры) k-**значной логики** или **итеративными алгебрами**.*

Часто подалгебры k-значной логики просто называют **замкнутыми классами функций**.

Пример. При $k \geqslant 3$ множество всех функций из P_k, принимающих не более $k-1$ значений, образует замкнутый класс. Действительно,

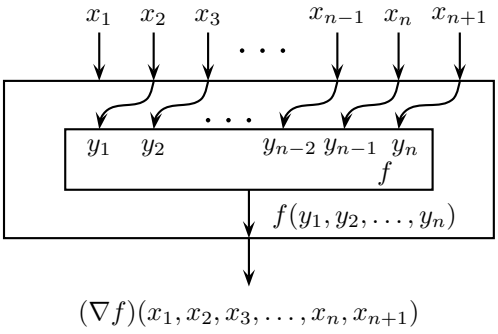


Рис. 8.5
Элемент ∇f

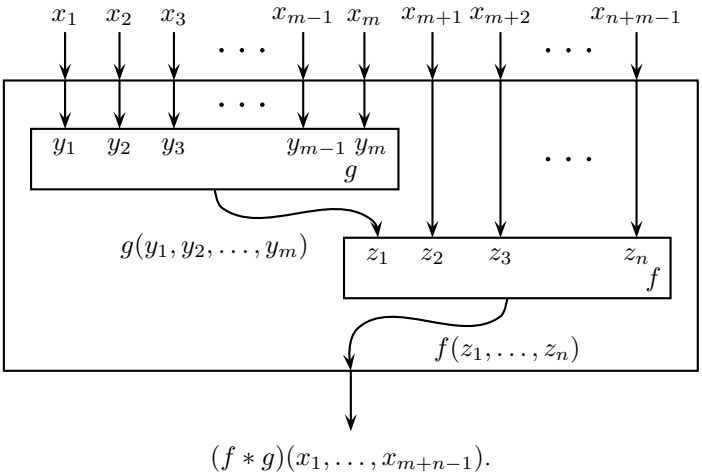


Рис. 8.6
Элемент $f * g$

операции ζ, τ, Δ, ∇ влияют только на число и порядок аргументов функции и не могут увеличить количество принимаемых значений. Функция, полученная из f и g операцией $*$, имеет вид

$$f(g(x_1, \ldots, x_m), x_{m+1}, \ldots, x_{m+n-1}).$$

Число ее значений не может превышать числа значений функции f. $\square$

8.4.4. *Множество P_k с определенными на нем операциями ζ, τ, Δ, $*$, называется* ***предитеративной алгеброй Поста ранга*** *k. Принадлежащие этой алгебре замкнутые классы функций называются* ***предитеративными алгебрами****. Предитеративные алгебры, содержащие все функции вида*

$$e_i^n(x_1, \ldots, x_n) = x_i,$$

называются ***клонами****.*

Слово «клон» является сокращением английского предложения *closed set of operation*.

Многие авторы совокупность операций ζ, τ, Δ, ∇, $*$ объединяют в одно понятие **суперпозиции**. Поскольку результат применения суперпозиции к конкретным функциям не является однозначно определенным, суперпозиция не является алгебраической операцией. Это и явилось причиной замены ее пятеркой указанных выше операций. С другой стороны, даже при небольшом количестве подстановок функций $g_1, \ldots, g_n$ в функцию f формула, выражающая результат таких подстановок через операции ζ, τ, Δ, ∇, $*$, является весьма громоздкой. Это подтверждают приведенные выше примеры, в которых требовалось всего лишь изменить аргументы функции. Однако в подавляющем большинстве случаев находить такие формулы не требуется. Например, обычно достаточно знать, что функцию

$$h(x, y, z) = f(x, g(x, f(y, g(x, y, z), x), z))$$

можно получить при помощи операций ζ, τ, Δ, ∇, $*$ из функций $f(x, y, z)$ и $g(x, y, z)$.

ЛИТЕРАТУРА

1. Андерсон Дж. А. *Дискретная математика и комбинаторика.* М.: Издательский дом «Вильямс», 2004.
2. Ахо А., Ульман Дж. *Теория синтаксического анализа, перевода и компиляции. Т. 1. Синтаксический анализ.* М.: Мир, 1978.
3. Бернштейн Т. В., Макаров Р. В., Храмова Т. В. *Дискретная математика. Учебное пособие.* Новосибирск: СибГУТИ, 2004.
4. Верещагин Н. К., Шень А. *Языки и исчисления.* М.: МЦНМО, 2000.
5. Виноградов И. М. *Основы теории чисел.* СПб.: Лань, 2009.
6. Гиндикин С. Г. *Алгебра логики в задачах.* М.: Наука, 1972.
7. Гладкий А. В. *Математическая логика.* М.: Российск. гос. гуманит. ун-т, 1998.
8. Гладкий А. В. *Формальные грамматики и языки.* М.: Наука, 1973.
9. Донской В. И. *Дискретная математика.* Симферополь: СОНАТ, 2000.
10. Евстигнеев В. А., Касьянов В. Н. *Толковый словарь по теории графов в информатике и программировании.* Новосибирск: Наука. Сиб. предприятие РАН, 1999.
11. Евстигнеев В. А., Мельников Л. С. *Задачи и упражнения по теории графов и комбинаторике.* Новосибирск: Новосиб. ун-т, 1981.
12. Ежов И. И., Скороход А. В., Ядренко М. И. *Элементы комбинаторики.* М.: Наука, 1977.
13. Емеличев В. А., Мельников О. И., Сарванов В. И., Тышкевич Р. И. *Лекции по теории графов.* М.: Наука, 1990.
14. Ершов Ю. Л., Палютин Е. А. *Математическая логика.* М.: Наука, 1979.
15. Ивлев Ю. В. *Логика: Сборник упражнений: Учеб. пособие.* М.: Дело, 2002.
16. Кострикин А. И. *Введение в алгебру.* М.: Наука, 1977.
17. Лавров И. А. *Логика и алгоритмы.* Новосибирск: Новосиб. гос. ун-т, 1970.
18. Лавров И. А., Максимова Л. Л. *Задачи по теории множеств, математической логике и теории алгоритмов.* М.: Наука, 1975.

19. Лихтарников Л. М., Сукачева Т. Г. *Математическая логика.* СПб.: Лань, 2009.
20. Логинов Б. М. *Введение в дискретную математику.* Калуга, 1998.
21. Ляпин Е. С., Евсеев А. Е. *Алгебра и теория чисел.* М.: Просвещение, 1974. Т. 1.
22. Макаров Р. Н. *Практические занятия по дискретной математике. Учебное пособие.* Новосибирск: СибГУТИ, 2001.
23. Мальцев А. И. *Алгебраические системы.* М.: Наука, 1969.
24. Мальцев А. И. *Алгоритмы и рекурсивные функции.* М.: Наука, 1965.
25. Мальцев А. И. *Итеративные алгебры Поста.* Новосибирск: Новосиб. ун-т, 1976.
26. Марченков С. С. *Замкнутые классы булевых функций.* М.: Физматлит, 2000.
27. *Математическая энциклопедия.* М.: Советская энциклопедия, 1977–1985. Тт. 1–5.
28. Непейвода Н. Н. *Прикладная логика: Учеб. пособие.* Новосибирск: Изд-во Новосиб. гос. ун-та, 2000.
29. Нефедов В. Н., Осипова В. А. *Курс дискретной математики.* М.: Изд-во МАИ, 1992.
30. *Современный словарь иностранных слов.* М.: Русский язык, 1993.
31. Судоплатов С. В., Овчинникова Е. В. *Элементы дискретной математики: Учебник.* М.: ИНФРА-М, Новосибирск: Изд-во НГТУ, 2002.
32. Хомич В. И. *Логика высказываний и исчисление высказываний: Учеб. пособие по курсу «Математическая логика и теория алгоритмов»* М.: Изд-во МГТУ им. Н. Э. Баумана, 2004.
33. Яблонский С. В. *Введение в дискретную математику.* М.: Наука, 1986.

Предметный указатель

Г

Д

П

P

C

СПИСОК ОБОЗНАЧЕНИЙ

a^{-n}	—	степень элемента группы
a^0	—	степень элемента группы
a^n	—	степень элемента группы
$c_0^1(x)$	—	операция
$c_1^1(x)$	—	операция
$e_1^1(x)$	—	операция
$G_1 \square G_2$	—	декартово произведение графов
MP	—	modus ponens
Z_k	—	все классы вычетов по модулю k
$[i]$	—	класс вычетов, дающих при делении на модуль в остатке i
$\&$	—	связка
$\bar{x}$	—	связка
$\cap$	—	пересечение множеств
$\cup$	—	объединение множеств
$\deg_+ v$	—	полустепень исхода вершины v
$\deg_- v$	—	полустепень захода вершины v
$\mathrm{indeg}(v)$	—	полустепень захода вершины v
$\vee$	—	связка
$\mathbb{Q}$	—	множество рациональных чисел
$\mathbb{R}$	—	множество действительных чисел
$\mid$	—	связка
$\oplus$	—	связка
$\mathrm{outdeg}(v)$	—	полустепень исхода вершины v
$\overline{A}$	—	дополнение множества A
$\setminus$	—	разность множеств
$\sim$	—	связка
$\supset$	—	связка
$\triangle$	—	симметрическая разность множеств
$\mathbf{Z}_k$	—	кольцо классов вычетов по модулю k

ОГЛАВЛЕНИЕ

Иван Анатольевич МАЛЬЦЕВ

ДИСКРЕТНАЯ МАТЕМАТИКА

УЧЕБНОЕ ПОСОБИЕ

Издание четвертое, стереотипное

Зав. редакцией литературы по информационным технологиям и системам связи *О. Е. Гайнутдинова*

ЛР № 065466 от 21.10.97
Гигиенический сертификат 78.01.10.953.П.1028 от 14.04.2016 г., выдан ЦГСЭН в СПб

Издательство «ЛАНЬ»
lan@lanbook.ru; www.lanbook.com
196105, Санкт-Петербург, пр. Ю. Гагарина, д. 1, лит. А.
Тел./факс: (812) 336-25-09, 412-92-72.
Бесплатный звонок по России: 8-800-700-40-71

Подписано в печать 23.09.22.
Бумага офсетная. Гарнитура Литературная. Формат 60×90 $^{1}/_{16}$.
Печать офсетная/цифровая. Усл. п. л. 18,25. Тираж 30 экз.

Заказ № 1330-22.

Отпечатано в полном соответствии
с качеством предоставленного оригинал-макета
в АО «Т8 Издательские Технологии».
109316, г. Москва, Волгоградский пр., д. 42, к. 5.