Ю. И. ГАЛУШКИНА, А. Н. МАРЬЯМОВ

КОНСПЕКТ ЛЕКЦИЙ ПО ДИСКРЕТНОЙ МАТЕМАТИКЕ

С упражнениями и контрольными работами

Ю. И. ГАЛУШКИНА, А. Н. МАРЬЯМОВ

КОНСПЕКТ ЛЕКЦИЙ ПО ДИСКРЕТНОЙ МАТЕМАТИКЕ

С упражнениями и контрольными работами

МОСКВА

АЙРИС ПРЕСС

2007

УДК 519.1(075.8)
ББК 22.176я73-2
Г16

Серийное оформление *А. М. Драгового*

Галушкина, Ю. И.

Г16 Конспект лекций по дискретной математике / Ю. И. Галушкина, А. Н. Марьямов. — М.: Айрис-пресс, 2007. — 176 с. — (Высшее образование).

ISBN 978-5-8112-2599-6

В книге в доступной форме изложены разделы, традиционно изучаемые в курсе дискретной математики. Книга рассчитана на студентов нематематических вузов, желающих ознакомиться с методами дискретной математики. Математическая подготовка, необходимая для чтения этой книги, ограничивается программой математики средней школы.

Содержание разделов книги взаимно связано друг с другом и включает: элементы математической логики, теории множеств, предикатов, графов, элементы комбинаторики, кодирования и теории конечных автоматов, а также введение в теории алгоритмов.

Все разделы снабжены большим количеством примеров и решенных задач, помогающих усвоить и закрепить изучаемый материал.

Книга может быть также полезна преподавателям, которые начинают читать курс дискретной математики.

ББК 22.176я73-2
УДК 519.1(075.8)

ISBN 978-5-8112-2599-6

Содержание

9-10

11-12

13-15

Предисловие

Данное учебное пособие ставит своей целью ознакомить учащихся с проблемами *дискретной математики*, которая называется так, потому что в ней нет понятия бесконечного множества, предельного перехода, непрерывности, дифференцируемости и т. д. В отличие от классической математики, которая занимается изучением непрерывных бесконечных структур, дискретная математика представляет собой область математики, в которой изучаются свойства структур конечного характера.

Основные блоки дискретной математики можно изобразить в виде следующей структурной схемы.

Эта схема представляет разделы, традиционно изучаемые в рамках дисциплины «Основы дискретной математики» и иллюстрирует целостность изучаемой науки. В конце каждого раздела приведены упражнения и задачи, помогающие усвоить и закрепить изучаемый материал.

Часть 1

Элементы математической логики

Лекции 1–4

Математическая логика — это анализ методом рассуждений, при этом в первую очередь исследуются формы рассуждений, а не их содержание, т. е. математическая логика исследует соотношения между основными понятиями математики, на базе которых доказываются математические утверждения. Простейшую из формальных логических теорий называют *алгеброй высказываний*, поэтому начнем знакомство с элементами математической логики с такого понятия, как *высказывание*, которое лежит в основе логико-математической теории дискретной математики.

1. Составные высказывания

Высказыванием называется повествовательное предложение, о котором в данной ситуации можно сказать, что оно истинно или ложно, но не то и другое одновременно.

Приведем примеры высказываний.

Пример 1. Волга впадает в Каспийское море.

Пример 2. Два больше трех.

Первое высказывание является истинным, а второе — ложным.

Таким образом, высказывание обладает свойством представлять истину или ложь, поэтому на высказывание можно смотреть как на величину, которая может принимать только одно из двух значений: «истина», «ложь».

Поставим в соответствие высказыванию логическую переменную x, которая принимает значение 1, если высказывание истинно, и 0, если высказывание ложно.

Мы не будем исследовать внутреннюю структуру высказываний, потому что такое исследование оказывается достаточно трудным и относится скорее к лингвистике, чем к математике. Поэтому мы будем поступать так, как если бы мы знали все о простых высказываниях, и будем изучать лишь их сочетания, т. е. как различными способами из отдельных высказываний можно построить новое высказывание.

Это новое высказывание называется *составным*, в то время как высказывания, из которых оно образовано, называются его простыми составляющими или компонентами. Любое высказывание, даже такое, которое на самом деле является сложным, может быть использовано в качестве одного из простых составляющих какого-то другого составного высказывания.

2. Простейшие связки

Значение истинности составного высказывания определяется значениями истинности его компонент.

Высказывания будем обозначать прописными буквами латинского алфавита X, Y, Z ... .

Составные высказывания будем получать из простых с помощью логических операций: *отрицание, конъюнкция, дизъюнкция, импликация, эквивалентность,* которые осуществляются при помощи логических связок: $\overline{}$; $\wedge$; $\vee$; $\rightarrow$; $\leftrightarrow$.

Название	Прочтение	Обозначение
Отрицание	не	$-$
Конъюнкция	и	$\wedge$
Дизъюнкция	или	$\vee$
Импликация	если...то	$\rightarrow$
Эквивалентность	тогда и только тогда, когда	$\leftrightarrow$

При рассмотрении той или иной связки мы хотим знать, каким именно образом истинность составного высказывания, порожденного этой связкой, зависит от истинности его компонент. Очень удобно изображать эту зависимость, пользуясь таблицами истинности, которые называются также интерпретациями логических операций. Каждой строке таблицы истинности взаимно однозначно соответствует набор составляющих высказываний и соответствующее значение составного высказывания. Наборы из нулей и единиц, соответствующих составляющим высказываниям, в каждой строке таблицы истинности имеют стандартное расположение, т. е. расположены в лексикографическом порядке (порядке возрастания).

Пусть даны два произвольных высказывания X и Y.

Отрицанием высказывания X называется высказывание $\overline{X}$, которое истинно, когда X ложно, и ложно, когда X истинно.

Таблица истинности для отрицания.

X	$\overline{X}$
0	1
1	0

Конъюнкцией двух высказываний X и Y называется высказывание $X \wedge Y$, которое истинно только в том случае, когда X и Y оба истинны.

Таблица истинности для конъюнкций.

X	Y	$X \wedge Y$
0	0	0
0	1	0
1	0	0
1	1	1

Дизъюнкцией двух высказываний X и Y называется высказывание $X \vee Y$, которое истинно, когда хотя бы одно из них истинно.

Таблица истинности дизъюнкций.

X	Y	$X \vee Y$
0	0	0
0	1	1
1	0	1
1	1	1

Импликацией двух высказываний X и Y называется высказывание $X \to Y$, которое ложно тогда и только тогда, когда X истинно, а Y ложно.

Таблица истинности для импликации.

X	Y	$X \to Y$
0	0	1
0	1	1
1	0	0
1	1	1

Эквивалентностью высказываний X и Y называется высказывание $X \leftrightarrow Y$, которое истинно тогда и только тогда, когда X и Y оба истинны или ложны.

Таблица истинности для эквивалентности.

X	Y	$X \leftrightarrow Y$
0	0	1
0	1	0
1	0	0
1	1	1

Для образования составных высказываний наряду с единичным использованием каждой основной связки можно пользоваться основными связками многократно, получая более сложные составные высказывания — аналогично тому, как с помощью основных арифметических операций образуются сложные алгебраические выражения.

Например, составными будут высказывания:

$$\overline{(X \wedge Y)}; \quad X \wedge \overline{X}; \quad (X \vee Y) \vee \overline{X}.$$

Их следует читать «изнутри наружу», подобно алгебраическим выражениям, в которых сначала группируются величины, заключенные в самые внутренние скобки, затем эти скобки в свою очередь группируются и т. д. Если скобок нет, то операции надо выполнять в следующем порядке: конъюнкция, дизъюнкция, импликация, эквивалентность, отрицание. Каждое составное высказывание имеет свою таблицу истинности, которая может быть построена стандартным образом.

3. Другие связки

Новые высказывания могут быть образованы при помощи нескольких логических операций и составлять формулы, некоторые из которых рассматриваются как логические операции, осуществляемые при помощи других логических связок: $|$; $\downarrow$; $\oplus$.

Название	**Прочтение**	**Обозначение**
Штрих Шеффера	Антиконъюнкция	$\vert$
Стрелка Пирса	Антидизъюнкция	$\downarrow$
Сумма по модулю два	Антиэквивалентность	$\oplus$

Штрих Шеффера, $X \mid Y$ или антиконъюнкция, по определению $(X \mid Y) = \overline{X \wedge Y}$.

Таблица истинности штриха Шеффера.

X	Y	$X \mid Y$
0	0	1
0	1	1
1	0	1
1	1	0

Стрелка Пирса, или антидизъюнкция, по определению $X \downarrow Y = \overline{X \vee Y}$.

Таблица истинности стрелки Пирса.

X	Y	$X \downarrow Y$
0	0	1
0	1	0
1	0	0
1	1	0

Сумма по модулю два, или антиэквивалентность, по определению $X \oplus Y = \overline{X \leftrightarrow Y}$.

Таблица истинности суммы по модулю два.

X	Y	$X \oplus Y$
0	0	0
0	1	1
1	0	1
1	1	0

Заметим, что таблицы истинности логических операций содержат 2^n строк, где n — число простых высказываний.

4. Логические отношения

Иногда бывает желательно рассмотреть взаимоотношение двух высказываний. Наиболее интересное из таких отношений имеет место, когда из одного высказывания логически следует другое. Если из X следует Y, мы говорим также, что Y является следствием X или что Y логически выводимо из X. Исходя из анализа логических возможностей для пары высказываний X и Y, отношение следствия можно охарактеризовать таким образом: из X следует Y, если Y истинно всякий раз, когда истинно X, т. е. если Y истинно во всех логически возможных случаях, в которых X истинно.

В случаях составных высказываний, имеющих одни и те же компоненты, таблицы истинности дают удобный метод для проверки того, имеет ли место отношение следствия.

Следующая таблица иллюстрирует этот метод:

X	Y	$X \leftrightarrow Y$	$X \to Y$	$X \vee Y$
0	0	1	1	0
0	1	0	1	1
1	0	0	0	1
1	1	1	1	1

Высказывание $X \leftrightarrow Y$ истинно в первом и четвертом случаях и в обоих этих случаях истинно также высказывание $X \to Y$. Мы видим, что из $X \leftrightarrow Y$ следует высказывание $X \to Y$. Сравнение двух последних столбцов показывает, что из высказывания $X \to Y$ не следует $X \vee Y$, из $X \vee Y$ не следует $X \to Y$.

При помощи таблиц истинности удобно осуществлять проверку эквивалентности двух составных высказываний, имеющих одни и те же компоненты. Для этого достаточно лишь посмотреть, одинаковы ли таблицы истинности у этих составных высказываний.

Из следующей таблицы истинности видно, что $X \to Y$ эквивалентно $\overline{X} \vee Y$.

X	Y	$X \to Y$	$\overline{X} \vee Y$
0	0	1	1
0	1	1	1
1	0	0	0
1	1	1	1

Два высказывания называются логически несовместимыми, если из истинности одного из них необходимо следует ложность другого. Другими словами, несовместимость высказываний X и Y означает, что они никогда не могут оказаться одновременно истинными. Если несколько составных высказываний построены из одних и тех же простых составляющих, то для проверки их совместимости нужно построить таблицы истинности для каждого из высказываний. Если среди всех строк найдется, по крайней мере, одна, в которой все составные высказывания истинны, то составные высказывания совместимы. В противном случае они оказываются несовместимыми.

5. Варианты импликации

Импликация двух высказываний отличается от эквивалентности, а также от дизъюнкции и конъюнкции тем, что она несимметрична. Так $X \vee Y$ эквивалентно $Y \vee X$; $X \wedge Y$ эквивалентно $Y \wedge X$; $X \leftrightarrow Y$ эквивалентно $Y \leftrightarrow X$, но $X \to Y$ не эквивалентно $Y \to X$. Высказывание $Y \to X$ называется **конверсией** высказывания $X \to Y$. Многие из наиболее распространенных ошибок в рассуждениях происходят от смешивания какого-либо высказывания с его конверсией. Интересно поэтому рассмотреть те импликации, которые могут быть образованы из высказываний X и Y.

В таблице истинности представлены четыре импликации и их названия.

X	Y	$\overline{X}$	$\overline{Y}$	Импликация $X \to Y$	Конверсия импликации $Y \to X$	Конверсия контрапозиции $\overline{X} \to \overline{Y}$	Контрапозиция $\overline{Y} \to \overline{X}$
0	0	1	1	1	1	1	1
0	1	1	0	1	0	0	1
1	0	0	1	0	1	1	0
1	1	0	0	1	1	1	1

Из таблицы видно, что $X \to Y$ эквивалентно $\overline{Y} \to \overline{X}$. Последнее называется контрапозицией первого. Контрапозиция является удобной формой импликации во многих рассуждениях. Аналогично, высказывание $\overline{X} \to \overline{Y}$ представляет собой конверсию контрапозиции. Так как контрапозиция эквивалентна $\overline{X} \to \overline{Y}$, то конверсия этой контрапозиции эквивалентна конверсии этой импликации.

С импликацией связано постоянное упоминание математиками «необходимое условие» и «достаточное условие».

X является достаточным условием для Y	Если имеет место X, то Y также будет иметь место	Импликация $X \to Y$
X является необходимым условием для Y	Если имеет место Y, то X также будет иметь место	Конверсия достаточного условия $Y \to X$
X является необходимым и достаточным условием для Y	X имеет место, если и только если имеет место Y	Двойная импликация $X \leftrightarrow Y$ эквивалентность

6. Основные законы, определяющие свойства введенных логических операций

1) Идемпотентность дизъюнкции и конъюнкции:

$$X \vee X \leftrightarrow X, \quad X \wedge X \leftrightarrow X.$$

2) Коммутативность дизъюнкции и конъюнкции:

$$X \vee Y \leftrightarrow Y \vee X, \quad X \wedge Y \leftrightarrow Y \wedge X.$$

3) Ассоциативность дизъюнкции и конъюнкции:

$$X \vee (Y \vee Z) \leftrightarrow (X \vee Y) \vee Z, \quad X \wedge (Y \wedge Z) \leftrightarrow (X \wedge Y) \wedge Z.$$

4) Дистрибутивность операций дизъюнкции и конъюнкции относительно друг друга:

$$X \vee (Y \wedge Z) \leftrightarrow (X \vee Y) \wedge (X \vee Z), \quad X \wedge (Y \vee Z) \leftrightarrow (X \wedge Y) \vee (X \wedge Z).$$

5) Двойное отрицание:

$$X \leftrightarrow \overline{\overline{X}}.$$

6) Закон де Моргана:

$$\overline{X} \vee \overline{Y} \leftrightarrow \overline{X \wedge Y}, \quad \overline{X} \wedge \overline{Y} \leftrightarrow \overline{X \vee Y}.$$

7) Склеивание:

$$(X \vee Y) \wedge (X \vee \overline{Y}) \leftrightarrow X, \quad (X \wedge Y) \vee (X \wedge \overline{Y}) \leftrightarrow X.$$

8) Поглощение:

$$X \vee (X \wedge Y) \leftrightarrow X, \quad X \wedge (X \vee Y) \leftrightarrow X.$$

9) Действие с логическими константами 0 и 1:

$$X \vee 0 \leftrightarrow X, \quad X \wedge 0 \leftrightarrow 0, \quad X \vee 1 \leftrightarrow 1, \quad X \wedge 1 \leftrightarrow X, \quad X \wedge \overline{X} \leftrightarrow 0.$$

10) Закон исключения третьего:

$$X \vee \overline{X} \leftrightarrow 1.$$

11) Тождество:

$$X \leftrightarrow X.$$

12) Отрицание противоречия:

$$\overline{\overline{X} \wedge X} \leftrightarrow 1.$$

13) Контрапозиция:

$$(X \to Y) \leftrightarrow (\overline{Y} \to \overline{X}).$$

14) Цепное заключение:

$$((X \to Y) \wedge (Y \to Z)) \leftrightarrow (X \to Z).$$

15) Противоположность:

$$(X \leftrightarrow Y) \leftrightarrow (\overline{X} \leftrightarrow \overline{Y}).$$

16) Модус поненс (modus ponens):

$$(X \wedge (X \to Y)) \to Y \leftrightarrow 1.$$

Сформулированные законы легко проверить с помощью таблицы истинности.

Заметим, что при исследовании различных высказываний на эквивалентность (равносильность) логическую связку $\leftrightarrow$ можно заменить обычным знаком равенства $=$.

Первое практическое занятие по теме «Логические операции»

Задача 1. Составьте таблицу истинности формулы: $X \oplus Y \to \overline{Z} \vee X \mid \overline{Y} \wedge \overline{X}$.
Решение. Расставим скобки: $(X \oplus Y) \to \left(\overline{Z} \vee (X \mid (\overline{Y} \wedge \overline{X}))\right)$.

X	Y	Z	$\overline{X}$	$\overline{Y}$	$\overline{Z}$	$X \oplus Y$	$\overline{Y} \wedge \overline{X}$	$X \mid (\overline{Y} \wedge \overline{X})$	$\overline{Z} \vee (X \mid (\overline{Y} \wedge \overline{X}))$	$(X \oplus Y) \to (X \mid (\overline{Y} \wedge \overline{X}))$
0	0	0	1	1	1	0	1	1	1	1
0	0	1	1	1	0	0	1	1	1	1
0	1	0	1	0	1	1	0	1	1	1
0	1	1	1	0	0	1	0	1	1	1
1	0	0	0	1	1	1	0	1	1	1
1	0	1	0	1	0	1	0	1	1	1
1	1	0	0	0	1	0	0	1	1	1
1	1	1	0	0	0	0	0	1	1	1

Задача 2. Докажите тождественную истинность формулы $\overline{X} \to (X \to Y)$.
Решение. Составим таблицу истинности:

X	Y	$\overline{X}$	$X \to Y$	$\overline{X} \to (X \to Y)$
0	0	1	1	1
0	1	1	1	1
1	0	0	0	1
1	1	0	1	1

Последний столбец состоит из 1, следовательно, доказана тождественная истинность формулы.

Задача 3. Докажите эквивалентность $X \wedge (X \vee Z) \wedge (Y \vee Z) \leftrightarrow (X \wedge Y) \vee (X \wedge Z)$.
Решение. Пусть $\varphi_1 = X \wedge (X \vee Z) \wedge (Y \vee Z)$. Составим таблицу истинности:

X	Y	Z	$X \vee Z$	$Y \vee Z$	$X \wedge (X \vee Z)$	$X \wedge (X \vee Z) \wedge (Y \vee Z)$
0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	1	0	0	1	0	0
0	1	1	1	1	0	0
1	0	0	1	0	1	0
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

Пусть $\varphi_2 = (X \wedge Y) \vee (X \wedge Z)$.

X	Y	Z	$X \wedge Y$	$X \wedge Z$	$(X \wedge Y) \vee (X \wedge Z)$
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

Заметим, что таблицы истинности для φ_1 и φ_2 совпадают, следовательно, эквивалентность доказана.

Задача 4. Для каждого из следующих высказываний: 1) найдите символическую форму; 2) постройте таблицу истинности. Воспользуйтесь буквенными обозначениями: X для «Джо умен»; Y для «Джим глуп»; Z для «Джо получит приз».

(a) Если Джо умен, а Джим глуп, то Джо получит приз.

(b) Джо получит приз в том и только в том случае, если он умен или если Джим глуп.

(c) Если Джим глуп, а Джо не удастся получить приз, то Джо не умен.

Решение. **(a)** $(X \wedge Y) \to Z$; **(b)** $Z \leftrightarrow (X \vee Y)$; **(c)** $(Y \wedge \overline{Z}) \to \overline{X}$.

X	Y	Z	$\overline{X}$	$\overline{Z}$	$X \wedge Y$	$(X \wedge Y) \to Z$	$X \vee Y$	$Z \leftrightarrow (X \vee Y)$	$Y \wedge \overline{Z}$	$(Y \wedge \overline{Z}) \to \overline{X}$
0	0	0	1	1	0	1	0	1	0	1
0	0	1	1	0	0	1	0	0	0	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	1	0	0	1	1	1	0	1
1	0	0	0	1	0	1	1	0	0	1
1	0	1	0	0	0	1	1	1	0	1
1	1	0	0	1	1	0	1	0	1	0
1	1	1	0	0	1	1	1	1	0	1
						(a)		**(b)**		**(c)**

Задача 5. Таблица истинности высказывания, составленного из двух простых высказываний, состоит из четырех строк; а таблица истинности высказывания, составленного из трех простых высказываний, — из восьми строк. Сколько строк должна иметь таблица истинности высказывания, составленного из четырех простых высказываний? Сколько — из пяти? Сколько — из n? Укажите способ систематической записи таблиц истинности для произвольного n?

Указание. Для систематической записи таблиц истинности для произвольного n можно применить метод «последовательного половинного деления столбцов» — столбец первой переменной делят пополам и заполняют верхнюю половину нулями, а нижнюю половину — единицами, затем каждую половину второго столбца делят пополам и опять заполняют полученные половины нулями и единицами и т. д.

Задача 6. Доказать равносильность, используя основные законы логических операций: $\overline{(X \wedge \overline{Y}) \vee (Y \wedge \overline{Z})} = (\overline{X} \wedge \overline{Y}) \vee (\overline{X} \wedge Z) \vee (Y \wedge Z)$.

Решение.

1. Используя законы де Моргана $\overline{X} \vee \overline{Y} \leftrightarrow \overline{X \wedge Y}$ и $\overline{X} \wedge \overline{Y} \leftrightarrow \overline{X \vee Y}$, получим:

$$\overline{(X \wedge \overline{Y}) \vee (Y \wedge \overline{Z})} = \overline{(X \wedge \overline{Y})} \wedge \overline{(Y \wedge \overline{Z})} = (\overline{X} \vee \overline{\overline{Y}}) \wedge (\overline{Y} \vee \overline{\overline{Z}}).$$

2. Используя закон двойного отрицания $X \leftrightarrow \overline{\overline{X}}$, получаем:

$$(\overline{X} \vee \overline{\overline{Y}}) \wedge (\overline{Y} \vee \overline{\overline{Z}}) = (\overline{X} \vee Y) \wedge (\overline{Y} \vee Z).$$

3. Применяя дистрибутивный закон $(X \vee Y) \wedge (X \vee Z) = X \vee (Y \wedge Z)$, получаем

$$(\overline{X} \vee Y) \wedge (\overline{Y} \vee Z) = \left((\overline{X} \vee Y) \wedge \overline{Y}\right) \vee \left((\overline{X} \vee Y) \wedge Z\right) =$$
$$= \left((\overline{X} \wedge \overline{Y}) \vee (Y \wedge \overline{Y})\right) \vee \left((\overline{X} \wedge Z) \vee (Y \wedge Z)\right).$$

4. Ассоциативность дизъюнкции: $X \vee (Y \vee Z) \leftrightarrow (X \vee Y) \vee Z$ позволяет упростить последнее выражение:

$$\left((\overline{X} \wedge \overline{Y}) \vee (Y \wedge \overline{Y})\right) \vee \left((\overline{X} \wedge Z) \vee (Y \wedge Z)\right) = (\overline{X} \wedge \overline{Y}) \vee (Y \wedge \overline{Y}) \vee (\overline{X} \wedge Z) \vee (Y \wedge Z).$$

5. Учитывая законы, включающие тождественно ложные высказывания, окончательно получаем:

$$(\overline{X} \wedge Y) \vee (Y \wedge \overline{Y}) \vee (\overline{X} \wedge Z) \vee (Y \wedge Z) = (\overline{X} \wedge \overline{Y}) \vee (\overline{X} \wedge Z) \vee (Y \wedge Z).$$

Задача 7. С помощью таблиц истинности проверить, являются ли эквивалентными высказывания: $f_1 = X \wedge (Y \to Z)$ и $f_2 = (\overline{X} \wedge Y) \vee (X \wedge Z)$.

Решение.

X	Y	Z	$\overline{X}$	$Y \to Z$	f_1	$\overline{X} \wedge Y$	$X \wedge Z$	f_2
0	0	0	1	1	0	0	0	0
0	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	1
0	1	1	1	1	0	1	0	1
1	0	0	0	1	1	0	0	0
1	0	1	0	1	1	0	1	1
1	1	0	0	0	0	0	0	0
1	1	1	0	1	1	0	1	1

Так как значения для высказываний f_1 и f_2 в таблице истинности не совпали, то они не эквивалентны.

Задача 8. Определите для каждого из следующих высказываний, будет ли оно логически истинным, противоречивым; ни тем, ни другим.

(а) $X \leftrightarrow X$; **(б)** $X \leftrightarrow \overline{X}$; **(в)** $(X \vee Y) \leftrightarrow (X \wedge Y)$; **(г)** $(X \to \overline{Y}) \to (Y \to \overline{X})$;
(д) $(X \to Y) \wedge (Y \to Z) \wedge \overline{(X \to Z)}$; **(е)** $(X \to Y) \to X$; **(ж)** $((X \to Y) \to X) \to X$.

Решение.

X	Y	$X \leftrightarrow X$	$\overline{X}$	$X \leftrightarrow \overline{X}$	$X \vee Y$	$X \wedge Y$	$(X \vee Y) \leftrightarrow (X \wedge Y)$
0	0	1	1	0	0	0	1
0	1	1	1	0	1	0	0
1	0	1	0	0	1	0	0
1	1	1	0	0	1	1	1
		(а)		**(б)**			**(в)**

Вывод: **(а)** — логически истинное; **(б)** — противоречивое; **(в)** — ни то, ни другое;

X	Y	$\overline{Y}$	$X \to \overline{Y}$	$Y \to \overline{X}$	$(X \to \overline{Y}) \to (Y \to \overline{X})$
0	0	1	1	1	1
0	1	0	1	1	1
1	0	1	1	1	1
1	1	0	0	0	1
					(г)

(г) — логически истинное;

X	Y	Z	$X \to Y$	$Y \to Z$	$(X \to Y) \wedge (Y \to Z)$	$X \to Z$	$\overline{X \to Z}$	$(X \to Y) \wedge (Y \to Z) \wedge$ $\wedge \overline{(X \to Z)}$
0	0	0	1	1	1	1	0	0
0	0	1	1	1	1	1	0	0
0	1	0	1	0	0	1	0	0
0	1	1	1	1	1	1	0	0
1	0	0	0	1	0	0	1	0
1	0	1	0	1	0	1	0	0
1	1	0	1	0	0	0	1	0
1	1	1	1	1	1	1	0	0
								(д)

(д) — противоречиво;

X	Y	$X \to Y$	$(X \to Y) \to X$	$((X \to Y) \to X) \to X$
0	0	1	0	1
0	1	1	0	1
1	0	0	1	1
1	1	1	1	1
			(е)	**(ж)**

(е) — ни то, ни другое; **(ж)** — логически истинно.

Задача 9. Покажите, что высказывание $(X \leftrightarrow Y) \to (X \to Y)$ — логически истинно, а $(X \leftrightarrow Y) \to (X \vee Y)$ — нет.

Задача 10. Постройте таблицы истинности следующих составных высказываний: **(а)** $X \wedge Y$; **(б)** $X \to \overline{Y}$; **(в)** $\overline{X} \vee \overline{Y}$; **(г)** $\overline{X} \vee Y$; **(д)** $X \wedge \overline{Y}$. Для каких пар имеет место отношение следствия или эквивалентности?

Ответ: **(б)** эквивалентно **(в)**; из **(а)** следует **(г)**; из **(д)** следует **(б)**, **(в)**.

Задача 11. Постройте таблицы истинности следующих составных высказываний и расположите их в таком порядке, чтобы из каждого высказывания следовали все, стоящие после него: (а) $\overline{X} \leftrightarrow Y$; (б) $X \to Y$; (в) $X \to (Y \to X)$; (г) $X \vee Y$; (д) $\overline{X} \wedge Y$.
Ответ: (в); (д); (а); (г); (б).

Задача 12. Постройте составные высказывания, эквивалентные а) $X \leftrightarrow Y$; б) $X \vee Y$, используя только связки отрицания и конъюнкции.

Задача 13. Если X и Y логически истинны, а Z — логически ложно, то что можно сказать о высказывании $(X \vee \overline{Y}) \wedge \overline{Z}$?
Ответ: логически истинно.

Задача 14. Докажите, что конъюнкция импликации и ее конверсия эквивалентны двойной импликации, т. е. $(X \to Y) \wedge (Y \to X) \leftrightarrow (X \leftrightarrow Y)$.

Задача 15. Чему эквивалентна конъюнкция контрапозиции и ее конверсии?

Задача 16. Докажите, что отрицание высказывания : «X есть необходимое и достаточное условие для Y» эквивалентно высказыванию: «X есть необходимое и достаточное условие для $\overline{Y}$».

Задача 17. Докажите, что контрапозиция эквивалентна первоначальной импликации.

Задача 18. Пусть X означает: «Я сдам этот экзамен»; а Y: «Я буду регулярно выполнять домашние задания». Запишите в символической форме следующие высказывания:

(а) «Я сдам этот экзамен только в том случае, если буду регулярно выполнять домашние задания».

(б) «Регулярное выполнение домашних заданий является необходимым условием для того, что я сдам этот экзамен».

(в) «Сдача этого экзамена является достаточным условием того, что я регулярно выполнял домашние задания».

(г) «Я сдам этот экзамен в том и только в том случае, если я буду регулярно выполнять домашние задания».

(д) «Регулярное выполнение домашних заданий есть необходимое и достаточное условие для того, чтобы я сдал этот экзамен».

Выясните, какому из перечисленных высказываний соответствуют следующие символические формы: $X \to Y$; $Y \leftrightarrow X$; $X \leftrightarrow Y$; $Y \to X$.

Задача 19. Докажите равносильность $\overline{X \to Y} = X \wedge \overline{Y}$ с помощью формул алгебры высказываний.
Решение. Используя формулу $X \to Y = \overline{X} \vee Y$, запишем: $\overline{X \to Y} = \overline{\overline{X} \vee Y}$, тогда $\overline{\overline{X} \vee Y} = \overline{\overline{X}} \wedge \overline{Y}$ по закону де Моргана, т. е. $\overline{X \to Y} = X \wedge \overline{Y}$, т. к. по закону двойного отрицания $\overline{\overline{X}} = X$, что и требовалось доказать.

Полученная формула дает правило построения отрицания для импликации, часто применяемое в математических рассуждениях: $\overline{X \to Y} = X \wedge \overline{Y}$.

Задача 20. Преобразуйте формулу $(X \to (Y \to Z)) \leftrightarrow \overline{Y \to X}$ к виду, не содержащему импликацию и эквивалентность.

Решение. Запишем цепочку преобразований:

$$(X \to (Y \to Z)) \leftrightarrow \overline{Y \to X} = \big(X \to (\overline{Y} \vee Z)\big) \leftrightarrow Y \wedge \overline{X} = \overline{X} \vee (\overline{Y} \vee Z) \leftrightarrow (Y \wedge \overline{X}) =$$

$$= \big((\overline{X} \vee (\overline{Y} \vee Z)) \wedge (Y \wedge \overline{X})\big) \vee \Big(\overline{\overline{X} \vee (\overline{Y} \vee Z)} \wedge \overline{(Y \wedge \overline{X})}\Big).$$

Задача 21. Проверьте, будут ли эквивалентны следующие формулы: **а)** $X \to (Y \oplus Z)$ и $(X \to Y) \oplus (X \to Z)$; **б)** $X \mid (Y \to Z)$ и $(X \mid Y) \to (X \mid Z)$; **в)** $X \downarrow (Y \leftrightarrow Z)$ и $\overline{(X \downarrow Y) \leftrightarrow (X \downarrow Z)}$.

Решение. Составим таблицы истинности:

(а)

X	Y	Z	$Y \oplus Z$	$X \to (Y \oplus Z)$	$X \to Y$	$X \to Z$	$(X \to Y) \oplus (X \to Z)$
0	0	0	0	1	1	1	0
0	0	1	1	1	1	1	0
0	1	0	1	1	1	1	0
0	1	1	0	1	1	1	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	0	0	1	1	0

Формулы не эквивалентны.

(б)

X	Y	Z	$Y \to Z$	$X \mid (Y \to Z)$	$X \mid Y$	$X \mid Z$	$(X \mid Y) \to (X \mid Z)$
0	0	0	1	1	1	1	1
0	0	1	1	1	1	1	1
0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	1
1	0	0	1	0	1	1	1
1	0	1	1	0	1	0	0
1	1	0	0	1	0	1	1
1	1	1	1	0	0	0	1

Формулы не эквивалентны.

(в)

X	Y	Z	$Y \leftrightarrow Z$	$X \downarrow (Y \leftrightarrow Z)$	$X \downarrow Y$	$X \downarrow Z$	$\overline{(X \downarrow Y) \leftrightarrow (X \downarrow Z)}$
0	0	0	1	0	1	1	0
0	0	1	0	1	1	0	1
0	1	0	0	1	0	1	1
0	1	1	1	0	0	0	0
1	0	0	1	0	0	0	0
1	0	1	0	0	0	0	0
1	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0

Формулы эквивалентны.

Задача 22. Составьте таблицы истинности для высказываний:

а) $X \mid X$; **б)** $(X \mid Y) \mid (X \mid Y)$,

и покажите, что любая таблица истинности может быть реализована посредством составного высказывания, в котором используется единственная связка: штрих Шеффера.

Задача 23. Постройте таблицы истинности для высказываний:

а) $X \downarrow X$; **б)** $(X \downarrow Y) \downarrow (X \downarrow Y)$.

Какие другие составные высказывания имеют те же таблицы истинности? Покажите, что любая таблица истинности может быть реализована посредством составного высказывания, в котором используется единственная связка: стрелка Пирса.

Задача 24. Докажите, что импликация $X \to Y$ эквивалентна $((X \wedge Y) \oplus X) \oplus 1$.

Решение. Доказательство проведем с помощью таблицы истинности.

X	Y	$X \to Y$	$X \wedge Y$	$(X \wedge Y) \oplus X$	1	$((X \wedge Y) \oplus X) \oplus 1$
0	0	1	0	0	1	1
0	1	1	0	0	1	1
1	0	0	0	1	1	0
1	1	1	1	0	1	1

Задача 25. Докажите эквивалентность формул:

$$f_1 = \big(X \wedge Y \vee (\overline{X} \to Y \wedge Z)\big) \leftrightarrow (\overline{X} \to \overline{Y}) \to Z;$$

$$f_2 = (X \to Y) \oplus (Y \oplus Z).$$

7. Булевы функции

Булева функция, или функция алгебры логики, является одним из основных объектов дискретной математики.

Булевы функции названы в честь Дж. Буля, положившего начало применению математики в логике.

Функцию $f(x_1, x_2, \dots, x_n)$, принимающую одно из двух значений 0 или 1, от n переменных, каждая из которых принимает одно из двух значений 0 или 1, будем называть булевой функцией $f(x_1, x_2, \dots, x_n)$ от n переменных.

Булева функция от n переменных сопоставляет каждому упорядоченному набору (кортежу), составленному из n элементов, 0 и 1, либо 1, либо 0.

Две булевы функции называются равными, если для любых одинаковых наборов значений переменных обе функции принимают одинаковые значения. Булевых функций одной переменной четыре, а двух переменных — шестнадцать и т. д. Число булевых функций от n переменных равно 2^{2^n}.

Рассмотрим функции одной и двух переменных, которые называются «элементарными» функциями и с помощью которых можно определить функции большего количества переменных.

Рассмотрим таблицы истинности таких функций.

Таблица истинности булевой функции одной переменной:

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
0	0	0	1	1
1	0	1	0	1

Функции $f_1(x)$ и $f_4(x)$) называются константами — соответственно 0 и 1.

Функция $f_2(x)$ совпадает с переменной x и называется тождественной $f_2(x) = x$:

Функция $f_3(x)$ принимает значения, противоположные значениям аргумента x, и называется отрицанием x, обозначается $\overline{x}$: $f_3(x) = \overline{x}$.

Таблица истинности булевой функции двух переменных:

x_1	x_2	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Следует отметить, что здесь к функциям двух переменных относятся и такие, которые в действительности зависят от одной переменной.

1. Функции f_1 и f_{16} представляют собой константы 0 и 1.

2. Функции f_4, f_6, f_{11}, f_{13} существенно зависят только от одной переменной: $f_4 = x_1$, $f_6 = x_2$, $f_{11} = \overline{x_2}$, $f_{13} = \overline{x_1}$.

3. Остальные функции существенно зависят от двух переменных, и для них есть названия и обозначения:

а) функция $f_2 = x_1 \wedge x_2$ и называется конъюнкцией,

б) функция $f_8 = x_1 \vee x_2$ и называется дизъюнкцией,

в) функция $f_{10} = x_1 \leftrightarrow x_2$ и называется эквивалентностью,

г) функция $f_7 = x_1 \oplus x_2$ и называется суммой по модулю два, или суммой Жегалкина,

д) функция $f_{12} = x_2 \to x_1$ и называется конверсией,

е) функция $f_{14} = x_1 \to x_2$ и называется импликацией,

ж) функция $f_{15} = x_1 \mid x_2$ и называется штрих Шеффера,

з) функция $f_9 = x_1 \downarrow x_2$ и называется стрелкой Пирса,

и) функции f_3 и f_5 логически несовместимы с импликацией и конверсией и называются функциями запрета.

8. Свойства элементарных булевых функций

Для булевых функций справедливы равенства, аналогичные формулам, сформулированным для высказываний.

1. Функции: конъюнкция, дизъюнкция, сумма по модулю два, стрелка Пирса, штрих Шеффера обладают свойством коммутативности.

2. Функции: конъюнкция, дизъюнкция, сумма по модулю два обладают свойством ассоциативности и свойством дистрибутивности.

3. Закон де Моргана: $\overline{x_1 \wedge x_2} = \overline{x_1} \vee \overline{x_2}$; $\overline{x_1 \vee x_2} = \overline{x_1} \wedge \overline{x_2}$.

4. Закон двойного отрицания: $\overline{\overline{x}} = x$.

5. Выражение дизъюнкции через конъюнкцию и суммы по модулю два: $x_1 \vee x_2 = x_1 \wedge x_2 \oplus x_2 \oplus x_1$.

6. Выражение дизъюнкции через импликацию:

$$(x_1 \to x_2) \to x_2 = x_1 \vee x_2.$$

7. Выражение отрицания через штрих Шеффера, стрелку Пирса, сумму по модулю два и эквивалентность: $x \mid x = x \downarrow x = \overline{x} = x \oplus 1 = x \leftrightarrow 0$.

8. Выражение конъюнкции через штрих Шеффера:

$$(x_1 \mid x_2) \mid (x_1 \mid x_2) = x_1 \wedge x_2.$$

9. Выражение дизъюнкции через стрелку Пирса:

$$(x_1 \downarrow x_2) \downarrow (x_1 \downarrow x_2) = x_1 \vee x_2.$$

10. Закон поглощения: $x_1 \wedge x_2 \vee x_1 = x_1$; $x_1 \wedge (x_1 \vee x_2) = x_1$.

11. Закон склеивания: $\overline{x} \vee x = \overline{x} \oplus x = 1$.

12. Для функций: конъюнкция, дизъюнкция и сумма по модулю два справедливы следующие тождества:

$$\begin{array}{lll} x \wedge x = x; & x \vee x = x; & x \oplus x = 0; \\ \overline{x} \wedge x = 0; & \overline{x} \vee x = 1; & x \oplus \overline{x} = 1; \\ x \wedge 0 = 0; & x \vee 0 = x; & x \oplus 0 = x; \\ x \wedge 1 = x; & x \vee 1 = 1; & x \oplus 1 = \overline{x}. \end{array}$$

13. Для функций конъюнкция и дизъюнкция справедливы тождества: $x_1 \vee \overline{x_1} \wedge x_2 = x_1 \vee x_2$; $x_1 \wedge (\overline{x_2} \vee x_1) = x_1 \wedge x_2$.

Для доказательства справедливости любых из приведенных тождеств нужно составить таблицы истинности для булевых функций.

Булеву функцию любого числа переменных можно задать формулой, содержащей функции одной и двух переменных посредством подстановки одних булевых функций вместо переменных в другие булевы функции, т. е. посредством суперпозиции булевых функций.

9. Дизъюнктивные и конъюнктивные нормальные формы алгебры высказываний

Конъюнктивным одночленом от переменных $x_1, x_2, \ldots, x_n$ называется конъюнкция этих переменных или их отрицаний.

Дизъюнктивным одночленом от переменных $x_1, x_2, \ldots, x_n$ называется дизъюнкция этих переменных или их отрицаний.

Формула, равносильная данной формуле алгебры высказываний и являющаяся дизъюнкцией элементарных конъюнктивных одночленов, называется **дизъюнктивной нормальной формой** (ДНФ) данной формулы.

Например: $(x_1 \wedge x_2 \wedge x_3) \vee (x_1 \wedge \overline{x_2}) \vee (x_3 \wedge x_2) \vee \overline{x_3}$ — ДНФ.

Формула, равносильная данной формуле алгебры высказываний и являющаяся конъюнкцией элементарных дизъюнктивных одночленов, называется **конъюнктивной нормальной формой** (КНФ) данной формулы.

Например: $(\overline{x_1} \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x_3}) \wedge x_2$ — КНФ.

Для каждой формулы алгебры высказываний можно найти множество дизъюнктивных и конъюнктивных нормальных форм.

Алгоритм построения

(1) Избавиться от всех логических операций, содержащихся в формуле, заменив их основными: конъюнкцией, дизъюнкцией, отрицанием. Это можно сделать, используя равносильные формулы:

$$x_1 \to x_2 = \overline{x_1} \vee x_2; \quad x_1 \leftrightarrow x_2 = (\overline{x_1} \vee x_2) \wedge (x_1 \vee \overline{x_2});$$
$$x_1 \leftrightarrow x_2 = (x_1 \wedge x_2) \vee (\overline{x_1} \wedge \overline{x_2}).$$

(2) Заменить знак отрицания, относящийся к выражениям типа $\overline{x_1 \wedge x_2}$ или $\overline{x_1 \vee x_2}$, знаками отрицания, относящимися к отдельным переменным высказываниям на основании формул:

$$\overline{x_1 \vee x_2} = \overline{x_1} \wedge \overline{x_2}; \quad \overline{x_1 \wedge x_2} = \overline{x_1} \vee \overline{x_2}.$$

(3) Избавиться от знаков двойного отрицания.

(4) Применить, если нужно, к операциям конъюнкции и дизъюнкции свойства дистрибутивности и формулы поглощения.

10. Совершенная дизъюнктивная и совершенная конъюнктивная нормальные формы

Любая булева функция может иметь много представлений в виде ДНФ и КНФ. Особое место среди этих представлений занимают совершенные ДНФ (СДНФ) и совершенные КНФ (СКНФ).

Совершенная дизъюнктивная нормальная форма (СДНФ) — это ДНФ, в которой в каждый конъюнктивный одночлен каждая переменная x_i из набора $f(x_1, x_2, \ldots, x_n)$ входит ровно один раз, причем входит либо сама x_i, либо ее отрицание $\overline{x_i}$.

Конструктивно СДНФ для каждой формулы алгебры высказываний, приведенной к ДНФ, можно определить так:

Совершенной дизъюнктивной нормальной формой (СДНФ) формулы алгебры высказываний называется ее ДНФ, обладающая следующими свойствами:

1. ДНФ не содержит двух одинаковых конъюнкций.

2. Ни одна конъюнкция не содержит одновременно двух одинаковых переменных.

3. Ни одна конъюнкция не содержит одновременно некоторую переменную и ее отрицание.

4. Каждая конъюнкция содержит либо переменную x_i, либо ее отрицание $\overline{x_i}$ для всех переменных, входящих в формулу.

Конструктивно СКНФ для каждой формулы алгебры высказываний, приведенной к КНФ, можно определить так:

Совершенной конъюнктивной нормальной формой (СКНФ) данной формулы алгебры высказываний называется такая ее КНФ, которая удовлетворяет следующим свойствам:

1. КНФ не содержит двух одинаковых дизъюнкций.

2. Ни одна из дизъюнкций не содержит одновременно двух одинаковых переменных.

3. Ни одна из дизъюнкций не содержит одновременно некоторую переменную и ее отрицание.

4. Каждая дизъюнкция СКНФ содержит либо переменную x_i, либо ее отрицание x_i для всех переменных, входящих в формулу.

Сформулируем следующие теоремы:

Теорема 1. Произвольную булеву функцию $f(x_1, x_2, \ldots, x_n)$ можно задать формулой $f(x_1, x_2, \ldots, x_n) = \bigvee(x_1^{\tau_1} \wedge x_2^{\tau_2} \wedge \ldots \wedge x_n^{\tau_n})$, где дизъюнкция берется по всем $\tau = (\tau_1, \tau_2, \ldots, \tau_n)$, где $f(\tau) = 1$ и

$$x_i = \begin{cases} \overline{x_i}, & \text{если } \tau = 0, \\ x_i, & \text{если } \tau = 1. \end{cases}$$

Теорема 2. Произвольную булеву функцию $f(x_1, x_2, \ldots, x_n)$ можно задать формулой $f(x_1, x_2, \ldots, x_n) = \bigwedge(x_1^{\overline{\tau_1}} \vee x_2^{\overline{\tau_2}} \vee \ldots \vee x_n^{\overline{\tau_n}})$, где конъюнкция берется по всем $\overline{\tau} = (\overline{\tau_1}, \overline{\tau_2}, \ldots, \overline{\tau_n})$, где $f(\overline{\tau}) = 0$ и

$$x_i = \begin{cases} x_i, & \text{если } \overline{\tau} = 0, \\ \overline{x_i}, & \text{если } \overline{\tau} = 1. \end{cases}$$

Эти формулы называются соответственно **совершенной дизъюнктивной нормальной формой** или **совершенной конъюнктивной нормальной формой** булевой функции $f(x_1, x_2, \ldots, x_n)$. Исходя из таблицы истинности булевой функции, можно построить СДНФ функции: для каждого набора $\tau = (\tau_1, \tau_2, \ldots, \tau_n)$, такого, что $f(\tau) = 1$, составляется конъюнкция $x_1^{\tau_1} \wedge x_2^{\tau_2} \wedge \ldots \wedge x_n^{\tau_n}$, а затем все эти конъюнкции соединяем знаком дизъюнкции.

Для построения СКНФ функции выписываем наборы $\overline{\tau} = (\overline{\tau_1}, \overline{\tau_2}, \ldots, \overline{\tau_n})$ такие, что $f(\overline{\tau}) = 0$. Для такого набора составляется дизъюнкция

$$x_1^{\overline{\tau_1}} \vee x_2^{\overline{\tau_2}} \vee \ldots \vee x_n^{\overline{\tau_n}},$$

а затем все такие дизъюнкции соединяют знаком конъюнкции.

Приведенные формулы позволяют сформулировать следующие утверждения:

1. Каждая булева функция от n переменных, отличная от константы 0, имеет единственную СДНФ.

2. Каждая булева функция от n переменных, отличная от константы 1, имеет единственную СКНФ.

Эти утверждения называются **теоремой о функциональной полноте.**

11. Многочлены Жегалкина

Согласно сформулированным утверждениям, можно говорить, что система булевых функций полна. Тогда любую булеву функцию можно представить в виде многочлена от своих переменных и такой многочлен называется многочленом Жегалкина.

Многочленом Жегалкина называется многочлен, являющийся суммой константы и различных одночленов, в которые каждая из переменных входит не выше, чем в первой степени.

Многочлен Жегалкина константы равен самой же константе; многочлен Жегалкина булевой функции одной переменной $f(x) = a_0 \oplus a_1 x$; многочлен Жегалкина булевой функции двух переменных

$$f(x_1, x_2) = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus a_{12}(x_1 \wedge x_2);$$

многочлен Жегалкина булевой функции трех переменных

$$f(x_1, x_2, x_3) = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus a_3 x_3 \oplus \\ \oplus a_{12}(x_1 \wedge x_2) \oplus a_{13}(x_1 \wedge x_3) \oplus a_{23}(x_2 \wedge x_3) \oplus a_{123}(x_1 \wedge x_2 \wedge x_3)$$

и т. д. Коэффициенты $a_{1,\dots,i}$ и свободный член a_0 принимают значения 0 или 1, а число слагаемых в формуле равно 2^n, где n — число переменных. Знак $\oplus$ — сумма Жегалкина или сумма по модулю два.

Теорема 3 (Жегалкина). Каждая булева функция $f(x_1, x_2, \dots, x_n)$ может быть представлена в виде многочлена Жегалкина и притом единственным образом, с точностью до порядка слагаемых.

Сформулируем алгоритм построения многочлена Жегалкина.

Выше было указано, что любую функцию, отличную от константы 0, можно представить в виде СДНФ. Если сравним таблицы истинности дизъюнкции и суммы по модулю два, видим, что они отличаются только последней строкой, т. е. на наборе 11. Так как в СДНФ на каждом наборе только одна конъюнкция равна 1, то все дизъюнкции можно заменить суммами по модулю два. Кроме того, известно, что $\overline{x} = x \oplus 1$. На этом и основан первый алгоритм построения многочлена Жегалкина:

1. Находим множество тех двоичных наборов, на которых функция принимает значение 1.

2. Составляем СДНФ.

3. В СДНФ каждый знак дизъюнкции меняем на знак суммы Жегалкина.

4. Упрощаем, если можно, полученное выражение, используя тождество $\overline{x_i} \oplus x_i = 1$.

5. В полученной формуле каждое отрицание $\overline{x_i}$ заменяем на $x_i \oplus 1$.

6. Раскрываем скобки в полученной формуле, содержащей только функции $\wedge$ и $\oplus$ и константу 1.

7. Приводим подобные члены, используя тождество $x_i \oplus x_i = 0$.

Используя метод неопределенных коэффициентов, получаем второй алгоритм определения многочлена Жегалкина:

1. Составляем систему линейных уравнений относительно 2^n неизвестных коэффициентов, содержащую 2^n уравнений, решением которой являются коэффициенты $a_0, a_1, \ldots, a_{1,2,\ldots,n}$ многочлена Жегалкина.

Многочлен Жегалкина называется **нелинейным**, если он содержит конъюнкции переменных, а если он не содержит конъюнкции переменных, то он называется **линейным**.

Функция $f(x_1, x_2, \ldots, x_n)$ называется линейной, если ее многочлен Жегалкина имеет вид $a_0 \oplus a_1 x_1 \oplus \ldots \oplus a_n x_n$, и нелинейной в противном случае.

Из определения многочлена Жегалкина следует, что для любой булевой функции коэффициенты при переменных $x_1, x_2, \ldots, x_n$ и свободный член вычисляются по формулам:

$$
\begin{gathered}
a_0 = f(0, \ldots, 0), \\
a_1 = f(0, \ldots, 0) \oplus f(1, \ldots, 0), \\
a_2 = f(0, \ldots, 0) \oplus f(0, 1, \ldots, 0), \\
\ldots \\
a_n = f(0, \ldots, 0) \oplus f(0, \ldots, 1).
\end{gathered}
$$

На этом основан алгоритм определения линейности (или нелинейности) булевой функции.

1. По таблицам истинности булевой функции $f(x_1, x_2, \ldots, x_n)$ и выше указанным формулам находим коэффициенты: $(a_0, a_1, \ldots, a_n)$.

2. Выписываем многочлен $\Phi(x_1, \dots, x_n) = a_0 \oplus a_1x_1 \oplus \dots \oplus a_nx_n$ и проверяем, задает ли он эту функцию. Для этого строим таблицу истинности многочлена $\Phi(x_1, x_2, \dots, x_n)$ и сравниваем ее с таблицей истинности функции $f(x_1, x_2, \dots, x_n)$.

Если таблицы истинности совпадают, то функция $f(x_1, x_2, \dots, x_n)$ линейная и $\Phi(x_1, x_2, \dots, x_n)$ — ее многочлен Жегалкина. В противном случае функция нелинейная.

Второе практическое занятие по теме «Булевы функции. Многочлены Жегалкина»

Задача 1. Составьте таблицу истинности булевой функции трех переменных $f(x_1, x_2, x_3) = x_1 \oplus x_2 \to \overline{x_3} \vee x_1 \mid (\overline{x_2} \wedge \overline{x_1})$ и найдите ее двоичный набор.

Решение. Для вычисления значений функции следует определить порядок выполнения операций. Это можно сделать многими способами. Пусть, например, порядок выполнения операций будет следующим:

$$f_1 = x_1 \oplus x_2; \quad f_2 = \overline{x_2} \wedge \overline{x_1}; \quad f_3 = x_1 \mid f_2; \quad f_4 = \overline{x_3} \vee f_3; \quad f_5 = f_1 \to f_4.$$

Последовательно составляются таблицы истинности всех указанных функций.

x_1	x_2	x_3	$\overline{x_1}$	$\overline{x_2}$	$\overline{x_3}$	f_1	f_2	f_3	f_4	f_5
0	0	0	1	1	1	0	1	1	1	1
0	0	1	1	1	0	0	1	1	1	1
0	1	0	1	0	1	1	0	1	1	1
0	1	1	1	0	0	1	0	1	1	1
1	0	0	0	1	1	1	0	1	1	1
1	0	1	0	1	0	1	0	1	1	1
1	1	0	0	0	1	0	0	1	1	1
1	1	1	0	0	0	0	0	1	1	1

Лексикографическое упорядочение наборов в таблице истинности булевой функции позволяет задать функцию двоичным набором длины 2^n, который будем обозначать буквой F. Двоичный набор данной функции $F = 11111111$. Отметим, что двоичный набор определяет булеву функцию в том и только в том случае, когда его длина есть степень двойки, а соответствующий показатель степени определяет число переменных данной функции.

Задача 2. Докажите тождественную истинность формулы $\overline{x} \to (x \to y)$.

Решение. Необходимо показать, что двоичный набор данной формулы $F = 1111$.

Составим таблицу истинности:

x	y	$\overline{x}$	$x \to y$	$\overline{x} \to (x \to y)$
0	0	1	1	1
0	1	1	1	1
1	0	0	0	1
1	1	0	1	1

Задача 3. Докажите эквивалентность функций: $f(x, y, z) = x \wedge (x \vee z) \wedge (y \vee z)$ и $f(x, y, z) = (x \wedge z) \vee (x \wedge z)$.

Решение. Для доказательства необходимо построить таблицы истинности этих функций, и если их двоичные наборы совпадут, то эквивалентность будет доказана.

x	y	z	$x \vee z$	$y \vee z$	$x \wedge (x \vee z)$	$x \wedge (x \vee z) \wedge (y \vee z)$
0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	1	0	0	1	0	0
0	1	1	1	1	0	0
1	0	0	1	0	0	0
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

x	y	z	$x \wedge y$	$x \wedge z$	$(x \wedge y) \vee (x \wedge z)$
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

Получаем $F_1 = 00000111$ и $F_2 = 00000111$. Значит, функции эквивалентны.

Задача 4. Используя СДНФ, найдите булеву функцию, принимающую значение 1 на следующих наборах переменных, и только на них:

$$f(0, 1, 0) = f(1, 0, 1) = f(1, 1, 1) = 1.$$

Решение. Алгоритм построения СДНФ.

1. Наборам 010; 101; 111 соответствуют конъюнкции: $\overline{x_1} \wedge x_2 \wedge \overline{x_3}$; $x_1 \wedge \overline{x_2} \wedge x_3$; $x_1 \wedge x_2 \wedge x_3$. Напомним, что для каждого набора из нулей и единиц τ_1, τ_2, τ_3 выписываем конъюнкцию $x_1^{\tau_1} \wedge x_2^{\tau_2} \wedge x_3^{\tau_3}$, причем, если $\tau_i = 1$, то соответствующая переменная x_i входит в конъюнкцию без отрицания.

2. Составим дизъюнкцию полученных конъюнкций, т. е. составляем СДНФ функции:

$$f(x_1, x_2, x_3) = (\overline{x_1} \wedge x_2 \wedge \overline{x_3}) \vee (x_1 \wedge \overline{x_2} \wedge x_3) \vee (x_1 \wedge x_2 \wedge x_3).$$

Задача 5. Составьте СКНФ функции $f(x_1, x_2) = x_1 \oplus x_2$.

Решение.

x_1	x_2	$x_1 \oplus x_2$
0	0	0
0	1	1
1	0	1
1	1	0

1. Выпишем $f(0, 0) = 0$; $f(1, 1) = 0$, булева функция принимает значение 0 на наборах $(0; 0)$ и $(1; 1)$.

2. Составим дизъюнкции, соответствующие этим наборам: $x_1 \vee x_2$ и $\overline{x_1} \vee \overline{x_2}$ (если $\overline{\tau} = 0$, то переменная входит в дизъюнкцию без отрицания, если $\overline{\tau} = 1$, то переменная в дизъюнкции берется с отрицанием).

3. Составим конъюнкцию полученных дизъюнкций, т. е. составляем СКНФ функции $f(x_1, x_2) = (x_1 \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2})$.

Задача 6. Постройте КНФ функций и доказать тождественную истинность с помощью таблицы истинности:

а) $f_1(x_1, x_2, x_3) = (x_1 \vee \overline{x_2}) \to x_3$;

б) $f_1(x_1, x_2, x_3, x_4) = (x_1 \wedge (x_2 \to x_3)) \to x_4$.

Решение. Напомним процедуру построения КНФ.

1. Исключаем связку $\to$ с помощью законов преобразования переменных: $(x \to y) = \overline{x} \vee y$.

2. Исключаем двойное отрицание с помощью правила $\overline{\overline{x}} = x$ и используем законы де Моргана: $\overline{x_1 \vee x_2} = \overline{x_1} \wedge \overline{x_2}$ или $\overline{x_1 \wedge x_2} = \overline{x_1} \vee \overline{x_2}$.

3. Для получения нормальной формы используем дистрибутивные законы:

$$x_1 \vee (x_2 \wedge x_3) = (x_1 \vee x_2) \wedge (x_1 \vee x_3),$$
$$x_1 \wedge (x_2 \vee x_3) = (x_1 \wedge x_2) \vee (x_1 \wedge x_3).$$

а) $f(x_1, x_2, x_3) = (x_1 \vee \overline{x_2}) \to x_3 = \overline{(x_1 \vee \overline{x_2})} \vee x_3 = (\overline{x_1} \wedge \overline{\overline{x_2}}) \vee x_3 =$

$= (\overline{x_1} \wedge x_2) \vee x_3 = (\overline{x_1} \vee x_3) \wedge (x_2 \vee x_3)$.

б) $f(x_1, x_2, x_3) = (x_1 \wedge (x_2 \to x_3)) \to x_4 = (x_1 \wedge (\overline{x_2} \vee x_3)) \to x_4 =$

$= \overline{(x_1 \wedge (\overline{x_2} \vee x_3))} \vee x_4 = \left(\overline{x_1} \vee \overline{(\overline{x_2} \to x_3)}\right) \vee x_4 = \left(\overline{x_1} \vee (\overline{\overline{x_2}} \wedge \overline{x_3})\right) \vee x_4 =$

$= (\overline{x_1} \vee (x_2 \wedge \overline{x_3})) \vee x_4 = ((\overline{x_1} \vee x_2) \wedge (\overline{x_1} \vee \overline{x_3})) \vee x_4 = (\overline{x_1} \vee x_2 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3} \vee x_4)$.

Таблица истинности для функции **(б)**:

x_1	x_2	x_3	x_4	f_1 $x_2 \to x_3$	f_2 $x_1 \wedge f_1$	$f_2 \to x_4$	$\overline{x_1}$	f_3 $\overline{x_1} \vee x_2$	f_6 $f_3 \vee x_4$	$\overline{x_3}$	f_4 $\overline{x_1} \vee \overline{x_3}$	f_5 $f_4 \vee x_4$	$f_6 \wedge f_5$
0	0	0	0	1	0	1	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1	1	1
0	0	1	0	1	0	1	1	1	1	0	1	1	1
0	0	1	1	1	0	1	1	1	1	0	1	1	1
0	1	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	1	0	0	1	1	1	1	1	1	1	1
0	1	1	0	1	0	1	1	1	1	0	1	1	1
0	1	1	1	1	0	1	1	1	1	0	1	1	1
1	0	0	0	1	1	0	0	0	0	1	1	1	0
1	0	0	1	1	1	1	0	0	1	1	1	1	1
1	0	1	0	1	1	0	0	0	0	0	0	0	0
1	0	1	1	1	1	1	0	0	1	0	0	1	1
1	1	0	0	0	0	1	0	1	1	1	1	1	1
1	1	0	1	0	0	1	0	1	1	1	1	1	1
1	1	1	0	1	1	0	0	1	1	0	0	0	0
1	1	1	1	1	1	1	0	1	1	0	0	1	1

Задача 7. Приведите к ДНФ формулу $f = \left((x \to y) \downarrow \overline{(y \to z)}\right)$.
Решение. Выразим логические операции $\to$ и $\downarrow$ через $\wedge$, $\vee$ и $^{\overline{\ \ }}$:

$$f = \left((\overline{x} \vee y) \downarrow \overline{(\overline{y} \vee z)}\right) = \overline{(\overline{x} \vee y) \vee \overline{(\overline{y} \vee z)}}.$$

В полученной формуле перенесем отрицание к переменным и сократим двойные отрицания:

$$f = \overline{(\overline{x} \vee y) \vee \overline{(\overline{y} \vee z)}} = (\overline{\overline{x}} \vee \overline{y}) \wedge (\overline{y} \vee z) = (x \wedge \overline{y}) \wedge (\overline{y} \vee z).$$

Используя закон дистрибутивности, приводим формулу к ДНФ:

$$f = (x \wedge \overline{y} \wedge \overline{y}) \vee (x \wedge \overline{y} \wedge z).$$

Задача 8. Приведите к КНФ формулу $f = (x \to y) \wedge ((\overline{y} \to z) \to \overline{x})$.
Решение. Преобразуем формулу f к формуле, не содержащей $\to$:

$$f = (\overline{x} \vee y) \wedge \left(\overline{(\overline{y} \to z)} \vee \overline{x}\right) = (\overline{x} \vee y) \wedge \left(\overline{(\overline{\overline{y}} \vee z)} \vee \overline{x}\right).$$

В полученной формуле перенесем отрицание к переменным и сократим двойные отрицания:

$$f = (\overline{x} \vee y) \wedge ((\overline{\overline{y}} \wedge \overline{z}) \vee \overline{x}) = (\overline{x} \vee y) \wedge ((\overline{y} \wedge \overline{z}) \vee \overline{x}).$$

По закону дистрибутивности получим: $f = (\overline{x} \vee y) \wedge (\overline{x} \vee \overline{y}) \wedge (\overline{x} \vee \overline{z})$, являющейся КНФ.

Замечание. Если полученную формулу упростить, используя законы дистрибутивности, эквивалентности и поглощения, то получим:

$$(\overline{x} \vee y) \wedge (\overline{x} \vee \overline{y}) \wedge (\overline{x} \vee \overline{z}) = \overline{x} \vee (y \wedge \overline{y}) \wedge (\overline{x} \vee \overline{z}) = \overline{x} \wedge (\overline{x} \vee \overline{z}) = \overline{x}.$$

Таким образом, мы получили формулу, которая является одновременно ДНФ и КНФ.

Задача 9. Найдите СДНФ и СКНФ функции $f(x_1, x_2, x_3)$, заданной следующей таблицей истинности:

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
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	0
1	1	1	1

Решение. По теореме о функциональной полноте СДНФ имеет вид:

$$(\overline{x_1} \wedge \overline{x_2} \wedge \overline{x_3}) \vee (x_1 \wedge \overline{x_2} \wedge x_3) \vee (\overline{x_1} \wedge x_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge x_3),$$

СКНФ имеет вид:

$$(x_1 \vee x_2 \vee \overline{x_3}) \wedge (x_1 \vee \overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3).$$

Описанный способ нахождения СДНФ и СКНФ по таблице истинности бывает часто более трудоемким. Для нахождения СДНФ данную формулу приводим сначала к ДНФ, а затем преобразовываем ее конъюнкции с помощью следующих действий:

а) если в конъюнкцию входит некоторая переменная со своим отрицанием, то мы удаляем эту конъюнкцию из ДНФ;

б) если в конъюнкцию одна и та же переменная входит несколько раз, то все они удаляются, кроме одной;

в) если в конъюнкцию не входят некоторые переменные, то для каждой из них к конъюнкции добавляется соответствующая формула вида $(x \vee \overline{x})$;

г) если в полученной ДНФ имеется несколько одинаковых конъюнкций, то оставляем только одну из них.

В результате получается СДНФ.

Задача 10. Найдите СДНФ для ДНФ $(x \wedge \overline{x}) \vee x \vee (y \wedge z \wedge y)$.

Решение.

1. Удаляем конъюнкцию $(x \wedge \overline{x})$, так как здесь переменная вместе со своим отрицанием. Остается $x \vee (y \wedge z \wedge y))$.

2. Из конъюнкции $y \wedge z \wedge y$ удаляем переменную y, так как она входит сюда два раза. Остается $x \vee (y \wedge z)$.

3. В первой конъюнкции нет переменной y, поэтому к ней добавляется формула $(y \vee \overline{y})$, а во второй конъюнкции нет переменной x, поэтому к ней добавляется формула $(x \vee \overline{x})$. Получаем:

$$(x \wedge (y \vee \overline{y})) \vee (y \wedge z \wedge (x \vee \overline{x})).$$

4. Используем дистрибутивные законы:

$$(x \wedge y) \vee (x \wedge \overline{y}) \vee (x \wedge y \wedge z) \vee (\overline{x} \wedge y \wedge z).$$

5. К первой и второй конъюнкциям добавляем $(z \vee \overline{z})$ и получаем:

$$(x \wedge y \wedge (z \vee \overline{z})) \vee (x \wedge \overline{y} \wedge (z \vee \overline{z})) \vee (x \wedge y \wedge z) \vee (\overline{x} \wedge y \wedge z).$$

6. Используем дистрибутивные законы:

$$(x \wedge y \wedge z) \vee (x \wedge y \wedge \overline{z}) \vee (x \wedge \overline{y} \wedge z) \vee (x \wedge \overline{y} \wedge \overline{z}) \vee (x \wedge y \wedge z) \vee (\overline{x} \wedge y \wedge z).$$

7. В полученной формуле имеется две одинаковые конъюнкции: $x \wedge y \wedge z$. Удалив одну из них, получим:

$$(x \wedge y \wedge z) \vee (x \wedge y \wedge \overline{z}) \vee (x \wedge \overline{y} \wedge z) \vee (x \wedge \overline{y} \wedge \overline{z}) \vee (\overline{x} \wedge y \wedge z).$$

В итоге мы получили соответствующую СДНФ.

Задача 11. Найдите СКНФ для КНФ $(x \vee z \vee \overline{y}) \wedge (x \vee z) \wedge y$.

Решение. Опишем алгоритм приведения КНФ к СКНФ аналогично вышеизложенному приведению ДНФ к СДНФ.

1. Во второй дизъюнкции не хватает переменной y, поэтому в дизъюнкцию добавим $(y \wedge \overline{y})$ и, используя дистрибутивные законы, получаем:

$$x \vee z = x \vee z \vee (y \wedge \overline{y}) = (x \vee z \vee y) \wedge (x \vee z \vee \overline{y}).$$

2. В третью дизъюнкцию добавим $(x \wedge \overline{x})$ и получим две дизъюнкции: $y = y \vee (x \wedge \overline{x}) = (y \vee x) \wedge (y \vee \overline{x})$. Добавив в каждую из них $z \wedge \overline{z}$, получим:

$$(y \vee x \vee (z \wedge \overline{z})) \wedge (y \vee \overline{x} \vee (z \wedge \overline{z})) = (y \vee x \vee z) \wedge (y \vee x \vee \overline{z}) \wedge (y \vee \overline{x} \vee \overline{z}) \wedge (y \vee \overline{x} \vee z).$$

3. Соберем в конъюнкцию все дизъюнкции:

$$(x \vee \overline{y} \vee z) \wedge (x \vee y \vee z) \wedge (x \vee z \vee \overline{y}) \wedge (y \vee x \vee z) \wedge (x \vee y \vee \overline{z}) \wedge (\overline{x} \vee y \vee z) \wedge (\overline{x} \vee y \vee \overline{z}).$$

4. Избавляемся от одинаковых дизъюнкций, оставляя только одну. В результате получается СКНФ:

$$(x \vee y \vee z) \wedge (\overline{x} \vee y \vee z) \wedge (x \vee \overline{y} \vee z) \wedge (x \vee y \vee \overline{z}) \wedge (\overline{x} \vee y \vee \overline{z}).$$

Задача 12. Задана булева функция трех переменных:

$$f(x_1, x_2, x_3) = \overline{x_2} \wedge \left((x_1 \vee x_3) \mid \overline{(\overline{x_2} \mid \overline{x_3})}\right);$$

а) постройте таблицу истинности, найдите двоичную форму F булевой функции и приведите функцию к СДНФ и СКНФ,

б) найдите двумя способами многочлен Жегалкина.

Решение.

а) $f_1 = x_1 \vee x_3$; $f_2 = \overline{x_2} \mid \overline{x_3}$; $f_3 = \overline{(\overline{x_2} \mid \overline{x_3})} = \overline{f_2}$;

$$f_4 = (x_1 \vee x_3) \mid \overline{(\overline{x_2} \mid \overline{x_3})} = f_1 \mid f_3; \quad f_5 = \overline{x_2} \wedge \left((x_1 \vee x_2) \mid \overline{(\overline{x_2} \mid \overline{x_3})}\right) = \overline{x_2} \wedge f_4.$$

x_1	x_2	x_3	$\overline{x_2}$	$\overline{x_3}$	f_1	f_2	f_3	f_4	f_5
0	0	0	1	1	0	0	1	1	1
0	0	1	1	0	1	1	0	1	1
0	1	0	0	1	0	1	0	1	0
0	1	1	0	0	1	1	0	1	0
1	0	0	1	1	1	0	1	0	0
1	0	1	1	0	1	1	0	1	1
1	1	0	0	1	1	1	0	1	0
1	1	1	0	0	1	1	0	1	0

Двоичная форма $F = 11000100$.
Наборы $N_f = (000, 001, 101)$, где $f(x_1, x_2, x_3) = 1$.
СДНФ функции $f(x_1, x_2, x_3) = (\overline{x_1} \wedge \overline{x_2} \wedge \overline{x_3}) \vee (\overline{x_1} \wedge \overline{x_2} \wedge x_3) \vee (x_1 \wedge \overline{x_2} \wedge x_3)$.
Наборы $N_f = (010, 011, 100, 110, 111)$, где $f(x_1, x_2, x_3) = 0$.
СКНФ функции

$$f(x_1, x_2, x_3) = (x_1 \vee \overline{x_2} \vee x_3) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3}).$$

б) Построим многочлен Жегалкина первым способом:
выписываем СДНФ функции

$$(\overline{x_1} \wedge \overline{x_2} \wedge \overline{x_3}) \vee (\overline{x_1} \wedge \overline{x_2} \wedge x_3) \vee (x_1 \wedge \overline{x_2} \wedge x_3);$$

заменяем знак дизъюнкции на знак суммы Жегалкина $\oplus$

$$(\overline{x_1} \wedge \overline{x_2} \wedge \overline{x_3}) \oplus (\overline{x_1} \wedge \overline{x_2} \wedge x_3) \oplus (x_1 \wedge \overline{x_2} \wedge x_3),$$

вынесем из первой и второй конъюнкции $(\overline{x_1} \wedge \overline{x_2})$:

$$(\overline{x_1} \wedge \overline{x_2}) \wedge (\overline{x_3} \oplus x_3) \oplus (\overline{x_1} \wedge \overline{x_2} \wedge x_3) = (\overline{x_1} \wedge \overline{x_2}) \oplus (x_1 \wedge \overline{x_2} \wedge x_3);$$

проделаем замены: $\overline{x_1} = x_1 \oplus 1$; $\overline{x_2} = x_2 \oplus 1$, получаем:

$$((x_1 \oplus 1) \wedge (x_2 \oplus 1)) \oplus ((x_1 \wedge (x_2 \oplus 1) \wedge x_3).$$

Далее раскроем скобки:

$$x_1 \wedge x_2 \oplus x_2 \oplus x_1 \oplus 1 \oplus x_1 \wedge x_2 \wedge x_3 \oplus x_1 \wedge x_3 = x_1 \wedge x_2 \wedge x_3 \oplus x_1 \wedge x_2 \oplus x_1 \wedge x_3 \oplus x_1 \oplus x_2 \oplus 1.$$

Итак, мы получили многочлен Жегалкина:

$$f(x_1, x_2, x_3) = x_1 \wedge x_2 \wedge x_3 \oplus x_1 \wedge x_2 \oplus x_1 \wedge x_3 \oplus x_1 \oplus x_2 \oplus 1.$$

Построим многочлен Жегалкина методом неопределенных коэффициентов, для этого составим следующие восемь уравнений:

$$f(0,0,0) = a_0 = 1; \quad a_0 = 1;$$
$$f(0,0,1) = a_0 \oplus a_3 = 1; \quad 1 \oplus a_3 = 1; \quad a_3 = 0;$$
$$f(0,1,0) = a_0 \oplus a_2 = 0; \quad 1 \oplus a_2 = 0; \quad a_2 = 1;$$
$$f(0,1,1) = a_0 \oplus a_2 \oplus a_3 \oplus a_{23} = 0; \quad 1 \oplus 1 \oplus 0 \oplus a_{23} = 0; \quad a_{23} = 0;$$
$$f(1,0,0) = a_0 \oplus a_1 = 0; \quad 1 \oplus a_1 = 0; \quad a_1 = 1;$$
$$f(1,0,1) = a_0 \oplus a_1 \oplus a_3 \oplus a_{13} = 1; \quad 1 \oplus 1 \oplus 0 \oplus a_{13} = 1; \quad a_{13} = 1;$$
$$f(1,1,0) = a_0 \oplus a_1 \oplus a_2 \oplus a_{12} = 0; \quad 1 \oplus 1 \oplus 1 \oplus a_{12} = 0; \quad a_{12} = 1;$$
$$f(1,1,1) = a_0 \oplus a_1 \oplus a_2 \oplus a_3 \oplus a_{12} \oplus a_{13} \oplus a_{23} \oplus a_{123} = 0;$$
$$1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus a_{123} = 0; \quad a_{123} = 1.$$

Составим многочлен Жегалкина:

$$f(x_1, x_2, x_3) = x_1 \wedge x_2 \wedge x_3 \oplus x_1 \wedge x_2 \oplus x_1 \wedge x_3 \oplus x_1 \oplus x_2 \oplus 1.$$

Задача 13. Проверьте на линейность функцию $f(x_1, x_2, x_3)$, если ее двоичный набор $F = 11100001$.

Решение. Применяем к функции $f(x_1, x_2, x_3)$ алгоритм проверки на линейность.

1. Вычисляем коэффициенты a_0, a_1, a_2, a_3) многочлена Жегалкина для данной функции:

$$a_0 = f(0,0,0) = 1; \quad a_1 = f(0,0,0) \oplus f(1,0,0) = 1 \oplus 0 = 1;$$
$$a_2 = f(0,0,0) \oplus f(0,1,0) = 1 \oplus 1 = 0 \quad a_3 = f(0,0,0) \oplus f(0,0,1) = 1 \oplus 1 = 0.$$

2. Вычисляем многочлен

$$\Phi(x_1, x_2, x_3) = a_0 \oplus a_1x_1 \oplus a_2x_2 \oplus a_3x_3 = x_1 \oplus 1.$$

Очевидно, что двоичный набор $F = 11110000$ многочлена $\Phi(x_1, x_2, x_3) = x_1 \oplus 1$ не совпадает с двоичным набором булевой функции, следовательно, функция $f(x_1, x_2, x_3)$ не линейна.

Задача 14. Докажите, что булева функция $f(x_1, x_2, x_3)$, заданная таблицей истинности линейна.

x_1	x_2	x_3	$f(x_1, x_2, x_3)$	$x_2 \oplus x_3 \oplus 1$
0	0	0	1	1
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	1	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Решение. Снова применим алгоритм определения линейности булевой функции.

1. Вычисляем коэффициенты a_0, a_1, a_2, a_3 многочлена Жегалкина функции $f(x_1, x_2, x_3)$:

$$a_0 = f(0,0,0) = 1;$$
$$a_1 = f(0,0,0) \oplus f(1,0,0) = 1 \oplus 1 = 0;$$
$$a_2 = f(0,0,0) \oplus f(0,1,0) = 1 \oplus 0 = 1;$$
$$a_3 = f(0,0,0) \oplus f(0,0,1) = 1 \oplus 0 = 1.$$

2. Таким образом, $\Phi(x_1, x_2, x_3) = x_2 \oplus x_3 \oplus 1$.

3. Достроим в таблице истинности последний столбик для $\Phi(x_1, x_2, x_3)$, напомним, что $0 \oplus 0 = 0$; $1 \oplus 0 = 1$; $1 \oplus 1 = 0$.

4. Столбики для $\Phi(x_1, x_2, x_3)$ и $f(x_1, x_2, x_3)$ совпали. Следовательно, функция $f(x_1, x_2, x_3)$ — линейная.

Задача 15. Задана булева функция трех переменных

$$f(x_1, x_2, x_3) = \overline{x_2} \wedge \left((x_1 \vee x_3) \mid \overline{(\overline{x_2} \mid \overline{x_3})}\right).$$

С помощью эквивалентных преобразований приведите функцию к ДНФ, КНФ, СДНФ, СКНФ.

Решение. Заменяем, $\overline{\overline{x_2} \mid \overline{x_3}} = \overline{\overline{\overline{x_2} \wedge \overline{x_3}}} = \overline{x_2} \wedge \overline{x_3}$,

$$(x_1 \vee x_3) \mid (\overline{x_2} \wedge \overline{x_3}) = \overline{(x_1 \vee x_3) \wedge (\overline{x_2} \wedge \overline{x_3})} = \overline{(x_1 \vee x_3)} \vee \overline{(\overline{x_2} \wedge \overline{x_3})} =$$
$$= (\overline{x_1} \wedge \overline{x_3}) \vee (\overline{\overline{x_2}} \vee \overline{\overline{x_3}}) = (\overline{x_1} \wedge \overline{x_3}) \vee (x_2 \vee x_3).$$

$f(x_1, x_2, x_3) = \overline{x_2} \wedge ((\overline{x_1} \wedge \overline{x_3}) \vee (x_2 \vee x_3))$, тогда

ДНФ $f(x_1, x_2, x_3) = (\overline{x_2} \wedge \overline{x_1} \wedge \overline{x_3}) \vee (x_2 \wedge \overline{x_2}) \vee (\overline{x_2} \wedge x_3)$,

КНФ $f(x_1, x_2, x_3) = \overline{x_2} \wedge (\overline{x_1} \vee x_2 \vee x_3) \wedge (\overline{x_3} \vee x_3 \vee x_2)$.

Строим СДНФ, для этого из ДНФ удаляем вторую конъюнкцию $x_2 \wedge \overline{x_2}$, а в третью конъюнкцию добавляем $x_1 \vee \overline{x_1}$, тогда:

$$(\overline{x_2} \wedge \overline{x_1} \wedge \overline{x_3}) \vee (\overline{x_2} \wedge x_3) \vee (x_1 \vee \overline{x_1}) = (\overline{x_2} \wedge \overline{x_1} \wedge \overline{x_3}) \vee (\overline{x_2} \wedge x_3 \wedge x_1) \vee (\overline{x_2} \wedge x_3 \wedge \overline{x_1}),$$

т. е. получили СДНФ функции

$$f(x_1, x_2, x_3) = (\overline{x_2} \wedge \overline{x_1} \wedge \overline{x_3}) \vee (\overline{x_2} \wedge x_3 \wedge x_1) \vee (\overline{x_2} \wedge x_3 \wedge \overline{x_1}).$$

Строим СКНФ, для этого из КНФ удаляем третью дизъюнкцию, а к первой добавляем $x_1 \wedge \overline{x_1}$:

$$(\overline{x_2} \vee (x_1 \wedge \overline{x_1})) \wedge (\overline{x_1} \vee x_2 \vee x_3) = (\overline{x_2} \vee x_1) \wedge (\overline{x_2} \vee \overline{x_1}) \wedge (\overline{x_1} \vee x_2 \vee x_3),$$

добавляем к первой и второй дизъюнкциям $x_3 \wedge \overline{x_3}$

$$((\overline{x_2} \vee x_1) \vee (x_3 \wedge \overline{x_3})) \wedge ((\overline{x_2} \vee \overline{x_1}) \vee (x_3 \wedge \overline{x_3}) \wedge (\overline{x_1} \vee x_2 \vee x_3)) =$$
$$= (\overline{x_2} \vee x_1 \vee x_3) \wedge (\overline{x_2} \vee x_1 \vee \overline{x_3}) \wedge (\overline{x_2} \vee \overline{x_1} \vee x_3) \wedge (\overline{x_2} \vee \overline{x_1} \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_2 \vee x_3).$$

Получили СКНФ функции

$$f(x_1, x_2, x_3) = (\overline{x_2} \vee x_1 \vee x_3) \wedge (\overline{x_2} \vee x_1 \vee \overline{x_3}) \wedge (\overline{x_2} \vee \overline{x_1} \vee x_3) \wedge (\overline{x_2} \vee \overline{x_1} \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_2 \vee x_3).$$

СДНФ и СКНФ проверить в задаче 12.

Контрольные вопросы

1. Что называется высказыванием?

2. Приведите пример высказываний. Какое высказывание называется истинным, а какое ложным?

3. Что называется составным высказыванием?

4. Перечислите виды логических операций над высказываниями и сформулируйте их определение.

5. Какие основные символы используются в теории высказываний?

6. Какие связки простейшие? Назовите другие связки.

7. Что такое таблица истинности высказывания и как она строится? Как еще называется эта таблица?

8. Какие существуют логические отношения между высказываниями?

9. Перечислите варианты импликации.

10. Сформулируйте основные законы алгебры высказываний. Как их доказать?

11. Что такое булева функция?

12. Как строится таблица истинности для булевых функций?

13. Что такое ДНФ и КНФ?

14. Дайте определение совершенного одночлена.

15. Приведите правило преобразования формул в СДНФ и СКНФ.

16. Как булевы функции связаны с формулами алгебры высказываний?

17. Дайте определение многочлена Жегалкина и сформулируйте теорему Жегалкина.

18. Сформулируйте первый алгоритм построения многочлена Жегалкина булевой функции.

19. В чем состоит метод неопределенных коэффициентов для построения многочлена Жегалкина?

20. Какой многочлен Жегалкина называется нелинейным?

21. Каков алгоритм определения линейности (нелинейности) булевой функции?

Часть 2

Множества и отображения

Лекции 5–7

1. Понятие множества

Любое понятие дискретной математики можно определить с помощью понятия множества. Под *множеством* понимают объединение в одно общее объектов, хорошо различаемых нашей интуицией или нашей мыслью. Таково интуитивное определение понятия множества, данное основателем теории множеств Георгом Кантором. Это понятие в математике является первичным и, следовательно, не имеет строгого определения. Объекты, составляющие множество, будем называть его элементами. Множества будем обозначать прописными буквами латинского алфавита: $A, B, C, \ldots$, а элементы множеств — строчными буквами $a, b, c, \ldots$. Основное отношение между элементом a и содержащим его множеством A, обозначается так: $a \in A$ (a принадлежит A). Если a не является элементом множества A, то пишут $a \notin A$ (a не принадлежит A). Некоторые множества имеют общепринятые обозначения: N — множество натуральных чисел; R — множество действительных чисел; Z — множество целых чисел.

2. Способы задания множеств

Имеется два существенно различных способа задания множеств. Можно либо указать правило для определения того, принадлежит или не принадлежит рассматриваемому множеству любой данный объект, либо дать полный перечень элементов этого множества.

Первый способ мы назовем описанием множества, а второй способ — перечислением множества. Например, обозначение $\{x \in U : \alpha(x)\}$ читается: «элементы множества U, обладающие свойством α» — это описание множества. Элементы перечисляемого множества принято заключать в скобки: $\{1, 2, 3, \dots\}$ — множество натуральных чисел; $\{2, 4, 6, \dots\}$ — множество четных чисел. Под многоточием в первом случае подразумеваются все последующие натуральные числа, а во втором — четные.

Нас часто будут интересовать множества логических возможностей, потому что анализ таких множеств часто играет основную роль при решении той или иной проблемы.

3. Подмножества

Множество, состоящее из некоторых элементов другого множества, называется подмножеством этого последнего множества. С целью изучения всех подмножеств данного множества введем следующую терминологию. Исходное множество будем называть *универсальным множеством*; подмножества, содержащие один элемент, будем называть *единичными множествами*; множество, вовсе не содержащее никаких элементов, будем называть *пустым множеством* и обозначать $\varnothing$.

В качестве примера возьмем универсальное множество U, состоящее из трех элементов $\{a, b, c\}$. Собственные подмножества U — это множества, которые содержат некоторые, но не все элементы U. Этими подмножествами являются три множества из двух элементов $\{a, b\}$, $\{a, c\}$, $\{b, c\}$ и три единичных множества $\{a\}$, $\{b\}$, $\{c\}$.

Будем считать подмножеством множества U и пустое множество $\varnothing$, не содержащее элементов U.

Другими словами, множество A называется подмножеством множества B (обозначаем $A \subset B$), если все элементы множества A принадлежат B. Это означает справедливость следующего утверждения: для любого элемента a, если $a \in A$, то $a \in B$ при условии $A \subset B$. Будем говорить также, что множество A содержится в B или имеется включение множества A в B. Множества A и B называются равными или совпадающими (обозначается $A = B$), если они состоят из одних и тех же элементов, т. е. если $A \subset B$ и $B \subset A$. Таким образом, чтобы доказать равенство множеств, требуется установить два включения.

4. Операции над множествами

В первой части были рассмотрены способы, которыми из данных высказываний могут быть образованы новые высказывания. Теперь мы будем рассматривать аналогичный процесс — образование новых множеств из данных множеств. Мы будем предполагать, что каждое из множеств, которое мы используем в этом процессе, является подмножеством некоторого универсального множества, и будем требовать, чтобы вновь образованное множество было подмножеством того же самого универсального множества. Как и всегда, мы можем задавать вновь образованное множество или путем описания, или путем перечисления.

Следует провести аналогию между логическими операциями и операциями над множествами.

отрицание	дополнение
конъюнкция	пересечение
дизъюнкция	объединение
импликация	разность

Чтобы нагляднее представить эти операции, изобразим их на диаграмме, называемой диаграммой Эйлера–Венна. Пусть прямоугольник обозначает универсальное множество, а круги внутри прямоугольника — подмножества.

Дополнением к множеству A называется множество элементов, которые не содержатся в A. Обозначают его $\overline{A}$ и читают «дополнение множества A к U» (см. рис. 2.1).

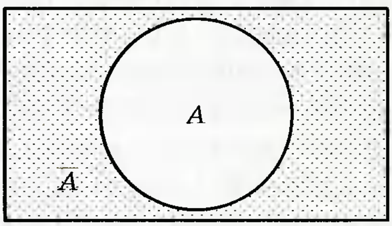


Рис. 2.1

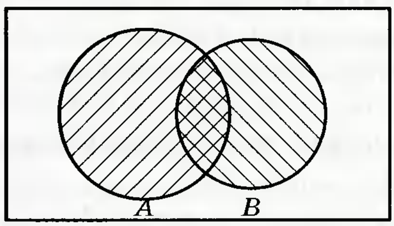


Рис. 2.2

Пересечением множеств A и B называется множество элементов, принадлежащих и A и B. Обозначают $A \cap B$ и читают «пересечение A и B» (см. рис. 2.2).

Если A и B — непустые множества, пересечение которых пусто, т. е. $A \cap B = \varnothing$, то их называют непересекающимися множествами.

Объединением множеств A и B называется множество элементов, принадлежащих либо A, либо B (либо обоим). Обозначают $A \cup B$ и читают «объединение A и B» (см. рис. 2.3).

Разностью множеств A и B называется множество элементов, принадлежащих A и не принадлежащих B. Обозначают $A \setminus B$ и читают «разность A и B» (см. рис. 2.4).

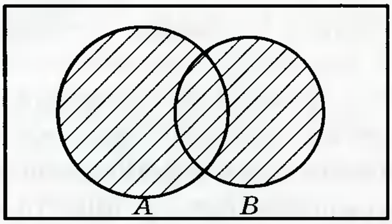


Рис. 2.3

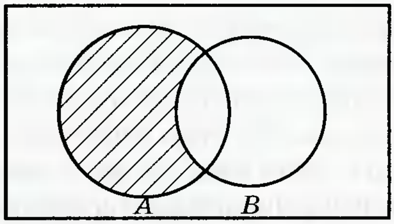


Рис. 2.4

5. Соотношение между множествами и составными высказываниями

Существует тесная связь между множествами — с одной стороны, и высказываниями — с другой, а также между операциями над множествами, с одной стороны, и операциями образования составных высказываний — с другой.

Если рассматривается несколько высказываний, то сопоставить каждому из этих высказываний некоторое множество можно вполне логичным путем. Сначала мы образуем множество всех логических возможностей для рассматриваемых высказываний и назовем его универсальным множеством. Затем каждому высказыванию мы поставим в соответствие подмножество тех логических возможностей универсального множества, для которых это высказывание истинно.

Определение. Пусть $X, Y, Z, \ldots$ означают некоторые высказывания, и пусть U — их множество логических возможностей. Пусть $A, B, C, \ldots$ означают подмножества U, для которых истинны соответственно высказывания $X, Y, Z, \ldots$. Тогда $A, B, C, \ldots$ называются соответственно множествами истинности высказываний $X, Y, Z, \ldots$.

Если X и Y — высказывания, то $X \vee Y$ и $X \wedge Y$ также высказывания, и, следовательно, они должны иметь множества истинности.

Чтобы найти множество истинности высказывания $X \vee Y$, заметим, что это высказывание истинно, когда истинно X или истинно Y (или оба). Таким образом, высказыванию $X \vee Y$ мы должны поставить в соответствие те логические возможности, которые лежат в A или в B (или в них обоих); иначе говоря, мы должны поставить в соответствие $X \vee Y$ множество $A \cup B$. С другой стороны, высказывание $X \wedge Y$ истинно, только когда истинно и

X и Y, так что высказыванию $X \wedge Y$ мы должны поставить в соответствие множество $A \cap B$.

Итак, существует тесная связь между логической операцией дизъюнкцией и операцией объединения множеств, а также между конъюнкцией и пересечением. А также между логической операцией отрицания и операцией дополнения множества, т. е. множеством истинности для $\overline{X}$ будет $\overline{A}$.

Множество истинности двух высказываний X и Y показаны на диаграмме Эйлера–Венна. Здесь отмечены различные логические возможности этих двух высказываний (см. рис. 2.5).

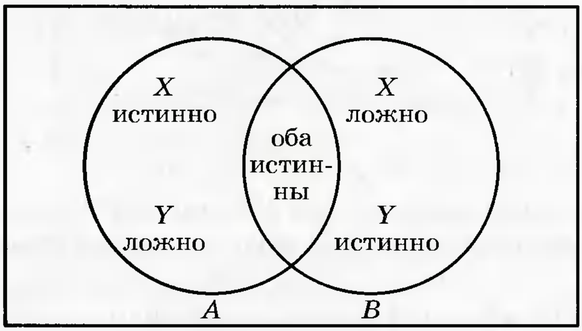


Рис. 2.5

Связь между высказыванием и его множеством истинности создает возможность «перевода» любой задачи, относящейся к составным высказываниям, в задачу теории множеств.

Возможно также и обратное: если поставлена какая-то задача, касающаяся множеств, то универсальное множество можно себе представить как некоторое множество логических возможностей, подмножества которого являются множествами истинности некоторых высказываний.

Следовательно, задачу, относящуюся к множествам, можно также «перевести» на язык составных высказываний.

6. Соотношения между высказываниями и соответствующими им множествами истинности

Мы рассмотрели такие множества истинности составных высказываний, которые образованы посредством связок $\vee$, $\wedge$, $\overline{}$. Все остальные связки можно определить через эти три основные и тем самым вывести, какие множества истинности им соответствуют. Например, известно, что импликация $X \to Y$ эквивалентна дизъюнкции $\overline{X} \vee Y$. Поэтому множество истинности для $X \to Y$ будет тем же, что и множество истинности для $\overline{X} \vee Y$, т. е. оно будет иметь вид $\overline{A} \cup B$.

На диаграмме Эйлера–Венна выделенная область показывает множество истинности этого высказывания (см. рис. 2.6).

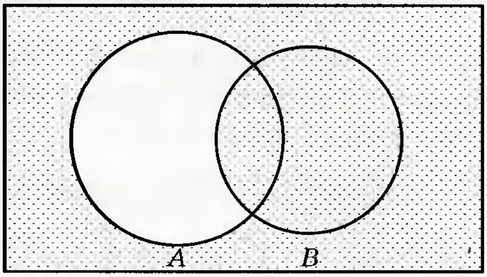


Рис. 2.6

Отметим, что незаштрихованная область на этой диаграмме показывает множество $A \setminus B = A \cap \overline{B}$, представляющее собой множество истинности высказывания $X \wedge \overline{Y}$. Поэтому заштрихованная область будет множеством $\overline{A \setminus B} = \overline{A \cap \overline{B}}$, которое является множеством истинности высказывания $\overline{X \wedge \overline{Y}}$. Таким образом, мы установили, что высказывания $X \to Y$, $\overline{X} \vee Y$, $\overline{X \wedge \overline{Y}}$ эквивалентны. Вообще, два высказывания эквивалентны тогда и только тогда, когда они имеют одни и те же множества истинности.

Заметим, что диаграммы Эйлера–Венна помогают обнаруживать отношения между высказываниями.

Предположим теперь, что X — логически истинное высказывание. Что представляет собой его множество истинности? Поскольку высказывание X логически истинно тогда и только тогда, когда оно истинно в каждом логически возможном случае, множеством истинности для высказывания X должно быть универсальное множество U.

Подобным же образом, если высказывание X логически ложно, то оно ложно в каждом логически возможном случае, и поэтому его множеством истинности будет пустое множество $\varnothing$.

Рассмотрим отношение следствия. Напомним, что из X следует Y тогда и только тогда, когда импликация $X \to Y$ логически истинна. Но высказывание $X \to Y$ тогда и только тогда логически истинно, когда его множество истинности совпадает с U, т. е. $(\overline{A \setminus B}) = U$ и $A \setminus B = \varnothing$. Но если $A \setminus B$ пусто, то B включает в себя A. Отношение включения обозначается, как мы отмечали, $A \subset B$ и читается «A является подмножеством B». Таким образом, высказывание $X \to Y$ логически истинно тогда и только тогда, когда $A \subset B$.

7. Выводы

Каждому высказыванию соответствует его множество истинности, каждой логической связке соответствует операция над множеством. Каждому отношению между высказываниями соответствует отношение между множествами истинности. Множествами истинности высказываний

$$X \vee Y; \quad X \wedge Y; \quad \overline{X} \quad \text{и} \quad X \to Y$$

служат соответственно:

$$A \cup B; \quad A \cap B; \quad \overline{A} \quad \text{и} \quad \overline{A \setminus B}.$$

Высказывание X логически истинно, если $A = U$, и логически ложно, если $A = \varnothing$. Высказывание X и Y эквивалентны тогда и только тогда, когда $A = B$; из X следует Y тогда и только тогда, когда $A \subset B$.

8. Абстрактные законы операций над множествами

Введенные операции над множествами подчинены некоторым очень простым абстрактным законам, которые будут перечислены в этом разделе.

Эти законы очень напоминают элементарные законы алгебры высказываний.

По этой причине множество, его подмножества и законы сочетания подмножеств образуют алгебраическую систему, называемую **булевой алгеброй.** Система составных высказываний, подчиняющаяся таким законам, тоже называется **булевой алгеброй.** Таким образом, любую из этих систем можно изучать или с алгебраической, или с логической точки зрения.

Ниже перечислены основные законы, действующие в булевых алгебрах.

Законы для объединения и пересечения:

1. $A \cup A = A$	**7.** $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
2. $A \cap A = A$	**8.** $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
3. $A \cup B = B \cup A$	**9.** $A \cup U = U$
4. $A \cap B = B \cap A$	**10.** $A \cap \varnothing = \varnothing$
5. $A \cup (B \cup C) = (A \cup B) \cup C$	**11.** $A \cap U = A$
6. $A \cap (B \cap C) = (A \cap B) \cap C$	**12.** $A \cup \varnothing = A$

Законы для дополнений:

1. $\overline{\overline{A}} = A$	**4.** $\overline{(A \cup B)} = \overline{A} \cap \overline{B}$
2. $A \cup \overline{A} = U$	**5.** $\overline{(A \cap B)} = \overline{A} \cup \overline{B}$
3. $A \cap \overline{A} = \varnothing$	**6.** $\overline{U} = \varnothing$

Законы для разностей множеств:

1. $A \setminus B = A \cap \overline{B}$	**6.** $A \setminus A = \varnothing$
2. $U \setminus A = \overline{A}$	**7.** $((A \setminus B) \setminus C) = A \setminus (B \cup C)$
3. $A \setminus U = \varnothing$	**8.** $A \setminus (B \setminus C) = (A \setminus B) \cup (A \cap C)$
4. $A \setminus \varnothing = A$	**9.** $A \cup (B \setminus C) = (A \cup B) \setminus (C \setminus A)$
5. $\varnothing \setminus A = \varnothing$	**10.** $A \cap (B \setminus C) = (A \cap B) \setminus (A \cap C)$

Доказательство каждого из перечисленных законов основано на определении равенства множеств и определений операций над множествами. Напомним, что множество A равно множеству B, если они состоят из одних и тех же элементов или оба пусты. Докажем один из законов для дополнений: $\overline{A \cup B} = \overline{A} \cap \overline{B}$.

Пусть $x \in \overline{A \cup B}$. По определению операции дополнения это означает, что $x \notin A \cup B$, но $x \in U$. Следовательно, $x \notin A$ и одновременно $x \notin B$. Таким образом, $x \in \overline{A}$ и $x \in \overline{B}$. Из определения операции пересечения получаем, что $x \in \overline{A} \cap \overline{B}$. Поэтому, учитывая произвольность элемента $x \in \overline{A \cup B}$, имеем $\overline{A \cup B} \subset \overline{A} \cap \overline{B}$.

Пусть теперь $x \in \overline{A} \cap \overline{B}$. Это значит, что $x \in \overline{A}$ и $x \in \overline{B}$. Таким образом, $x \notin A$ и $x \notin B$. Поэтому $x \notin A \cup B$. Следовательно, $x \in U \setminus (A \cup B) = \overline{(A \cup B)}$. Поскольку x — произвольный элемент из $\overline{A} \cap \overline{B}$, то окончательно получаем $\overline{A} \cap \overline{B} \subset \overline{A \cup B}$.

Приходим к выводу, что $\overline{A} \cap \overline{B} = \overline{A \cup B}$.

Третье практическое занятие по теме «Операции над множествами»

Задача 1. Для каких из следующих пар множеств имеет место одно из соотношений: $A \subset B$; $B \subset A$; $A = B$:

1) $A = \{a, b, c, d\}$; $B = \{a, c, d\}$, **2)** $A = \varnothing$; $B = \varnothing$,
3) $A = \varnothing$; $B = \{a, b, c\}$; $B = \{b, c, a\}$?

Задача 2. Даны множества $A = \{0, 1, 2, 3, 4, 5, 6, 7\}$; $B = \{3, 4, 5, 6, 7, 8, 9\}$; $C = \{-3, -2, -1, 0, 1, 2, 3, 4\}$; $D = \{2, 3, 4, 5, 6\}$. Задайте списками множества:

1) $A \cup B \cup C \cup D$; **2)** $A \cap B \cap C \cap D$; **3)** $(A \cap B) \cup (C \cap D)$; **4)** $(A \cup B) \cap (C \cup D)$; **5)** $(A \setminus B) \cup (B \setminus A)$.

Задача 3. Изобразите с помощью диаграмм Эйлера–Венна множества:

1) $A \subset B$ и $B \subset C$; **2)** $A \subset B$; $B \subset C$ и $A \setminus B = \varnothing$; **3)** $A \subset B$; $B \subset C$ и $C = A \cup B$;

4) $A \subset B$; $B \subset C$ и $A \cap B \neq \varnothing$; **5)** $(A \setminus B) \cup (B \setminus A)$.

Решение. Изобразим с помощью диаграммы Эйлера–Венна множество

$$(A \setminus B) \cup (B \setminus A)$$

(см. рис. 2.7).

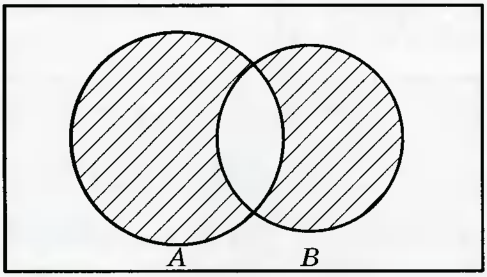


Рис. 2.7

Это множество является объединением двух разностей, называется симметрической разностью и обозначается $A \oplus B$, т. е. $(A \setminus B) \cup (B \setminus A) = A \oplus B$.

Задача 4. Опрос 100 студентов дал следующие результаты о количестве студентов, изучающих различные иностранные языки: испанский — 28; немецкий — 30; французский — 42; испанский и немецкий — 8; испанский и французский — 10; немецкий и французский — 5; все три языка — 3.

а) Сколько студентов не изучает ни одного языка?

б) Сколько студентов изучает один французский язык?

в) Сколько студентов изучает немецкий язык в том и только в том случае, если они изучают французский язык?

Решение. Нарисовать диаграмму Эйлера–Венна в виде трех кругов, обозначающих множество студентов, изучающих соответственно французский, немецкий и испанский языки. В каждую из восьми областей вписать данные, используя приведенные цифры. Начинать с конца списка и двигаться к началу.

Ответ: **а)** 20; **б)** 30; **в)** 38.

Задача 5. Следующий опрос 100 студентов (см. задачу 4) выявил следующие данные о числе студентов, изучающих различные иностранные языки: только немецкий — 18; немецкий, но не испанский — 23; немецкий и французский — 8; немецкий — 26; французский — 48; французский и испанский — 8; никакого языка — 24.

а) Сколько студентов изучают испанский язык?

б) Сколько студентов изучают немецкий и испанский языки?

в) Сколько студентов изучают французский язык, в том и только в том случае, если они не изучают испанский?

Ответ: **а)** 18; **б)** ни одного; **в)** 50.

Задача 6. В отчете об опросе 100 студентов (см. задачу 4) сообщалось, что количество студентов, изучающих различные языки, таково: все три языка — 5; немецкий и испанский — 10; французский и испанский — 8; немецкий и французский — 20; испанский — 30; немецкий — 23; французский — 50. Инспектор, представивший этот отчет, был уволен. Почему?

Задача 7. Докажите, пользуясь диаграммой Эйлера–Венна, что высказывание $X \vee (\overline{X} \vee Y)$ — логически истинно.

Решение. Этому высказыванию соответствует множество $A \cup (\overline{A} \cup B)$, отвечающая ему диаграмма изображена на рис. 2.8.

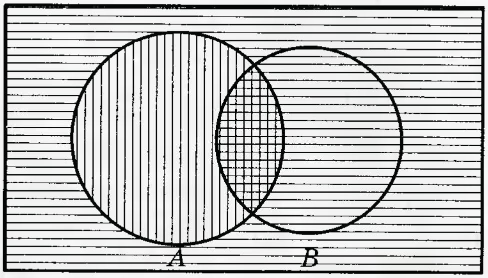


Рис. 2.8

Множество A заштриховано вертикальными линиями, а множество $\overline{A} \cup B$ — горизонтальными. Вся заштрихованная область является их объединением и совпадает с множеством U, так что наше составное высказывание логически истинно.

Задача 8. Воспользовавшись диаграммой Эйлера–Венна, определите, какие из следующих высказываний логически истинны:

а) $X \vee \overline{X}$; **б)** $X \wedge \overline{X}$; **в)** $X \vee (\overline{X} \wedge Y)$; **г)** $X \to (Y \to X)$; **д)** $X \wedge \overline{(Y \to X)}$.

Ответ: (а) и (г) — логически истинны, (б), (д) — логически ложны.

Задача 9. Докажите, пользуясь диаграммой Эйлера–Венна, что $X \vee (Y \wedge Z)$ эквивалентно $(X \vee Y) \wedge (X \vee Z)$.

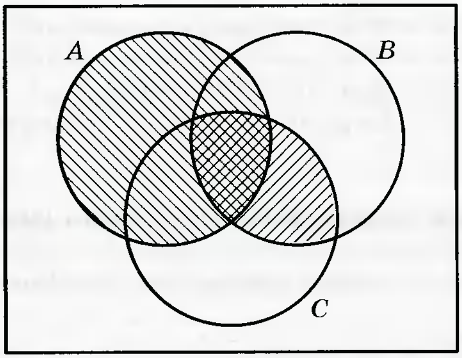
Решение. Множество истинности высказывания $X \vee (Y \wedge Z)$ совпадает со всей заштрихованной областью на диаграмме слева, а множество истинности высказывания $(X \vee Y) \wedge (X \vee Z)$ совпадает с дважды заштрихованной областью на диаграмме справа (см. рис. 2.9).

Так как эти множества совпадают, то наши два высказывания эквивалентны.

Задача 10. Покажите, пользуясь диаграммой Эйлера–Венна, что из Y следует $X \to Y$.

Решение. Множество истинности высказывания $X \to Y$ совпадает с заштрихованной областью на диаграмме (см. рис. 2.10).

Так как эта заштрихованная область включает в себя множество B, то мы видим, что из Y следует $X \to Y$.



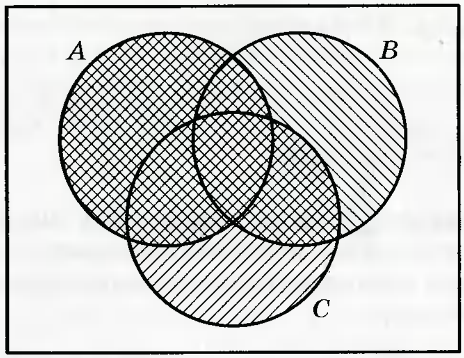


Рис. 2.9

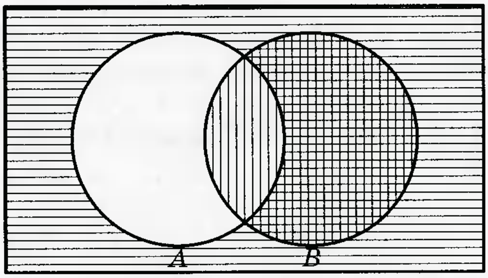


Рис. 2.10

Задача 11. Найдите множества истинности каждого высказывания и, воспользовавшись диаграммой Эйлера–Венна, определите, какие из выписанных ниже пар высказываний состоят из высказываний, одно из которых является следствием другого:

а) $X; X \wedge Y$; **б)** $X \wedge \overline{Y}; \overline{X} \to \overline{Y}$ **в)** $X \to Y; Y \to X$; **г)** $X \wedge Y; X \wedge \overline{Y}$.

Задача 12. Три или более высказывания называются несовместимыми, если они не могут быть истинными все сразу. Что можно сказать о множествах истинности таких высказываний?

Задача 13. Для следующих трех составных высказываний:

а) введите буквенные обозначения для компонент;

б) дайте символическое выражение;

в) найдите множества истинности;

г) проверьте их совместимость.

Если этот курс интересен, то я буду упорно над ним работать. Если этот курс не интересен, то я получу по нему плохую отметку. Я не буду упорно работать, но получу по этому курсу хорошую отметку.

Ответ: Несовместимы.

Задача 14. Каждому множеству поставьте в соответствие высказывание, имеющее это множество своим множеством истинности, и, воспользовавшись таблицами истинности, определите, какие из следующих множеств пусты:

а) $(A \cup B) \cap (\overline{A} \cup \overline{B})$; б) $(A \cap B) \cup (\overline{B} \cup C)$; в) $(A \cap B) \setminus A$; г) $(A \cup C) \cap (\overline{A} \cup \overline{B})$.

Ответ: (б) и (в).

Задача 15. Каждому множеству поставьте в соответствие высказывание, имеющее это множество своим множеством истинности, и, воспользовавшись таблицами истинности, определите, являются ли попарно различными следующие множества:

а) $A \cap (B \cup C)$; б) $(C \setminus B) \cup (B \cup C)$; в) $(C \cup B) \cap \overline{(\overline{C} \cup \overline{B})}$; г) $(A \cap B) \cup (A \cap C)$; д) $(A \cap B \cap \overline{C}) \cup (A \cap \overline{B} \cap C) \cup (\overline{A} \cap B \cap \overline{C}) \cup (\overline{A} \cap \overline{B} \cap C)$.

Задача 16. Каждому множеству поставьте в соответствие высказывание, имеющее это множество своим множеством истинности, и, воспользовавшись таблицами истинности, определите, в каких из следующих пар множеств одно из множеств является подмножеством другого:

а) $A; A \cap B$; б) $A \cap \overline{B}; B \cap \overline{A}$; в) $A \setminus B; B \setminus A$; г) $\overline{A} \cap \overline{B}; A \cup B$.

Задача 17. Докажите, как с помощью таблиц истинности, так и с помощью диаграммы Эйлера–Венна, что высказывание $X \wedge (Y \vee Z)$ эквивалентно высказыванию $(X \wedge Y) \vee (X \wedge Z)$.

Задача 18. Проверьте все законы операций над множествами для объединения и пересечения с помощью диаграмм Эйлера–Венна. Переведите эти законы в законы для составных высказываний. Проверьте их с помощью таблиц истинности.

Задача 19. Проверьте все законы операций над множествами для дополнений и для разностей с помощью диаграмм Эйлера–Венна, переведите их в законы для составных высказываний и проверьте их с помощью таблиц истинности.

Задача 20. Из законов булевой алгебры над множествами получите следующие результаты:

а) $A = (A \cap B) \cup (A \cap \overline{B})$; б) $A \cup B = (A \cap B) \cup (A \cap \overline{B}) \cup (\overline{A} \cap B)$; в) $A \cap (A \cap B) = A$; г) $A \cup (\overline{A} \cap B) = A \cup B$.

9. Кортежи и декартово произведение множеств

Определение. Пусть даны множества $X_1, X_2, ..., X_n$. Кортежем длины n, составленным из элементов этих множеств, называется конечная последовательность $\alpha = \langle x_1, x_2, ..., x_n \rangle$, где для всех k, $1 \leqslant k \leqslant n$, имеем $x_k \in X_k$.

Элемент x_k называется k-й координатой или k-й компонентой кортежа α.

Два кортежа равны в том и только в том случае, когда они имеют одинаковую длину, причем их координаты, стоящие на местах с одинаковыми номерами, равны, т. е. кортежи $\alpha = \langle x_1, \ldots, x_m \rangle$ и $\beta = \langle y_1, \ldots, y_n \rangle$ равны только в том случае, когда $m = n$, причем $x_k = y_k$ для всех $1 \leqslant k \leqslant n$.

Кортежи длины два называют упорядоченными парами, длины три — упорядоченными тройками, ..., длины n — упорядоченными n-ками. Для краткости речи слово «упорядоченные» часто опускают.

Кортеж, не содержащий ни одной координаты, т. е. кортеж длины 0, называется пустым.

Основные отличия понятий кортежа и множества следующие:

а) в множестве порядок элементов не играет роли, а кортежи, отличающиеся порядком элементов, различны, даже в случае, когда они имеют одинаковый состав;

б) в множестве все элементы различны, а в кортеже координаты могут повторяться.

В дальнейшем, чтобы различать множества и кортежи, будем элементы множества заключать в фигурные скобки, а координаты кортежа — в угловые.

Пусть $A_1, A_2, \ldots, A_n$ — некоторые множества. Их декартовым произведением называют множество, состоящее из кортежей вида $\langle a_1, a_2, \ldots, a_n \rangle$, где $a_1 \in A_1$; $a_2 \in A_2$; ...; $a_n \in A_n$. Декартово произведение обозначается так: $A_1 \times A_2 \times \ldots \times A_n$.

Произведение

$$\underbrace{A \times A \times \ldots \times A}_{n \text{ раз}}$$

сокращенно обозначается как A^n и называется декартовой n-й степенью множества A.

10. Бинарные отношения

Пусть A и B два конечных множества. Декартовым произведением множеств A и B называют множество $A \times B$, состоящее из всех упорядоченных пар $\langle a, b \rangle$, где $a \in A$, $b \in B$.

Бинарным отношением между элементами множеств A и B называется любое подмножество R множества $A \times B$, т. е. $R \subset A \times B$.

По определению, бинарным отношением называется множество пар. Если R — бинарное отношение (т. е. множество пар), то говорят, что параметры x и y связаны бинарным отношением R, если пара $\langle x, y\rangle$ является элементом R, т. е. $\langle x, y\rangle \in R$.

Высказывание: «предметы x и y связаны бинарным отношением R» записывают в виде $x\,R\,y$.

Таким образом, $x\,R\,y \leftrightarrow \langle x, y\rangle \in R$.

Если $R \subset A \times A$, то говорят, что бинарное отношение определено на множестве A.

Областью определения бинарного отношения R называется множество, состоящее из таких x, для которых $\langle x, y\rangle \in R$ хотя бы для одного y.

Область определения бинарного отношения будем обозначать δ_R.

Областью значений бинарного отношения R называется множество всех y, для которых $\langle x, y\rangle \in R$ хотя бы для одного x.

Область значений бинарного отношения будем обозначать ρ_R.

Рассмотрим специальные бинарные отношения:

а) Бинарное отношение R на непустом множестве A называется **рефлексивным**, если $\langle x, x\rangle \in R$ для всех $x \in A$, и **иррефлексивным**, если $\langle x, x\rangle \notin R$ для всех $x \in A$.

б) Бинарное отношение R на непустом множестве A называется **симметричным**, если $\langle x, y\rangle \in R \Rightarrow \langle y, x\rangle \in R$, и **антисимметричным**, если $\langle x, y\rangle \in R$ и $\langle y, x\rangle \in R \Rightarrow x = y$.

в) Бинарное отношение R на непустом множестве A называется **транзитивным**, если $\langle x, y\rangle \in R$ и $\langle y, z\rangle \in R \Rightarrow \langle x, z\rangle \in R$.

Рефлексивное, транзитивное и симметричное бинарное отношение R на множестве A называется **эквивалентностью** на A.

11. Отображение множеств

Соответствие f, сопоставляющее каждому элементу x множеств X один и только один элемент множества Y, называется **отображением** множества X во множество Y.

Элемент множества Y, соответствующий при отображении f элементу x из X, обозначают $f(x)$ и называют **образом** элемента x при этом отображении.

Если $f(x) = y$, то элемент x называют **прообразом** элемента y при отображении f.

Совокупность всех прообразов элемента y при отображении f называется **полным прообразом** этого элемента и обозначается $f^{-1}(y)$, т. е. $f^{-1} = \{x : f(x) = y\}$.

Правая часть читается: «совокупность таких x, что $f(x) = y$».

Каждому подмножеству A множества X ($A \subset X$) соответствует его образ $f(A)$ при отображении f. Этот образ состоит из всех элементов множества Y, которые являются образами какого-нибудь элемента из A: $f(A) = \{y : y = f(a), a \in A\}$.

Каждому подмножеству B множества Y ($B \subset Y$) соответствует его полный прообраз $f^{-1}(B)$ при отображении f. Он состоит из всех элементов, образы которых принадлежат B: $f^{-1}(B) = \{x : f(x) \in B\}$.

Множество A называется областью определения отображения f, а множество $f(A)$ называется множеством значений этого отображения.

Частный случай отображения множества X в множество Y имеет место, если каждый элемент множества Y имеет прообраз. В этом случае отображение f называется **сюръективным.**

Если для каждого элемента $y \in Y$ существует не более одного прообраза, т. е. $f(x_1) \neq f(x_2)$ для любых $x_1, x_2 \in X$, если $x_1 \neq x_2$, то отображение f называется **инъективным.**

Если отображение f сюръективно и инъективно, то оно называется **биективным** (взаимно-однозначным).

С точки зрения отображений два множества называются количественно-эквивалентными, если между ними можно установить биективное отображение.

Если между элементами множеств A и B существует биективное отображение, то множества A и B называются равномощными.

Для конечных множеств A и B понятие равномощности означает, что они имеют одно и то же число элементов. Таким образом, если A — конечное множество, содержащее n элементов, то **мощностью множества** A называется число n и обозначается $|A|$, т. е. $|A| = n$.

Очевидно, что отношение равномощности является отношением эквивалентности, поэтому **равномощные** множества часто называют **эквивалентными.**

12. Функции

В основе всех разделов дискретной математики лежит понятие функции.

Функцией называется бинарное отношение f, если из $\langle x, y\rangle \in f$ и $\langle x, z\rangle \in f$ следует, что $y = z$.

Две функции f и g равны, если они состоят из одних и тех же элементов. Область определения функции и область ее значений задается так же, как и для бинарных отношений. Если область определения $\delta_f = X$ и область значений $\rho_f \subset Y$, то говорят, что функция f задана на множестве X со значениями во множестве Y, при этом f отображает множество X во множество Y. Это отображение обозначается как $f : X \to Y$.

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

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

Пусть $f : X \to Y$. Функция f называется инъективной, если для любых x_1, x_2, y из $y = f(x_1)$ и $y = f(x_2)$ следует, что $x_1 = x_2$.

Функция f называется сюръективной, если для любого элемента $y \in Y$ существует элемент $x \in X$ такой, что $y = f(x)$.

Функция f называется биективной, если f одновременно сюръективна и инъективна.

Если существует биективная функция $f : X \to Y$, то говорят, что f осуществляет взаимно-однозначное соответствие между множествами X и Y.

Если $f : X \to Y$, а $g : Y \to Z$, то функция $F : X \to Z$, определенная для каждого $x \in X$ формулой $F(x) = g(f(x))$, называется композицией (суперпозицией) функции f и g, или **сложной функцией.**

Пусть задана функция $f : X \to Y$ и ρ_f — множество ее значений. Совокупность всевозможных упорядоченных пар вида $\langle y, f^{-1}(y)\rangle$, $y \in \rho_f$ образует функцию, которая называется **обратной функцией** для функции f и обозначается f^{-1}.

Обратная функция f^{-1} ставит в соответствие каждому элементу $y \in \rho_f$ его прообраз $f^{-1}(y)$, т. е. некоторое множество элементов. Заметим, что для того, чтобы f^{-1} являлась функцией, достаточно, чтобы функция f была инъективной.

Четвертое практическое занятие по теме «Отношения. Отображения. Функции»

Задача 1. Из множеств $\{a, b, c\}$ и $\{1, 2\}$ составьте кортежи.
Решение. Из данных множеств можно составить 6 кортежей длины 2. $\langle a, 1\rangle$; $\langle a, 2\rangle$; $\langle b, 1\rangle$; $\langle b, 2\rangle$; $\langle c, 1\rangle$; $\langle c, 2\rangle$.

Задача 2. Сравните кортежи:

а) $\langle 1^2, 2^2, 3^2\rangle$ и $\left\langle \sqrt{1}, \sqrt{16}, \sqrt{81}\right\rangle$; **б)** $\langle 1, 2, 3\rangle$ и $\langle 3, 1, 2\rangle$;

в) $\langle 1, 2, 3\rangle$ и $\langle 1, 2, 3, 4\rangle$.

Решение. **а)** Кортежи $\langle 1^2, 2^2, 3^2\rangle$ и $\left\langle\sqrt{1}, \sqrt{16}, \sqrt{81}\right\rangle$ равны, поскольку $1^2 = \sqrt{1}$; $2^2 = \sqrt{16}$; $3^2 = \sqrt{81}$;

б) кортежи $\langle 1, 2, 3\rangle$ и $\langle 3, 1, 2\rangle$ различны. Хотя имеют одинаковую длину и одно и то же множество координат, но эти координаты располагаются в разном порядке;

в) кортежи $\langle 1, 2, 3\rangle$ и $\langle 1, 2, 3, 4\rangle$ различны, так как имеют разную длину.

Задача 3. Равны ли следующие кортежи:

1) $\langle a, \{a, b, c\}, b, c\rangle$ и $\langle a, \{a, b, c\}, \{b, c\}\rangle$;
2) $\langle a, \{a, b, c\}, b, c\rangle$ и $\langle a, \{a, b, c\}, b, c\rangle$;
3) $\langle a, \{a, b, c\}, b, c\rangle$ и $\langle a, \{a, b, c\}, c, b\rangle$;
4) $\langle a, \{a, b, c\}, b, c\rangle$ и $\langle a, \{a, b, c\}, a, b, c\rangle$?

Задача 4. Пусть $A = \{1, 2, 3\}$, $B = \{x, y\}$.

Выписать все элементы декартова произведения $A \times B$ и $B \times A$.
Решение.

$$A \times B = \{\langle 1, x\rangle ; \langle 2, x\rangle ; \langle 3, x\rangle ; \langle 1, y\rangle ; \langle 2, y\rangle ; \langle 3, y\rangle\},$$
$$B \times A = \{\langle x, 1\rangle ; \langle x, 2\rangle ; \langle x, 3\rangle ; \langle y, 1\rangle ; \langle y, 2\rangle ; \langle y, 3\rangle\}.$$

Задача 5. Пусть $A = \{1, 2\}$. Выписать все элементы декартова произведения $A \times A$.
Решение. $A \times A = \{\langle 1, 1\rangle ; \langle 1, 2\rangle ; \langle 2, 1\rangle ; \langle 2, 2\rangle\}$.

Задача 6. Из цифр 1, 2, 3, 4, 5 составьте все двузначные числа. Как связано получившееся множество с декартовым произведением $A \times A$, где $A = \{1, 2, 3, 4, 5\}$?

Задача 7. Рассмотрим два множества $A = \{a, b, c, d, e, f, g, h\}$ и $B = \{1, 2, 3, 4, 5, 6, 7, 8\}$.

Составьте множество пар $\langle x, y\rangle \in A \times B$. Что это множество представляет?
Ответ: множество клеток шахматной доски.

Задача 8. Найдите правую и левую область отношения $R = \{\langle 1, 5\rangle ; \langle 1, 6\rangle ; \langle 1, 7\rangle\}$.
Решение. $D_{\text{пр.}} = \{5, 6, 7\}$; $D_{\text{лев}} = \{1\}$.

Задача 9. Если $A = \{2, 3, 4, 5, 6, 7, 8\}$, запишите бинарное отношение $R = \{\langle x, y\rangle$: $x, y \in A$, x делит y, и $x \leqslant 3\}$.
Решение. $R = \{\langle 2, 2\rangle ; \langle 2, 4\rangle ; \langle 2, 6\rangle ; \langle 2, 8\rangle ; \langle 3, 3\rangle ; \langle 3, 6\rangle\}$.

Задача 10. Рассмотрим множество шахматных клеток $S = A \times B$ из задачи 7. Определите бинарное отношение R: ход ладьи на множестве S по следующему правилу: $\langle c_1, c_2\rangle \in R$ тогда и только тогда, когда $\langle c_1, c_2\rangle \in S$ и ладья может пройти с клетки c_1 на клетку c_2 одним ходом на пустой доске (напомним, что ладья за один ход может изменить либо горизонтальную координату, либо вертикальную, но не обе координаты одновременно).
Решение. $R = \{\langle c_1, c_2\rangle$: $c_1 = \langle s_1, t_1\rangle$; $c_2 = \langle s_2, t_2\rangle$; $s_1, s_2 \in A$; $t_1, t_2 \in B$ и $\langle s_1 = s_2$ и $t_1 \neq t_2\rangle$ или $\langle s_1 = s_2$ и $t_1 = t_2\rangle\}$.

Задача 11. Каждому алгебраическому уравнению ставится в соответствие его степень. Укажите множество значений для этого отображения.
Ответ: Множество N.

Задача 12. Пусть X — множество пальто в гардеробе, Y — множество крючков. В каком случае отображение множества пальто X в множество крючков Y будет инъективным, сюръективным, биективным?
Решение. Отображение, при котором каждому пальто сопоставляется крючок, на котором оно висит, является инъективным, если на каждом крючке висит не более одного пальто (некоторые крючки могут быть пустыми).

Отображение является сюръективным, если на каждом крючке висит хотя бы одно пальто (на некоторых крючках может быть несколько пальто).

Отображение является биективным, если на каждом крючке висит ровно одно пальто.

Заметим, что если существует биективное отображение конечного множества A в конечное множество B, то в множествах A и B поровну элементов. Если же существует инъективное отображение конечного множества A в конечное множество B, то можно сказать, что в B не меньше элементов, чем в A.

Если же в A больше элементов, чем в B, то не существует инъективного отображения A в B.

В случае, когда существует сюръективное отображение A в B, то в A не меньше элементов, чем в B.

Задача 13. Среди следующих отображений укажите сюръективные отображения:

1) X — множество кругов, Y — множество действительных чисел, каждому кругу сопоставляется его площадь;

2) X — множество кругов, Y — множество положительных действительных чисел, каждому кругу сопоставляется его площадь;

3) $X = \{x : \ -3 \leqslant x \leqslant 5\}$, $Y = R$, $f : x \to x^2$ (R — множество действительных чисел);

4) $X = \{x : \ -3 \leqslant x \leqslant 5\}$, $Y = \{x : \ 0 \leqslant x \leqslant 25\}$, $f : x \to x^2$.

Ответ: **2)** и **4)**.

Задача 14. Является ли отображением соответствие «Столицей государства X является город Y»?
Ответ: Да.

Задача 15. Являются ли следующие отношения функциями:

(1) $\{\langle 1, 2\rangle ; \langle 2, 3\rangle ; \langle 3, 2\rangle\}$; **(2)** $\{\langle 1, 2\rangle ; \langle 1, 3\rangle ; \langle 2, 3\rangle\}$;

(3) $\{x, x^2 - 2x - 3 : x \in R\}$?

Решение. Отношение **(1)** — функция; отношение **(2)** — не является функцией; отношение **(3)** — функция, которая обычно обозначается $y = x^2 - 2x - 3$.

Задача 16. Является ли отношение $\{\langle 1, a\rangle ; \langle 1, b\rangle ; \langle 2, a\rangle\}$, определенное на декартовом произведении множеств $A = \{1, 2\}$ и $B = \{a, b\}$, функцией?

Решение. Не является, так как элементу $1 \in A$ ставятся в соответствие различные элементы $a, b \in B$.

Задача 17. Является ли функция $f(x) = x^2$ инъективной?
Решение. Не является, так как существуют такие различные значения x_1 и x_2, для которых $f(x_1) = f(x_2)$. Например, пусть $x_1 = 1$ и $x_2 = -1$. Тогда $f(x_1) = 1^2 = (-1)^2 = f(x_2)$.

Задача 18. Является ли отношение $\{\langle a_1, b_1\rangle ; \langle a_2, b_2\rangle ; \langle a_3, b_2\rangle\}$, определенное на декартовом произведении множеств $A = \{a_1, a_2, a_3\}$ и $B = \{b_1, b_2, b_3\}$, функцией?

Если да, то является ли данная функция сюръекцией, инъекцией?
Решение. Поскольку каждому элементу множества A соответствует единственный элемент множества B, то можно утверждать, что данное отношение является функцией. Данная функция не является инъекцией, поскольку существуют различные элементы множества A, которым соответствует один и тот же элемент из B. Так, различным элементам a_2 и a_3 соответствует элемент b_2. Данная функция также не является сюръекцией, поскольку элемент b_3 не входит ни в одну упорядоченную пару.

Задача 19. Является ли отношение $\{\langle 1, 4\rangle ; \langle 2, 3\rangle ; \langle 3, 2\rangle ; \langle 4, 1\rangle\}$, заданное на декартовом квадрате множества $A = \{1, 2, 3, 4\}$, биективным отображением?
Решение. Сначала проверим, является ли данное отношение отображением. Так как каждый элемент x из A входит в пары вида $\langle x, y\rangle$ лишь один раз, можно утверждать, что мы имеем дело с отображением. Это отображение инъективно, поскольку для различных первых элементов пар вторые элементы этих пар также различны. Отображение является сюръекцией, так как правая часть отношения совпадает с множеством A. Раз отношение является инъективным и сюръектиным, следовательно, оно является биективным.

Задача 20. Отношение R на множестве всех книг библиотеки определили следующим образом. Пара книг a и b принадлежат R, если и только если в этих книгах есть ссылка на одни и те же литературные источники. Является ли R

а) рефлексивным отношением;

б) симметричным отношением;

в) транзитивным отношением?

Решение. Отношение R рефлексивно, раз две одинаковые книги содержат ссылки на одни и те же литературные источники. Данное отношение также симметрично, так как если в книгах a и b имеется ссылка на какой-либо литературный источник, то в книгах b и a, очевидно, есть ссылки на тот же литературный источник.

Свойство транзитивности, вообще говоря, не выполняется, поскольку возможны случаи, когда книги a и b содержат ссылки на общие литературные источники, книги b и c также ссылаются на общие литературные источники, а книги a и c не имеют общих ссылок. Отсюда можно сделать вывод, что для данного отношения не выполняются условия эквивалентности.

Задача 21. (Для самостоятельного решения.)

Пусть отношение R задано на декартовом произведении множеств K и P, где K — множество ключевых слов, а P — множество Web-страниц. Пара $\langle x, y\rangle$ принадлежит R, если и только если ключевое слово x содержится на странице y. Является или нет R функцией? Объясните почему.

Ответ: Нет.

Задача 22. (Для самостоятельного решения.)

Пусть отношение R задано на декартовом произведении множеств B и P, где B — множество всех книг в книжном магазине, P — множество цен. Пара $\langle x, y\rangle$ принадлежит R, если и только если книга x имеет цену y. Является ли R функцией? Если да, то является ли данная функция сюръективной, инъективной?

Ответ: Не инъективной, не сюръективной не является.

Задача 23. (Для самостоятельного решения.)

Пусть отношение R задано на декартовом произведении множеств D и N, где D — множество всех документов, содержащихся в папке «Входящие», а N — множество всех номеров, служащих для регистрации этих документов. Объясните, почему данное отношение является функцией и притом биективной.

Задача 24. Какая из указанных функций $f : [0, 1] \to [0, 3]$

а) $x \to 3\sin\dfrac{\pi x}{2}$; б) $x \to 3^x$; в) $x \to 12\left(x - \dfrac{1}{2}\right)^2$

инъективна, сюръективна или биективна?

Решение. а) Для произвольного $y \in [0, 3]$ уравнения $y = 3\sin\dfrac{\pi x}{2}$ имеем единственное решение

$$x = \frac{2}{\pi}\arcsin\frac{y}{3},$$

принадлежащее $[0, 1]$, поэтому функция $x \to 3\sin\dfrac{\pi x}{2}$ является биективной;

б) Если $y \in [0, 3]$, то уравнение $y = 3^x$ имеет не более одного решения $x \in [0, 1]$. При $y \in [1, 3]$ решением является $x = \log_3 y$, а при $y \in [0, 1]$ решений нет. Следовательно, $y = 3^x$ — инъективна;

в) Из уравнения

$$y = 12\left(x - \frac{1}{2}\right)^2, \quad y \in [0, 3],$$

находим

$$x_1 = \frac{1}{2} - \frac{1}{2}\sqrt{\frac{y}{3}}, \quad x_2 = \frac{1}{2} + \frac{1}{2}\sqrt{\frac{y}{3}},$$

причем, если $0 \leqslant y \leqslant 3$, то оба корня принадлежат $]0, 1]$; если $y = 0$, то корни совпадают $x_1 = x_2 = \dfrac{1}{2}$ и принадлежат $[0, 1]$. Следовательно, для всех $y \in [0, 3]$ уравнение $y = 12\left(x - \dfrac{1}{2}\right)^2$ на $[0, 1]$ имеет хотя бы одно решение. Поэтому рассматриваемая функция сюръективна.

Контрольные вопросы

1. Какие основные символы, используемые в теории множеств, вы знаете?

2. Что такое множество? Как его обозначить? Как можно задать множество? Что такое подмножество?

3. Какие основные операции выполняются над множествами?

4. Какое множество можно назвать универсальным?

5. Что такое диаграмма Эйлера–Венна? Проиллюстрируйте с помощью диаграммы Эйлера–Венна объединение и пересечение трех множеств.

6. Каковы соотношения между множествами и составными высказываниями?

7. Сформулируйте и докажите основные тождества алгебры множеств.

8. Что называется кортежем и какие кортежи называются равными?

9. Что такое: декартово произведение множеств; декартова степень некоторого множества A; бинарное отношение, заданное на множестве A?

10. Назовите основные свойства бинарных отношений. Какое отношение называется рефлексивным, симметричным, антисимметричным, транзитивным? Какое отношение называется отношением эквивалентности?

11. Дайте определение отображения множества A во множество B. Поясните термин «мощность множества».

12. Что такое сюръекция, инъекция, биекция?

13. Дайте определение функции.

Часть 3

Элементы комбинаторного анализа

Лекция 8

На практике часто встречаются задачи, где необходимо подсчитать число всех возможных способов размещения некоторых предметов конечного множества или число всех возможных способов выполнения определенного действия из конечного множества таких действий. Задачи такого типа называются **комбинаторными**, а методы их решения — **методами комбинаторного анализа.** Поскольку комбинаторика имеет дело с конечными множествами, то ее часто называют **теорией конечных множеств.**

Таким образом, комбинаторика — это раздел дискретной математики, посвященный решению задач выбора и расположения элементов некоторого конечного множества в соответствии с заданными свойствами.

1. Основные правила комбинаторики

Мы уже рассматривали задачу: подсчет числа элементов в декартовом произведении множеств $\{1, 2, 3\} \times \{x, y\}$. Число таких элементов, как мы видели, равно произведению числа элементов первого множества на число элементов второго множества, т. е. в нашем случае это $3 \cdot 2 = 6$.

За этой простой задачей стоит правило, которое называется первым основным правилом комбинаторики: правило произведения.

Пусть необходимо выполнить последовательно k действий. Если первое действие можно выполнить n_1 способами, второе n_2 способами и так далее до k — действия, которое можно выполнить n_k способами, то все k действий можно выполнить $n_1 \cdot n_2 \cdot \ldots \cdot n_k$ способами.

Другим часто применяемым в комбинаторике правилом является правило суммы. Это правило формулируется следующим образом: если элемент x может быть выбран m способами, а элемент y — другими n способами, то выбор «либо x либо y» может быть осуществлен $m + n$ способами.

2. Перечислительная комбинаторика или теория перечислений

Будем рассматривать задачи, связанные с нахождением числа способов построения кортежей из элементов конечного множества. Простейшими такими кортежами являются размещения, перестановки, сочетания. Эти задачи образуют часть комбинаторики, называемой перечислительной комбинаторикой или теорией перечислений.

Пусть A — конечное множество, состоящее из n элементов $|A| = n$.

а) *Размещения.* Кортежи длины k $(1 \leqslant k \leqslant n)$, состоящие из различных элементов n-элементного множества A (кортежи отличаются один от другого как самими элементами, так и их порядком), называются размещениями из n элементов множества A по k. Число таких размещений будем обозначать A_n^k (буква A от французского слова *arrangement* — размещение). Схема выбора состоит в выборе k элементов из n-элементного множества без возвращения.

Тогда необходимо совершить k действий, причем первое действие можно совершить n способами, второе $(n-1)$ способами и т. д., k-е действие $n-(k-1)$ способами. Согласно комбинаторному правилу умножения, получим формулу: $A_n^k = n \cdot (n-1) \cdot \ldots \cdot (n-k+1)$.

Если умножить и разделить полученное выражение на $(n-k)!$, получим:

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

где $n! = n \cdot (n-1) \cdot \ldots \cdot 3 \cdot 2 \cdot 1$ и называется факториалом числа n (читается n-факториал). Причем: $0! = 1$; $1! = 1$; $2! = 1 \cdot 2 = 2$; $3! = 1 \cdot 2 \cdot 3 = 6$; $4! = 1 \cdot 2 \cdot 3 \cdot 4 = 24$; $5! = 4! \cdot 5 = 120\ldots$.

Пусть, например, дано множество $A\{1, 3, 5\}$. Выпишем все размещения из трех элементов по два: $\langle 1, 3\rangle$, $\langle 1, 5\rangle$, $\langle 3, 5\rangle$, $\langle 3, 1\rangle$, $\langle 5, 3\rangle$, $\langle 5, 1\rangle$.

Число этих размещений можно найти по формуле

$$A_3^2 = \frac{3!}{(3-2)!} = \frac{1 \cdot 2 \cdot 3}{1!} = 6.$$

б) *Перестановки.* Пусть у нас есть n-элементное множество A, будем строить из этого множества размещения в виде кортежей длины n. Эти размещения будут отличаться друг от друга только порядком, поскольку в каждом из них встречаются по одному разу все элементы множества A. Такие размещения называются перестановками и обозначаются P_n (буква P от английского слова *permutation* — перестановка). Поскольку $P_n = A_n^n$, то число перестановок вычисляется по формуле $P_n = n!$.

Выясним, например, сколько трехзначных чисел можно составить из цифр 1, 2, 3, если каждая входит в число только один раз.

Составим такие числа: $\langle 1,2,3\rangle$, $\langle 1,3,2\rangle$, $\langle 2,1,3\rangle$, $\langle 2,3,1\rangle$, $\langle 3,1,2\rangle$, $\langle 3,2,1\rangle$ — то есть шесть чисел. С помощью введенной формулы можно сразу определить число перестановок, не выписывая их:

$$P_3 = 3! = 1\cdot 2\cdot 3 = 6.$$

в) *Сочетания.* Из n-элементного множества A будем строить упорядоченные множества длины k ($1 \leqslant k \leqslant n$), не учитывая порядок элементов, т.е. размещения с одними и теми же элементами, расположенными в разном порядке, будем считать равными.

Такие размещения называются сочетаниями и обозначаются C_n^k (буква C от английского слова *combination* — комбинация).

Число сочетаний из n элементов по k меньше числа размещений из n элементов по k в P_k раз, т. е. $C_n^k = \dfrac{A_n^k}{P_k}$.

Используя это утверждение, выведем формулу для вычисления числа сочетаний:

$$C_n^k = \frac{A_n^k}{P_k} = \frac{n!}{(n-k)!\cdot k!} = \frac{n(n-1)\ldots(n-k+1)}{k!}.$$

Из этой формулы непосредственно вытекает, что $C_0^0 = C_n^0 = C_n^n = 1$; $C_n^1 = n$; $C_n^k = C_n^{n-k}$.

Выясним, какие парные сочетания можно составить из цифр 1, 3, 5 и сколько их. Выпишем эти сочетания:

$$\langle 1,3\rangle, \quad \langle 1,5\rangle, \quad \langle 3,5\rangle,$$

т. е. $C_3^2 = \dfrac{3!}{1!\cdot 2!} = 3$ или $C_3^2 = C_3^{3-2} = C_3^1 = 3$.

Непосредственной проверкой легко доказать следующие тождества:

а) $C_n^k \cdot C_k^r = C_n^r \cdot C_{n-r}^{k-r}$, где $0 \leqslant r \leqslant k \leqslant n$.

Действительно:

$$C_n^k \cdot C_k^r = \frac{n!\cdot k!}{(n-k)!\cdot k!\cdot (k-r)!\cdot r!} = \frac{n!}{(n-r)!\cdot r!}\cdot\frac{(n-r)!}{(n-k)!\cdot(k-r)!} = C_n^r\cdot C_{n-r}^{k-r}.$$

б) $C_{n-1}^k + C_{n-1}^{k-1} = C_n^k$:

$$C_{n-1}^k + C_{n-1}^{k-1} = \frac{(n-1)!}{(n-k-1)!k!} + \frac{(n-1)!}{(n-k)!(k-1)!} =$$

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

3. Комбинации элементов с повторениями

Все приведенные формулы справедливы в том случае, когда n элементов множества A различны. Если же некоторые элементы повторяются, то в этом случае рассматриваются комбинации **с повторениями**, число которых вычисляется по другим формулам.

Размещениями с повторениями из n элементов по k называются кортежи длины k, составленные из n — элементного множества A. Число этих кортежей обозначают $\overline{A_n^k}$. Черта указывает на возможность повторения элементов $\overline{A_n^k} = n^k$. Например, сколько пятизначных номеров можно составить из элементов множества $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$? Такими номерами являются кортежи длины 5, составленные из девятиэлементного множества, где схема выбора состоит в выборе 5 элементов из девятиэлементного множества с возвращением, т. е. для каждого из пяти элементов есть девять способов выбора, т. е. $\overline{A_9^5} = 9^5 = 59\,049$.

Перестановкой с повторениями состава $(n_1, ..., n_k)$ из букв $(a_1, \ldots, a_k)$ называют любой кортеж длины $n = n_1 + n_2 + \ldots + n_k$, в который a_1 входит n_1 раз, a_2 входит n_2 раз, ..., a_k — n_k раз. Число таких перестановок обозначают

$$P(n_1, n_2, ..., n_k) = \frac{n!}{n_1! n_2! ... n_k!} = \frac{(n_1 + n_2 + \ldots + n_k)!}{n_1! n_2! ... n_k!}.$$

Еще один пример. Сколько слов можно получить, переставляя буквы в слове «математика»? Слово «математика» является кортежем длины 10, имеющем состав $(2, 3, 2, 1, 1, 1)$ (буква «м» входит два раза, буква «а» входит 3 раза, буква «т» входит два раза, буквы «е», «и», «к» входят по одному разу). Значит, при перестановках букв получится $P(2, 3, 2, 1, 1, 1) = \dfrac{10!}{2! \cdot 3! \cdot 2! \cdot 1! \cdot 1! \cdot 1!} = 151\,200$ слов.

Сочетания с повторениями. Пусть имеются предметы n видов и из них составляется набор, содержащий k элементов, т. е. различными исходами будут всевозможные наборы длины k, отличающиеся составом, и при этом отдельные наборы могут содержать повторяющиеся элементы. Такие наборы называются сочетаниями с повторениями, а их общее число определяется формулой : $\overline{C_n^k} = C_{n+k-1}^k$.

Например, нужно выяснить, сколько наборов из 7 пирожных можно составить, если в продаже имеются 4 сорта?

Искомое число равно $C_{4+7-1}^7 = C_{10}^7 = C_{10}^3 = \dfrac{10 \cdot 9 \cdot 8}{1 \cdot 2 \cdot 3} = 120.$

4. Бином Ньютона

С числами C_n^k связано функциональное тождество, называемое формулой бинома Ньютона. Из элементарной математики хорошо известны формулы сокращенного умножения:

$$(a+b)^2 = a^2 + 2ab + b^2; \quad (a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3.$$

Эти формулы можно записать так:

$$(a+b)^2 = C_2^0 a^2 b^0 + C_2^1 ab + C_2^2 a^0 b^2;$$
$$(a+b)^3 = C_3^0 a^3 b^0 + C_3^1 a^2 b^1 + C_3^2 ab^2 + C_3^3 a^0 b^3.$$

Имеет место и общая закономерность: справедливо равенство:

$$(a+b)^n = C_n^0 a^n b^0 + C_n^1 a^{n-1} b + C_n^2 a^{n-2} b^2 + \ldots + C_n^n a^0 b^n.$$

Это равенство и называется биномом Ньютона, а коэффициенты $C_n^0, C_n^1, C_n^2, \ldots, C_n^n$ называются биномиальными коэффициентами.

Если положить $a = b = 1$, то из формулы бинома Ньютона вытекает следующее важное соотношение: $(1+1)^n = C_n^0 + C_n^1 + C_n^2 + \ldots + C_n^n = 2^n$ — формула суммы биномиальных коэффициентов.

Если положить в биноме Ньютона $a = 1$, $b = -1$, то

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

Поскольку $C_n^k = C_n^{n-k}$, то биномиальные коэффициенты, равноотстоящие от концов в формуле бинома Ньютона, равны.

Пятое практическое занятие по теме «Комбинаторные формулы. Бином Ньютона»

Задача 1. Составьте все перестановки:

1) из трех букв: a, b, c;

2) из четырех цифр: 5, 4, 3, 2.

Решение.

1) abc, acb, bac, bca, cab, cba;

2) 5432, 5423, 5342, 5324, 5243, 5234, 4532, 4523, 4325, 4352, 4235, 4253, 3542, 3524, 3452, 3425, 3245, 3254, 2345, 2354, 2435, 2453, 2534, 2543.

Задача 2. Составьте все размещения:

1) из четырех букв a, b, c, d по 3 буквы в каждом (без повторений);

2) из четырех цифр: 1, 3, 5, 7 по 2 цифры в каждом.

Решение.

1) *abc abd bcd acd acb adb bdc adc bac bad cbd cad bca bda cdb cda cab dab dbc dac cba dba dcb dca*;

2) 1 3; 1 5; 1 7; 3 1; 3 5; 3 7; 5 1; 5 3; 5 7; 7 1; 7 3; 7 5.

Задача 3. Вычислите:

1) A_6^3; 2) A_7^4; 3) A_8^5; 4) $\dfrac{A_6^3}{A_5^2}$; 5) $\dfrac{A_8^3 + A_7^4}{A_6^3}$; 6) $\dfrac{A_{10}^6 - A_{10}^5}{A_9^5 - A_9^4}$.

Задача 4. Вычислите:

1) P_4; 2) P_6; 3) P_9; 4) $\dfrac{P_8}{P_6}$; 5) $\dfrac{P_5 + P_4}{P_3}$; 6) $\dfrac{P_6 - P_4}{P_3}$.

Задача 5. Вычислите:

1) $\dfrac{P_8}{A_8^7}$; 2) $\dfrac{A_7^4 - P_5}{A_5^2}$; 3) $\dfrac{2P_3 + 3A_4^2}{5P_5 - P_3}$; 4) $\dfrac{P_8 P_7}{7P_7}$.

Задача 6. Составьте все сочетания (без повторений) из пяти букв: a, b, c, d, e по 3 буквы в каждом.

Решение. abc, abd, abe, acd, ace, ade, bcd, bce, bde, cde.

Задача 7. Вычислите:

1) C_6^2; 2) C_8^3; 3) C_{11}^4; 4) C_{12}^7; 5) C_{100}^{98}; 6) C_{20}^{17}; 7) C_{40}^{38}; 8) C_{54}^{52}.

Задача 8. Проверьте равенства:

1) $C_{10}^{15} = \dfrac{A_{15}^5}{P_5}$; 2) $C_6^2 = \dfrac{A_m^{m-8}}{P_{m-8}}$.

Задача 9. Решите уравнения:

1) $A_{x+1}^2 = 30$; 2) $5C_x^3 = C_{x+2}^4$; 3) $C_x^3 = \dfrac{5x(x-3)}{4}$; 4) $C_x^3 + C_x^2 = 15(x-1)$;

5) $\dfrac{A_x^5 + A_x^3}{A_x^3} = 43$; 6) $12C_x^1 + C_{x+4}^2 = 162$; 7) $C_{x+1}^5 = \dfrac{3A_x^3}{8}$; 8) $\dfrac{A_x^4 \cdot P_{x-4}}{P_{x-2}} = 42$.

Ответ: 1) 5; 2) 3; 14; 3) 7; 4) 9; 5) 10; 6) 8; 7) 8; 8) 7.

Задача 10. Сколько номеров, состоящих из трех букв, за которыми идут две цифры, можно составить, используя 32 буквы и 10 цифр?

Решение. Обозначим множество из 32 букв через A, а множество из 10 цифр через B. Тогда $|A| = 32$; $|B| = 10$. Каждый номер требуемого вида является кортежем длины 5 из декартова произведения $A \times A \times A \times B \times B$. Тогда $|A \times A \times A \times B \times B| = 32 \cdot 32 \cdot 32 \cdot 10 \cdot 10 = 3\,276\,800$.

Задача 11. Сколько существует пятизначных номеров:

1) не содержащих цифру 8;

2) не содержащих цифры 0 и 8;

3) составленных из цифр 2, 3, 5, 7?

Ответ: **1)** 59 049, **2)** 32 768, **3)** 1024. Указание: здесь $\overline{A_n^k} = n^k$.

Задача 12. Сколькими способами можно разложить 28 различных предметов по четырем ящикам, так, чтобы в каждом ящике оказалось по 7 предметов?

Решение. Число способов равно: $P(7,7,7,7) = \dfrac{28!}{(7!)^4}$.

Задача 13. В технической библиотеке имеются книги по математике, физике, химии и т. д., всего по 16 разделам науки. Поступили очередные четыре заказа на литературу. Сколько наборов из четырех заказов можно составить по 16 разделам науки?

Решение. Искомое число равно числу сочетаний с повторениями из 16 элементов по 4, т. е. $\overline{C_{16}^4} = C_{16+4-1}^4 = C_{19}^4 = 3876$.

Задача 14. Сколькими способами можно составить дозор из трех солдат и одного офицера. Если имеется 80 солдат и 3 офицера?

Ответ: 246 480.

Задача 15. Найдите:

1) четвертый член разложения $(a+3)^7$;

2) четвертый член разложения $(a+\sqrt{b})^{12}$;

3) восьмой член разложения $(a^2+b^3)^{13}$;

4) средний член разложения $(\sqrt{a}-\sqrt{b})^8$;

5) средний член разложения $(x\sqrt{x}-1)^{14}$;

6) два средних члена разложения $(\sqrt{a}-\sqrt[3]{b})^{13}$.

Ответ: **1)** $945a^4$; **2)** $495a^4b^4$; **3)** $1287(a^{16}b^{15})$; **4)** $70a^2b^2$; **5)** $-3432x^{10}\sqrt{x}$; **6)** $-1716a^3 \cdot b^2\sqrt[3]{b}$; $1716a^3\sqrt{a}b^2$.

Задача 16. Определите x из условия, что третий член разложения бинома $(x+x^{\lg x})^5$ равен 1 000 000.

Ответ: 10; $\dfrac{1}{\sqrt{100\,000}}$.

Задача 17. Найдите тот член разложения бинома

$$\left(z\sqrt{z}+\frac{1}{\sqrt[3]{z}}\right)^m,$$

который после упрощения содержит z^5, если сумма биномиальных коэффициентов этого разложения равна 128.

Ответ: $35z^5$.

Контрольные вопросы

1. Что такое комбинаторика и для чего она нужна?

2. Что называется:

— перестановкой n-элементного множества;

— размещением из n элементов по m элементов;

— сочетанием из n элементов по m элементов?

3. В чем отличие размещений от перестановок?

4. В чем отличие сочетаний от размещений?

5. Сколькими способами можно разместить три книги на книжной полке?

6. Запишите формулу для вычисления числа сочетаний элементов, используемую в формуле бинома Ньютона.

7. Как найти число перестановок с повторениями?

8. Сколько существует пятизначных чисел, у которых каждая следующая цифра:

— меньше предыдущей,

— больше предыдущей.

9. Сколько прямых можно провести через n точек, если никакие три из них не лежат на одной прямой?

10. Сколько разных слов можно составить перестановкой букв в слове «чачача»?

11. Вычислите: $(a + b + c)^2$; $(a + b + c)^3$.

12. Покажите, что сумма $C_p^1 + C_p^2 + \ldots + C_p^{p-1}$ делится на p, где p — простое число.

13. Докажите свойства биномиальных коэффициентов.

Часть 4

Логика предикатов или логика первого порядка

Лекции 9–10

Логика предикатов — новая логическая система, представляющая развитие логики высказываний. Исторически понятие о предикатах явилось следствием логического анализа высказываний, т. е. выяснения их логической структуры. Выяснения того, какой логикой может быть выражен смысл этих высказываний.

1. Предикаты

Рассмотрим пример: «x простое число». Это выражение не является высказыванием, но если в нем переменную x заменить на определенное число, то получим высказывание. Причем при замене x на число 3 получим истинное высказывание, тогда как при замене x на 8 получим ложное высказывание.

Таким образом, выражение: «x простое число» можно рассматривать как функцию $P(x)$, зависящую от переменной x. Область определения $P(x)$ — множество чисел, а область значения — высказывание.

Определение. Предикат — функция, значениями которой являются высказывания о n объектах, представляющих значения аргументов.

Чтобы задать n-местный предикат $P(x_1, x_2, \ldots, x_n)$, следует указать множества $X_1, X_2, \ldots, X_n$ — области изменения переменных $x_1, x_2, \ldots, x_n$, причем чаще всего рассматривается случай, когда $X_1 = X_2 = \ldots = X_n$.

С теоретико-множественной точки зрения предикат определяется заданием подмножества M в декартовом произведении $X_1 \times X_2 \times \ldots \times X_n$.

Переменные $x_1, x_2, \ldots, x_n$ называются *предметными переменными*. Элементы множеств $X_1, X_2, \ldots, X_n$ называются *предметами*. Множество

M — множество кортежей длины n $\langle x_1, x_2, \ldots, x_n \rangle$ называется *полем предиката* $P(x_1, x_2, \ldots, x_n)$.

Будем обозначать предметные переменные малыми буквами конца латинского алфавита (иногда будем снабжать эти буквы индексами) $x, y, z, \ldots, x_1, x_2, \ldots, x_n$.

Предметы из множеств $X_1, X_2, \ldots, X_n$ — малыми буквами начала латинского алфавита $a, b, c, \ldots, a_1, a_2, a_3 \ldots$

Предикаты — большими буквами латинского алфавита с приписанными предметными переменными или без них $A(x, x)$, B, $F(x, y)$, $P(x_1, \ldots, x_n)$.

Число переменных будем указывать как верхний индекс у предиката: $P^k(x_1, x_2, \ldots, x_k)$ — k местный предикат, $Q^2(x, y)$ — двуместный предикат, $P(x)$ — одноместный предикат.

Итак, k-местный предикат — $P^k(x_1, x_2, \ldots, x_k)$ есть функция, предметные переменные которой принимают значения из некоторого множества M_k, а сама она принимает только два значения: истина (1) или ложь (0), т. е.

$$P^k(x_1, x_2, \ldots, x_k) : \; M_k \to \{1, 0\}.$$

Например, если X — множество действительных чисел, то $x^2 > 1$ — одноместный предикат.

Если X, Y — множества действительных чисел, то $xy = 5$ — двуместный предикат.

Предикат называется *разрешимым*, если существуют такие кортежи, компоненты которых обращают предикат в истинное высказывание.

Если предикат при подстановке любых конкретных элементов из соответствующих множеств обращается в истинное высказывание, он называется *тождественно истинным*.

Если предикат при подстановке любых конкретных элементов из соответствующих множеств обращается в ложное высказывание, он называется *тождественно ложным*.

К предикатам, определенным на одном и том же множестве, можно применять операции алгебры высказываний: конъюнкцию, дизъюнкцию, импликацию, эквивалентность, отрицание и получать новые предикаты.

Например, если к предикатам «$x = y$» и «$x < y$» — обозначим их соответственно $P(x, y)$ и $Q(x, y)$ — применить операцию конъюнкции, то получим новый предикат $P(x, y) \wedge Q(x, y)$.

2. Применение предикатов в алгебре

Рассмотрим предикаты, в которых свободной является лишь одна переменная, которую обозначим через x, и обсудим применение предикатов в алгебре.

Типичным примером является уравнение, например, $x^2 - 3x + 2 = 0$. Свободная переменная может принимать здесь любое числовое значение. Для некоторых чисел x (а именно $x = 1$, $x = 2$) утверждение, содержащееся в этом уравнении, истинно, в остальных оно ложно. В подобных случаях, когда истинность или ложность предиката зависит только от значения, принимаемого свободной переменной x, множество допустимых значений x можно рассматривать как множество логических возможностей U, а множество всех значений этой переменной, при которых высказывание истинно — как его множество истинности.

В приведенном выше примере множество U состоит из всех действительных чисел, а множеством истинности является множество $\{1, 2\}$.

В результате введения понятия множества истинности для предикатов мы сможем сказать, что решить уравнение — значит найти один элемент или все элементы его множества истинности. При решении системы двух уравнений у нас имеется предикат, представляющий конъюнкцию двух уравнений. Поэтому мы ищем пересечение двух множеств истинности. Если это пересечение пусто, то система уравнений не имеет решений. Такие уравнения называются *несовместными*, поскольку их множества истинности не имеют общих элементов x.

Понятие множества истинности удобно не только в вопросах, связанных с решением уравнений, но и при рассмотрении неравенств.

Если U — множество действительных чисел, то множество истинности неравенства $x < 0$ состоит из всех отрицательных действительных чисел. Множество же истинности неравенства $x > -3$ состоит из всех действительных чисел, больших, чем -3. Если мы потребуем, чтобы эти неравенства выполнялись одновременно, то множеством истинности будет множество, являющееся пересечением двух исходных множеств, т. е. все действительные числа между -3 и 0.

Понятие множества истинности предиката позволяет выяснить, чем разнятся между собой уравнения и тождества. Когда мы решаем уравнение, мы тем самым ищем один из элементов множества истинности этого уравнения или все его элементы. Если же мы доказываем тождество, то тем самым утверждаем, что оно справедливо для всех x. Таким образом, тождество представляет собой уравнение, множеством истинности которого является универсальное множество U, т. е. является логически истинным или тождественно истинным.

3. Булева алгебра предикатов

Так как к предикатам можно применять логические операции, то для них справедливы основные законы булевой алгебры.

1. $\overline{\overline{P}} = P$
2. $P \vee Q = Q \vee P$
3. $P \wedge Q = Q \wedge P$
4. $P \vee (Q \vee R) = (P \vee Q) \vee R$
5. $P \wedge (Q \wedge R) = (P \wedge Q) \wedge R$
6. $P \vee (Q \wedge R) = (P \vee Q) \wedge (P \vee R)$
7. $P \wedge (Q \vee R) = (P \wedge Q) \vee (R \wedge R)$
8. $\overline{P \vee Q} = \overline{P} \wedge \overline{Q}$; $\overline{P \wedge Q} = \overline{P} \vee \overline{Q}$
9. $P \vee P = P$; $P \wedge P = P$
10. $P \vee 1 = 1$; $P \wedge 0 = 0$; $P \vee 0 = P$; $P \wedge 1 = P$; $P \vee \overline{P} = 1$; $P \wedge \overline{P} = 0$
11. $P \vee (P \wedge Q) = P$
12. $P \wedge (P \vee Q) = P$

4. Кванторы

Помимо операций алгебры высказываний, в логике предикатов есть две операции, которые связаны с природой предикатов. Пусть дан предикат $P(x)$, зависящий от одной переменной и определенный на поле M.

а) Выражение $(\forall x)P(x)$ означает высказывание, истинное только в том случае, когда предикат $P(x)$ истинен для всех предметов из поля M. Выражение $(\forall x)P(x)$ читается «для всякого x, $P(x)$», здесь символ $\forall$ — квантор общности.

б) Выражение $(\exists x)P(x)$ означает высказывание, истинное только в том случае, когда предикат $P(x)$ истинен хотя бы для одного предмета из поля M. Выражение $(\exists x)P(x)$ читается «существует x, что $P(x)$»; символ $\exists$ — квантор существования.

Рассмотрим примеры применения операций квантирования к предикатам. Пусть даны предикаты над полем натуральных чисел:

1) $x^2 = x \cdot x$, тогда $(\forall x)(x^2 = x \cdot x)$ — истинное высказывание;

2) $x + 2 = 7$, тогда $(\forall x)(x + 2 = 7)$ — ложное высказывание; а $(\exists x)(x + 2 = 7)$ — истинное высказывание;

3) $x + 2 = x$, тогда $(\forall x)(x + 2 = x)$ — ложное высказывание.

Название	Прочтение	Обозначение
Квантор общности	«все», «всякий», «каждый», «любой»	$\forall$
Квантор существования	«Хотя бы один», «найдется», «существует»	$\exists$

Квантор общности — это оператор, приводящий в соответствии любому заданному предикату $y = P(x)$ такую двузначную логическую переменную z, которая принимает значение 1 тогда и только тогда, когда $y = 1$ при всех значениях x.

Квантор существования — это оператор, приводящий в соответствии любому одноместному предикату $y = P(x)$ такую двузначную логическую переменную z, которая принимает значение 0 тогда и только тогда, когда $y = 0$ при всех значениях x.

Рассмотрим некоторые общие свойства введенных операторов.

В соответствии с определениями кванторов логическая переменная z в выражениях

$$\left.\begin{aligned} z &= (\forall x)P(x) \\ z &= (\exists x)P(x) \end{aligned}\right\}$$

уже не является функцией предметной переменной x.

Для того чтобы отметить отсутствие функциональной зависимости z от x, предметную переменную x в таких случаях называют *связанной*. Несвязанные переменные называют *свободными*.

Например, в предикате

$$(\forall x)A(x, y) \vee (\exists z)B(z, v)$$

переменные x и z — связанные, а y и v — свободные.

Если квантор общности или квантор существования применяется не к одноместному предикату, а к какому-нибудь k-местному предикату, то в результате этого получается снова предикат, но за счет связывания одной предметной переменной получаемый предикат будет $(k - 1)$-местным.

5. Формулы логики предикатов

Наряду с *определенными предикатами* — для которых истинность или ложность известны для каждого набора значений свободных предметных переменных, будем рассматривать переменные предикаты, для которых не определены значения. Будем обозначать переменные предикаты большими буквами из конца латинского алфавита с приписанными предметными переменными или без них:

$$W(x_1, \ldots, x_n); \quad U(x, y), \ldots .$$

Применяя к переменным предикатам операции $\wedge$; $\vee$; $\to$; $\leftrightarrow$; $\overline{}$; $\forall$; $\exists$, получим формулы логики предикатов, т. е. *формулой логики предикатов* называется выражение, составленное из переменных предикатов с помощью логических операций и кванторов и обращающееся в конкретный предикат при подстановке вместо переменных конкретных предикатов.

Например, $((\forall x)W(x, y) \vee B) \to U(z)$ — формула логики предикатов.

Формула логики предикатов называется *тавтологией*, если при подстановке любых конкретных предикатов она всегда обращается в тождественно истинный предикат.

Сформулируем следующие правила.

(1) Формула логики предикатов называется *атомарной*, т. е. *элементарной*, если в ней нет связанных переменных.

(2) Пусть F — формула, тогда $\overline{F}$ — тоже формула. Свободные и связанные переменные формулы $\overline{F}$ — это соответственно свободные и связанные переменные формулы F.

(3) Пусть F и G — формулы, причем в них нет предметных переменных, которые были бы связаны в одной формуле и свободны в другой.

Тогда $F \wedge G$, $F \vee G$, $F \to G$, $F \leftrightarrow G$ — формулы, в которых свободные переменные формул F и G остаются свободными, а связанные — связанными.

(4) Пусть F — формула, содержащая свободную переменную x. Тогда $(\forall x)F$, $(\exists x)F$ — тоже формулы, в которых переменная x связана, а остальные свободные переменные, входящие в F, остаются свободными.

Заметим, что по определению формулы никакая переменная не может быть одновременно свободной и связанной.

Значение формулы определено лишь тогда, когда задана какая-то *интерпретация* входящих в нее символов. Под интерпретацией понимают систему $M = \langle M, f \rangle$, состоящую из непустого множества M и соответствия f, которое сопоставляет каждой формуле определенный предикат. При заданной интерпретации предметные переменные пробегают множество M, а логические символы и символы кванторов имеют свой обычный смысл.

6. Равносильные формулы логики предикатов

Пусть формулы F и G имеют одно и то же множество свободных переменных (в частности, пустое). Формулы F и G *равносильны в данной интерпретации*, если они принимают одинаковые значения на любом наборе свободных переменных, т. е. выражают в данной интерпретации один и тот же предикат.

Формулы F и G равносильны *на множестве* M, если они принимают одинаковые значения во всех интерпретациях заданных на множестве M.

Формулы F и G равносильны в логике предикатов, если они равносильны на всех множествах ($F \equiv G$).

Рассмотрим правила перехода от одних формул к другим, им равносильным.

(1) *Перенос квантора через отрицание.* Пусть $W(X)$ — формула, содержащая свободную переменную x. Тогда справедливы равносильности:

$$\overline{(\forall x)W(x)} \equiv (\exists x)\overline{W(x)}, \quad \overline{(\forall x)\overline{W(x)}} \equiv (\exists x)W(x),$$

$$\overline{(\exists x)W(x)} \equiv (\forall x)\overline{W(x)}, \quad \overline{(\exists x)\overline{W(x)}} \equiv (\forall x)W(x).$$

(2) *Вынос квантора за скобки.* Пусть формула $W(x)$ содержит свободную переменную x, а формула B не содержит переменной x. Формулы $W(x)$ и B удовлетворяют третьему правилу создания формул. Тогда справедливы равносильности:

$$(\exists x)(W(x) \wedge B) \equiv (\exists x)W(x) \wedge B, \quad (\forall x)(W(x) \wedge B) \equiv (\forall x)W(x) \wedge B,$$

$$(\exists x)(W(x) \vee B) \equiv (\exists x)W(x) \vee B, \quad (\forall x)(W(x) \vee B) \equiv (\forall x)W(x) \vee B.$$

(3) *Перестановка одноименных кванторов.* Имеем

$$(\exists x)(\exists y)W(x,y) \equiv (\exists y)(\exists x)W(x,y), \quad (\forall x)(\forall y)W(x,y) \equiv (\forall y)(\forall x)W(x,y).$$

(4) *Переименование связанных переменных.* Заменяя связанную переменную формулы W другой переменной, не входящей в эту формулу, в кванторе и всюду в области действия квантора, получим формулу, равносильную W.

7. Приведенные и нормальные формы в логике предикатов

Рассмотрим способ упрощения формул, опирающийся на приведенные равносильности.

Формулы, в которых из логических символов имеются только символы конъюнкция, дизъюнкция и отрицание, причем символ отрицания встречается над символами предикатов, будем называть *приведенными*.

Например, формула $(\forall x_1)A_1^{(1)}(x_1) \vee (\exists x_2)\overline{A_2^{(2)}(x_2,x_3)}$ — приведенная; формула $\overline{(\forall x_2)}A_1^{(1)}(x_2) \to A_2^{(1)}(x_1)$ — неприведенная.

Для любой формулы существует равносильная ей приведенная формула, причем множества свободных и связанных переменных этих формул совпадают.

Такая приведенная формула называется *приведенной формой* данной формулы.

В логике высказываний мы ввели две нормальные формы — дизъюнктивную нормальную форму и конъюнктивную нормальную форму.

В логике предикатов также имеется нормальная форма, цель которой — упрощение процедуры доказательств.

Приведенная формула называется *нормальной*, если она не содержит символов кванторов или все символы кванторов стоят впереди (т. е. логические символы и символы предикатов стоят в области действия каждого квантора).

Для любой приведенной формулы существует равносильная ей нормальная формула той же длины (под длиной формулы будем понимать общее число входящих в нее символов предикатов, логических символов и символов кванторов).

Нормальная формула называется *нормальной формой* данной формулы.

Приведем несколько формул, находящихся в нормальной форме:

$$(\forall x)(\forall y)(P(x,y) \wedge Q(y)), \quad (\forall x)(\forall y)\left(\overline{P(x,y)} \to Q(y)\right),$$
$$(\forall x)(\forall y)(\exists z)(Q(x,y) \to P(z)).$$

Алгоритм преобразования формул в нормальную форму:

1. Исключить логические связки $\leftrightarrow$ и $\to$ с помощью формул

$$F \leftrightarrow G = (F \to G) \wedge (G \to F), \quad F \to G = \overline{F} \vee G.$$

2. Использовать закон $\overline{\overline{F}} = F$, законы де Моргана:

$$\overline{(F \vee G)} = \overline{F} \wedge \overline{G}, \quad \overline{(F \wedge G)} = \overline{F} \vee \overline{G},$$

законы

$$\overline{((\forall x)F(x))} = (\exists x)(\overline{F(x)}), \quad \overline{((\exists x)F(x))} = (\forall x)(\overline{F(x)}),$$

чтобы пронести знак отрицания внутрь формулы.

3. Переименовать связанные переменные, если это необходимо.

4. Использовать равносильные формулы логики предикатов, чтобы вынести кванторы в самое начало формулы для приведения ее к нормальной форме. Например, приведем формулу $(\forall x)P(x) \to (\exists x)Q(x)$ к нормальной форме:

$$(\forall x)P(x) \to (\exists x)Q(x) = \overline{((\forall x)P(x))} \vee (\exists x)Q(x) = (\exists x)(\overline{P(x)}) \vee (\exists x)Q(x) =$$
$$= (\exists x)\left(\overline{P(x)} \vee Q(x)\right).$$

Следовательно, нормальная форма формулы $(\forall x)P(x) \to (\exists x)Q(x)$ — это $(\exists x)(\overline{P(x)} \vee Q(x))$.

8. Исчисление предикатов

Исчисление предикатов называют еще теорией первого порядка.

В исчислении предикатов, так же как и в исчислении высказываний, на первом по важности месте стоит проблема *разрешимости*.

Но в исчислении высказываний проблема разрешимости состояла в решении вопроса является ли данная сложная функция тождественно истинной, выполнимой или тождественно ложной?

Теперь же вопрос следует поставить иначе. Принимает ли данная функция значение 1 при:

а) любых предметных переменных и любых предикатах,

б) на некотором множестве предметных переменных и любых предикатах,

в) при некоторых значениях предметных переменных и при некоторых предикатах,

г) является ли она тождественно ложной, т. е. невыполнимой?

Таким образом, в логике предикатов, в отличие от логики высказываний, нет эффективного способа для распознавания общезначимости функций.

Поэтому в исчислении предикатов указывается некоторая совокупность формул, которые называются аксиомами и составляют *аксиоматическую теорию*, и указывается конечное множество отношений между формулами, составляющее *правила вывода*.

Аксиоматическая теория и правила вывода и составляют исчисления предикатов.

Символами исчисления предикатов или алфавитом исчисления предикатов являются символы предметных переменных, символы предикатов, логические символы (отрицание и импликация), символы кванторов, а также скобки и запятая.

Сформулируем аксиомы исчисления предикатов и правила вывода исчисления предикатов.

Аксиомы исчисления предиката.

Пусть A, B и C — любые формулы.

Аксиома 1. $A \to (B \to A)$.

Аксиома 2. $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$.

Аксиома 3. $(\overline{B} \to \overline{A}) \to \big((\overline{B} \to A) \to B\big)$.

Аксиома 4. $(\forall x_i)A(x_i) \to A(x_j)$, где формула $A(x_i)$ не содержит переменной x_j.

Аксиома 5. $A(x_i) \to (\exists x_j)A(x_j)$, где формула $A(x_i)$ не содержит переменной x_j.

Правила вывода исчисления предикатов.

(1) Пусть $(A(x) \to B)$ и B не содержит переменной x, тогда

$$\Big(((\exists x)A(x)) \to B\Big).$$

Это правило связывания квантором существования.

(2) Пусть $B \to A(x)$ и B не содержит переменной x, тогда

$$\Big(B \to ((\forall x)A(x))\Big).$$

Это правило связывания квантором общности.

(3) Связанную переменную формулы B можно заменить другой переменной, не являющейся свободной в B. Это правило переименования связанной переменной.

Шестое практическое занятие по теме «Предикаты»

Задача 1. Пусть U — множество всех действительных чисел. Постройте множество истинности для каждого из следующих предикатов:

а) $x^2 - 4 = 0$, б) $x^2 + 4 = 0$, в) $x^2 - 4x + 3 = 0$, г) $x^2 - 4x + 4 = 0$, д) $x^2 - 4x + 5 = 0$.

Ответ: а) $\{2, -2\}$, б) $\varnothing$, в) $\{3, 1\}$, г) $\{2\}$, д) $\varnothing$.

Задача 2. Пусть U — множество всех действительных чисел. Найдите множество истинности конъюнкций следующих предикатов:

а) $x^2 + x - 2 = 0$; $x^2 = 4$, б) $x^2 - 4 = 0$; $x^2 - 4x + 4 = 0$,

в) $x^3 - 6x^2 + 11x - 6 = 0$; $x^2 - 4x + 3 = 0$, г) $x^3 = 1$; $x^2 - 4x + 4 = 0$.

Найдите множества истинности конъюнкций следующих пар неравенств:

д) $x \geqslant 3$; $x \leqslant 10$, е) $x^2 \leqslant 4$; $x - 1 \geqslant 1$, ж) $x \leqslant 0$; $x^2 - 2x \leqslant 0$.

Задача 3. На множестве однозначных натуральных чисел даны два предиката: предикат $P(x)$: «число 3 делитель x»; предикат $Q(x)$: «$x \leqslant 6$». Найдите множества истинности предикатов:

(1) $P(x) \vee Q(x)$, (2) $P(x) \wedge \overline{Q(x)}$, (3) $\overline{P(x)} \to Q(x)$, (4) $\overline{P(x) \to Q(x)}$.

Решение. (1) $\{1, 2, 3, 4, 5, 6, 9\}$, (2) $\{9\}$, (3) $\{1, 2, 3, 4, 5, 6, 9\}$, (4) $\{9\}$.

Задача 4. Предикат $P(x)$: «x есть простое число»; предикат $Q(x)$: «x есть действительное число»; предикат $T(x)$: «x меньше y». Запишите следующие утверждения, используя кванторы:

а) каждое рациональное число есть действительное число;

б) существует число, которое является простым;

в) для каждого числа x существует такое число y, что $x < y$.

Решение. а) $(\forall x)\,(Q(x) \leftrightarrow R(x))$; б) $(\exists x)P(x)$; в) $((\forall x)(\exists x)T(x))$.

Задача 5. Основные аксиомы натуральных чисел таковы:

A_1: Для каждого числа существует одно и только одно число, непосредственно следующее за ним.

A_2: Нет числа, за которым непосредственно следует 0.

A_3: Для каждого числа, отличного от нуля, существует одно и только одно непосредственно предшествующее ему число.

Пусть $f(x)$ и $y(x)$ представляют соответственно число, непосредственно следующее за x и непосредственно предшествующее x.

Пусть «x равно y» — предикат, обозначенный через $E(x, y)$. Записать аксиомы с помощью формул логики предикатов.

Решение. A_1: $(\forall x)(\exists y)E(y, f(x)) \wedge (\forall z)(E(z, f(x)) \to E(y, z))$,

A_2: $\overline{((\exists x)E(0, f(x)))}$,

A_3: $(\forall x)\overline{(E(x, 0))} \to ((\exists y)(E(y, g(x))) \wedge (\forall z)(E(z, g(x)) \to E(y, z)))$.

Задача 6. Определите значение формул:

а) $(\forall x)P(x)$; б) $(\exists x)\overline{P(x)}$

в интерпретации $M = \langle M, f\rangle$ где $M = \{1, 2\}$; f: $P(1)$ — истина; $P(2)$ — ложь.

Решение.

а) $(\forall x)P(x)$ — ложь, так как $P(x))$ — не истина как для $x = 1$, так и для $x = 2$;

б) $(\exists x)\overline{P(x)}$ — истина, так как $\overline{P(2)}$ в этой интерпретации истина.

Задача 7. Оцените формулу $(\forall x)(\exists y)P(x, y)$ в интерпретации $M\{1, 2\}$

$P(1,1)$	$P(1,2)$	$P(2,1)$	$P(2,2)$
И	Л	Л	И

Решение. Если $x = 1$, то существует $y = 1$ такой, что $P(1, y)$ есть истина.

Если $x = 2$, также существует такой y, что $P(2, y)$ — истина.

Следовательно, в указанной интерпретации для каждого x из M существует такое значение y, что $P(x, y)$ есть истина, т. е. $(\forall x)(\exists y)P(x, y)$ есть истина в этой интерпретации.

Задача 8. Рассмотрим формулу G: $(\forall x)P(x) \to Q(f(x), a)$, здесь a — константа, f — одноместная функция, P — одноместный предикат, Q — двуместный предикат. Интерпретация формулы G задана следующим образом: $M\{1, 2\}$; $a : \frac{a}{1}$; $f(1) = 2$; $f(2) = 1$; $P(1)$ — Л; $P(2)$ — И; $Q(1, 1)$ — И; $Q(1, 2)$ — И; $Q(2, 1)$ — Л; $Q(2, 2)$ — И.

Оцените эту формулу.

Решение. Если $x = 1$, то

$$P(x) \to Q(f(x), a) = P(1) \to Q(f(1), a) = P(1) \to Q(2, 1) = \text{Л} \to \text{И} = \text{И}.$$

Если $x = 2$, то

$$P(x) \to Q(f(x), a) = P(2) \to Q(f(2), a) = P(2) \to Q(1, 1) = \text{И} \to \text{И} = \text{И}.$$

Так как $P(x) \to Q(f(x), a)$ истинно для всех x из области M, то формула $(\forall x)(P(x) \to Q(f(x), a))$ истинна в заданной интерпретации.

Задача 9. Используя интерпретации из задачи 8, оцените следующие формулы:

(а) $(\exists x)(P(f(x)) \wedge Q(x, f(a)))$; **(б)** $(\exists x)(P(x) \wedge Q(x, a))$;

(в) $(\forall x)(\exists y)(P(x) \wedge Q(x, y))$.

Ответ: **(а)** — истина; **(б)** — ложь; **(в)** — ложь.

Задача 10. Рассмотрим формулы

$$F_1 : (\forall x)(P(x) \to Q(x)); \quad F_2 : P(a).$$

Докажите, что $Q(a)$ есть логическое следствие формул F_1 и F_2.

Решение. Рассмотрим любую интерпретацию. Из определения следует, что формула есть логическое следствие формул F_1 и F_2 тогда и только тогда, когда для каждой интерпретации данная формула истинна, если $F_1 \wedge F_2$ — истина.

Рассмотрим любую интерпретацию, которая удовлетворяет

$$(\forall x)(P(x) \to Q(x)) \wedge P(a).$$

В этой интерпретации $P(a)$ — истина.

Пусть $Q(a)$ не является истиной в этой интерпретации, тогда $\overline{P(a)} \vee Q(a) = P(a) \to Q(a)$ — не истина. Это значит, что $(\forall x)(P(x) \to Q(x))$ — ложь, что невозможно. Следовательно, $Q(a)$ должно быть истиной в каждой интерпретации, которая удовлетворяет $(\forall x)(P(x) \to Q(x)) \wedge P(a)$. Это означает, что $Q(a)$ есть следствие из F_1 и F_2.

Задача 11. Докажите следующее:

1) $(\forall x)P(x) \wedge (\exists y)\overline{P(y)}$ — противоречива (невыполнима), т. е. не существует интерпретации, удовлетворяющей этой формуле.

2) $P(a) \to \overline{(\exists x)P(x)}$ — непротиворечива (выполнима), т. е. существует такая интерпретация, что формула истина в этой интерпретации.

3) а) $(\forall x)P(x) \to (\exists y)P(y)$,

б) $(\forall x)P(x) \vee (\exists y)\overline{P(y)}$ общезначимы, т. е. не существует никакой интерпретации, которая удовлетворяет формулам.

Задача 12. Получите нормальную форму для формулы

$$(\forall x)(\forall y)((\exists z)P(x, z) \wedge P(y, z) \to (\exists u)Q(x, y, u)).$$

Решение.

$$\begin{aligned}
&(\forall x)(\forall y)((\exists z)P(x, z) \wedge P(y, z) \to (\exists u)Q(x, y, u)) = \\
&\quad = (\forall x)(\forall y)\left(\overline{(\exists z)P(x, z) \wedge P(y, z)} \vee (\exists u)Q(x, y, u)\right) = \\
&\quad = (\forall x)(\forall y)\left((\forall z)(\overline{P(x, z) \wedge P(y, z)}) \vee (\exists u)Q(x, y, u)\right) = \\
&\quad = (\forall x)(\forall y)\left((\exists z)(\overline{P(x, z)} \wedge \overline{P(y, z)}) \vee (\exists u)Q(x, y, u)\right) = \\
&\quad\quad = (\forall x)(\forall y)(\forall z)(\exists u)\left(\overline{P(x, z)} \vee \overline{P(y, z)} \vee Q(x, y, u)\right)
\end{aligned}$$

(здесь выносим кванторы влево).

Следовательно, последняя формула есть нормальная форма первой формулы.

Задача 13. Преобразуйте следующие формулы в нормальную форму:

(1) $(\forall x)\,(P(x) \to (\exists y)Q(x,y))$;

(2) $(\exists x)\Big(\overline{(\exists y)P(x,y)} \to ((\exists z)Q(z) \to R(x))\Big)$;

(3) $(\forall x)(\forall y)\Big((\exists z)P(x,y,z) \land ((\exists u)Q(x,u) \to (\exists v)Q(y,v))\Big)$.

Контрольные вопросы

1. Что называется предикатом? Приведите примеры предикатов.

2. Какой предикат называется разрешимым, тождественно истинным. Тождественно ложным?

3. Перечислите операции, которые можно осуществить над предикатами. Как применяются предикаты в алгебре? Что такое множество истинности предиката?

4. Из чего состоит алфавит логики предикатов? Что такое квантор?

5. Что называется формулой логики предикатов?

6. Сформулируйте основные правила построения формул.

7. В чем состоит смысл термина «интерпретация» в логике предикатов?

8. Сформулируйте основные правила перехода к новым равносильным формулам.

9. Какая формула называется непротиворечивой, противоречивой, общезначимой?

10. Какая формула называется приведенной? Что такое приведенная форма?

11. Какая формула называется нормальной формой? Сформулируйте алгоритм приведения формулы к нормальной форме.

12. Что называют исчислением предикатов?

13. Сформулируйте аксиомы исчисления предикатов.

Часть 5

Элементы теории графов

Лекции 11–12

Теория графов — область дискретной математики, особенностью которой является геометрический подход к изучению объектов. Теория графов и связанные с ней методы исследования используются на разных уровнях во всей современной математике. Особенно широкое применение методы теории графов находят в таких областях прикладной математики, как программирование, теория конечных автоматов, в решении вероятностных и комбинаторных задач.

1. Некоторые основные понятия

Во многих прикладных задачах изучаются системы связей между различными объектами. Объекты называются вершинами и отмечаются точками или кружочками, а связи между вершинами — отрезками, соединяющими пары точек, и эти отрезки называются ребрами. Рассмотрение таких систем и приводит к понятию графа.

Граф представляет собой непустое конечное множество вершин V и множество ребер E, оба конца которых принадлежат множеству V. Обозначать граф будем $G(V, E) = \langle V, E\rangle$, $V \neq \varnothing$, $E \subset V \times V$.

При изображении графов на рисунках или схемах ребра могут быть прямолинейными или криволинейными; длины ребер и расположение вершин произвольны.

Вершины, которые не принадлежат ни одному ребру, называются *изолированными*.

Обозначать вершины будем буквами $v_1, v_2, v_3, \ldots, v_n$, т.е. $V = \{v_1, v_2, \ldots, v_n\}$.

Пусть v_1, v_2 — вершины, $e = \langle v_1, v_2\rangle$ — соединяющее их ребро. Тогда вершина v_1 и ребро e *инцидентны*. Вершина v_2 и ребро e также *инцидентны*. Два ребра, инцидентные одной вершине, называются *смежными*. Две вершины, инцидентные одному ребру, также называются *смежными*.

Число вершин графа G обозначим p, а число ребер обозначим q

$$p : p(G) = |V|; \quad q : q(G) = |E|.$$

Граф называется полным, если каждые две различные вершины его соединены одним и только одним ребром.

2. Степень вершины

Вершины в графе могут отличаться друг от друга тем, скольким ребрам они принадлежат.

Степенью вершины называется число ребер графа, которым принадлежит эта вершина. Степень графа еще называют его *валентностью* и обозначают $d(v)$. Вершина графа, для которой $d(v) = 0$, является *изолированной*, для которой $d(v) = 1$ — *висячей*.

Вершина называется *нечетной*, если $d(v)$ — нечетное число. Вершина называется *четной*, если $d(v)$ — четное число. Степень каждой вершины полного графа на единицу меньше числа его вершин.

В графе $G(V, E)$ сумма степеней всех его вершин — число четное, равное удвоенному числу ребер графа. Число нечетных вершин любого графа четно. Во всяком графе с n вершинами, где $n \geqslant 2$, всегда найдутся, по меньшей мере, две вершины с одинаковыми степенями.

Если в графе с n вершинами ($n \geqslant 2$) в точности две вершины имеют одинаковую степень, то в этом графе всегда найдется либо в точности одна вершина степени 0, либо в точности одна вершина степени $n - 1$.

3. Маршруты, цепи, циклы

Маршрутом в графе называется чередующаяся последовательность вершин и ребер, в которой любые два соседних элемента инцидентны:

$$v_0, e_1, v_1, e_2, v_2, \ldots, e_k, v_k.$$

Если $v_0 = v_k$, то маршрут замкнут, в противном случае открыт.

Если все ребра различны, то маршрут называется цепью. Если все вершины (а значит, и ребра) различны, то маршрут называется простой цепью. В цепи $v_0, e_1, v_1, e_2, v_2, \ldots, e_k, v_k$ вершины v_0 и v_k называются концами цепи, т. е. цепь концами v_0 и v_k соединяет вершины v_0 и v_k. Цепь, соединяющая вершины v_0 и v_k, обозначается $\langle v_0, v_k \rangle$. Очевидно, что если есть цепь, соединяющая вершины v_0 и v_k, то есть и простая цепь, соединяющая эти вершины. Замкнутая цепь называется *циклом*; замкнутая простая цепь называется *простым циклом*. Число циклов в графе $G(V, E)$ обозначается $z(G)$. Граф без циклов называется *ациклическим*.

Длиной маршрута называется количество ребер в нем (с повторениями). Если маршрут $M = v_0, e_1, v_1, e_2, v_2, ..., e_k, v_k$, то длина маршрута M равна k, обозначается $|M| = k$.

4. Связность графа

Две вершины графа называются связными, если существует соединяющая их простая цепь. В противном случае две вершины называются несвязными.

Граф называется связным, если каждые две вершины его связные. Граф называется несвязным, если хотя бы две его вершины несвязные.

5. Ориентированные графы

Если элементы множества E графа $G(V, E)$ — упорядоченные пары, то граф называется *ориентированным* или *орграфом*.

Ребро e графа G называется ориентированным, если одну вершину считают началом ребра, а другую — концом, на рисунке его изображают стрелкой между вершинами. Таким образом, граф, все ребра которого ориентированы, называется ориентированным графом.

Одна и та же вершина ориентированного графа может служить началом для одних ребер и концом для других, поэтому различают две степени вершины: степень выхода и степень входа.

Степенью выхода вершины орграфа называется число *выходящих* из вершины ребер.

Степенью входа вершины орграфа называется число *входящих* в вершину ребер.

В орграфах в зависимости от сочетания степеней входа и выхода для данной вершины рассматриваются три случая.

Изолированной вершиной называется вершина, у которой и степень входа и степень выхода равны 0.

Источником называется вершина, степень выхода которой положительна, а степень входа равна 0.

Стоком называется вершина, степень входа которой положительна, а степень выхода равна 0.

Путем в ориентированном графе называется последовательность ориентированных ребер, т. е. для орграфов цепь называется путем.

Простым путем в ориентированном графе называется путь, в котором ни одна вершина не содержится более одного раза.

Замкнутый путь в ориентированном графе называется ориентированным циклом или контуром.

Длиной пути называется число ребер в этом пути.

Полным ориентированным графом называется граф, каждая пара вершин которого соединена в точности одним ориентированным ребром. *Замечание.* Если с каждого ребра полного ориентированного графа снять направление, то образуется полный граф с неориентированными ребрами, т. е. неориентированный полный граф.

Всякий полный ориентированный граф с n вершинами имеет простой ориентированный путь, проходящий через все вершины графа.

Петлей называется ребро, у которого начальная и конечная вершины совпадают. Петля обычно считается *неориентированной.*

Мультиграфом называется граф, в котором пара вершин соединяется несколькими различными ребрами. Для ориентированного мультиграфа вершины v_i и v_j могут соединяться несколькими ребрами в каждом из направлений.

6. Изоморфизм графов

Два графа $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ называются изоморфными, если между множествами их вершин существует биективное (взаимнооднозначное) соответствие, такое, что вершины соединены ребрами в одном из графов в том и только в том случае, когда соответствующие им вершины соединены в другом графе. Если ребра графа ориентированы, то их направление в изоморфных графах должно совпадать. Изоморфизм графов есть отношение эквивалентности, так как обладает свойствами рефлексивности, симметричности, транзитивности. Для того чтобы граф G_1 был изоморфен графу G_2, необходимо и достаточно существования такой подстановки, которая бы установила взаимнооднозначное соответствие между вершинами графа, а также между их ребрами.

При замене графа любым ему изоморфным все свойства графа сохраняются. Строго говоря, графы, отличающиеся только нумерацией вершин, являются изоморфными.

Алгоритм распознавания изоморфизма двух графов $G_1(X;E)$ и $G_2(Y;E)$.

1. Подсчитываем число вершин каждого графа (число вершин должно совпадать, в противном случае графы неизоморфные).

2. Выписываем все элементы обоих графов в естественной упорядоченности и определяем пары $(x_i; x_j)$ и $(y_i; y_j)$ для каждого элемента, где x_i; y_i — число исходов для каждой вершины графов G_1 и G_2, а x_j; y_j — число заходов для соответствующих графов.

3. Для каждого элемента x графа G_1 ищем такой элемент у графа G_2, что выполняется условие: число исходов x совпадает с числом исходов y, и число заходов x совпадает с числом заходов y. Найденные элементы x и y соединяем ребром, т. е. строим граф соответствия (если соответствия нет, то графы не изоморфны).

4. Выписываем подстановку, которая переводит граф G_1 в граф G_2.

7. Плоские графы

Граф $G(V,E)$ называется плоским, если на плоскости его можно изобразить так, что все пересечения его ребер являются вершинами графа $G(V,E)$.

В качестве характеристики плоского представления графа вводится понятие грани.

Гранью в плоском представлении графа называется часть плоскости, ограниченная простым циклом и не содержащая внутри других циклов.

8. Операции над графами

В ряде задач теории графов используются двуместные операции над графами.

Рассмотрим графы $G_1(V_1,E_1)$ и $G_2(V_2,E_2)$.

а) *Дополнением* графа $G_1(V_1,E_1)$ называется граф $\overline{G_1}(V_1,\overline{E_1})$, множеством вершин которого является множество V_1, а множеством его ребер является множество $\overline{E_1} = \{e \in V_1 \times V_1 : e \notin E_1\}$.

б) *Объединением* графов $G_1(V_1,E_1)$ и $G_2(V_2,E_2)$ при условии, что $V_1 \cap V_2 = \varnothing$; $E_1 \cap E_2 = \varnothing$, называется граф $G_1(V_1,E_1) \cup G_2(V_2,E_2)$, множеством вершин которого является множество $V_1 \cup V_2$, а множеством его ребер является множество $E_1 \cup E_2$.

в) *Пересечением* графов $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ называется граф $G_1(V_1, E_1) \cap G_2(V_2, E_2)$, множеством вершин которого является множество $V_1 \cap V_2$, а множеством его ребер — множество $E_1 \cap E_2$.

г) *Суммой по модулю два* графов $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ при условии, что $V_1 \cap V_2 = \varnothing$; $E_1 \cap E_2 = \varnothing$, называется граф $G_1(V_1, E_1) \oplus G_2(V_2, E_2)$, множеством вершин которого является множество $V_1 \cup V_2$, а множеством его ребер — множество $E_1 \oplus E_2$. Другими словами, этот граф не имеет изолированных вершин и состоит только из ребер, присутствующих либо в первом графе, либо во втором графе, но не в обоих графах одновременно.

Легко убедиться в том, что операции: объединение, пересечение и сумма по модулю два обладают свойством *коммутативности*.

9. Способы задания графов

Существует три эквивалентных способа задания графов: аналитический, геометрический и матричный. Рассмотрим каждый из них.

Аналитический способ задания графов

Граф $G(V, E)$ задан, если задано множество элементов V и отображение E множества V в V. Отображение E может быть как однозначным, так и многозначным. В общем случае на V и E никаких ограничений не накладывается.

Пусть дано множество $V = \{v_1, v_2, \dots, v_n\}$, которое имеет мощность $|V| = n$. Вместо $V = \{v_1, v_2, \dots, v_n\}$ иногда пишут

$$V = \{v_i\}, \quad i \in \{1, 2, \dots, n\}.$$

Для того чтобы задать отображение E на V или, что то же самое, отображение V в V, необходимо каждому элементу $v_i \in V$ поставить в соответствие некоторое подмножество множества V, которому соответствует отображение E. Это подмножество обозначают через E_{v_i}. Поэтому $E_{v_i} \subset V$. Совокупность двух объектов: множества V и отображение E на V задает некоторый граф.

Другой формой аналитического способа задания является задание графа как совокупности множества элементов V и подмножества множества упорядоченных пар $\langle v_i, v_j \rangle \in V \times V$. Подмножество множества пар $\langle v_i, v_j \rangle$ декартова произведения $V \times V$ эквивалентно бинарному отношению R, заданному на множестве V. Поэтому множество V и бинарное отношение R на множестве V также определяет некоторый граф G.

Геометрический способ задания графов

Множество элементов V графа G изображают кружками, это множество вершин. Каждую вершину $v_i \in V$ соединяют линиями с теми вершинами $v_i \in V$, для которых выполняется условие $v_i \in E_{v_i}$. Множество линий, которое соответствует множеству упорядоченных пар $\langle v_i, v_j \rangle$, есть множество ребер графа.

Матричный способ задания графов

Квадратная матрица

$$\begin{Vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{Vmatrix},$$

элементами которой являются нули и единицы, а также некоторое число m, называется *матрицей смежности* графа $G(V, E)$ тогда и только тогда, когда ее элементы образуются по следующему правилу: элемент $a_{i,j}$, стоящий на пересечении v_i-й строки и v_j-го столбца, равен единице, если имеется ребро, идущее из вершины v_i в вершину v_j, и $a_{i,j}$ равен нулю в противном случае. Элемент $a_{i,i}$ равен единице, если при вершине v_i имеется петля, и равен нулю в противном случае. Элемент $a_{i,j}$ равен некоторому числу m, где m — число ребер графа, идущее из вершины v_i в вершину v_j.

Пусть $v_i, \dots, v_n$ — вершины, а $e_1, \dots, e_m$ — ребра некоторого ориентированного графа $G(V, E)$. Матрица размером $(m \times n)$, где

$$a_{i,j} = \begin{cases} 1, \text{ если } e_j \text{ исходит из } v_i; \\ -1, \text{ если } e_j \text{ заходит в } v_i; \\ 0, \text{ если } e_j \text{ не инцидентна } v_i \end{cases}$$

называется матрицей инцидентности для ребер ориентированного графа.

Таким образом, если граф $G(V, E)$ задан одним из указанных способов: аналитическим, геометрическим или матричным, всегда можно перейти к любому другому способу задания. Результаты, которые получены на одном языке, можно интерпретировать в другом. Наиболее часто для задания графа используется аналитический и матричный способы, а геометрический способ служит для иллюстрации полученных результатов.

10. Некоторые типы графов

Эйлеровы графы

К задачам на эйлеровы графы относятся головоломки, в которых требуется вычертить на плоскости одним росчерком замкнутые кривые, обводя каждый участок в точности один раз. Введем следующие понятия.

Эйлеровым путем в графе называется путь, содержащий все ребра графа.

Эйлеровым циклом или *эйлеровой цепью* называется цикл, содержащий все ребра графа и притом по одному разу.

Граф, обладающий эйлеровым циклом, называется эйлеровым графом.

Замкнутую линию, если ее можно начертить, не отрывая карандаша от бумаги, проходя при этом каждый участок в точности один раз, принято называть уникурсальной.

Рисунок графа, обладающего эйлеровым путем или эйлеровым циклом, является уникурсальной линией.

Докажем следующие две теоремы.

Теорема 1. Если граф $G(V, E)$ обладает эйлеровым циклом, то он связный и все его вершины четные.

Доказательство. Связность графа следует из определения эйлерова цикла. Эйлеров цикл содержит каждое ребро и притом только один раз, поэтому, сколько раз эйлеров путь приведет конец карандаша в вершину, столько и выведет, причем уже по другому ребру. Следовательно, степень каждой вершины графа должна состоять из двух одинаковых слагаемых: одно — результат подсчета входов в вершину, другое — выходов.

Теорема 2. Если граф $G(V, E)$ связный и все его вершины четные, то он обладает эйлеровым циклом.

Доказательство. Если начать путь из произвольной вершины графа $G(V, E)$, то найдется цикл, содержащий все ребра графа. Пусть v_i — произвольная вершина. Из v_i начнем путь по l по одному из ребер и продолжим его, проходя каждый раз по новому ребру. Все вершины графа имеют четные степени, поэтому если l есть «выход» из v_i, то должен быть и «вход» в v_i, также как и для любой другой вершины. И если есть «вход» в вершину, то должен быть и «выход». Так как число ребер конечно, то этот путь должен окончиться, причем в вершине v_i. Если путь, замкнувшийся в v_i, проходит через все ребра графа, то мы получим искомый эйлеров цикл.

Для построения эйлерова цикла в связном графе со всеми вершинами четной степени применяется следующий алгоритм:

1. Выйти из произвольной вершины v_i. Каждое пройденное ребро зачеркнуть. Если путь l_1 замыкается в v_i и проходит через все ребра графа, то получим искомый эйлеров цикл.

2. Если остались непройденные ребра, то должна существовать вершина v_2, принадлежащая l_1 и ребру, не вошедшему в l_1.

3. Так как v_2 — четная, то число ребер, которым принадлежит v_2 и которые не вошли в путь l_1, тоже четно. Начнем новый путь l_2 из v_2 и используем только ребра, не принадлежащие l_1. Этот путь кончится в v_2.

4. Объединим теперь оба цикла: из v_i пройдем по пути l_1 к v_2, затем по l_2 и, вернувшись в v_2, пройдем по оставшейся части l_1 обратно в v_i.

5. Если снова найдутся ребра, которые не вошли в путь, то найдем новые циклы. Так как число ребер и вершин конечно, то процесс закончится.

Таким образом, замкнутую фигуру, в которой все вершины четные, можно начертить одним росчерком без повторений и начиная с любой точки.

На практике эйлеровым графом может быть план выставки; это позволяет расставить указатели маршрута, чтобы посетитель смог пройти по каждому залу в точности по одному разу.

Гамильтоновы графы

Граф, обладающий гамильтоновым циклом, называется *гамильтоновым графом*.

Гамильтоновым циклом, или путем в графе, называется цикл, или путь, проходящий через каждую вершину графа в точности по одному разу.

Эйлеровы и гамильтоновы пути сходны по способу задания. Первые содержат все ребра, и притом по одному разу, вторые — все вершины по одному разу. Но, несмотря на внешнее сходство, задачи их отыскания резко отличаются по степени трудности. Для решения вопроса о существовании эйлерова цикла в графе достаточно выяснить, все ли его вершины четные.

Критерий же существования гамильтонова цикла на произвольном графе еще не найден.

Однако есть несколько достаточных условий существования гамильтоновых циклов в графе:

1. Всякий полный граф является гамильтоновым, так как он содержит простой цикл, которому принадлежат все вершины данного графа.

2. Если граф, помимо простого цикла, проходящего через все его вершины, содержит и другие ребра, то он также является гамильтоновым.

3. Если граф имеет один гамильтонов цикл, то он может иметь и другие гамильтоновы циклы.

Седьмое практическое занятие по теме «Графы»

Задача 1. Изобразите графически:

1) неориентированное и ориентированное ребра;

2) неориентированный граф $G(V, E)$, заданный множеством $V = \{v_0, v_1, v_2, v_3, v_4, v_5\}$ и соответствием: $E(v_0) = \{v_1, v_2\} = \{v_0, v_2, v_4\}$; $E(v_1) = \{v_0, v_2, v_4\}$; $E(v_2) = \{v_0, v_1, v_5\}$; $E(v_3) = \{v_4\}$; $E(v_5) = \{v_2\}$;

3) плоский граф;

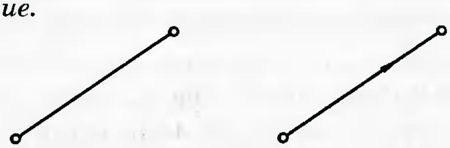
4) полный неориентированный граф на трех, четырех и пяти вершинах;

5) неполный ориентированный граф на пяти вершинах;

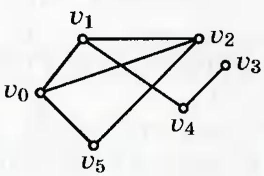
6) петлю графа;

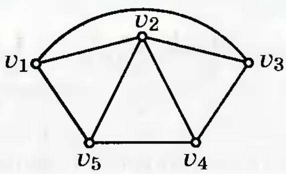
7) неориентированный и ориентированный мультиграф.

Решение.

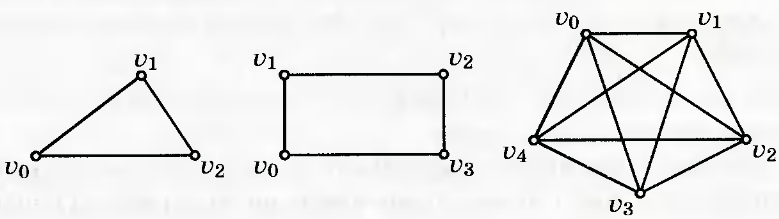
1) 

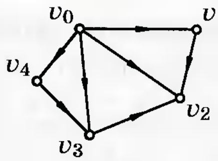
неориентированное ребро   ориентированное ребро

2) 

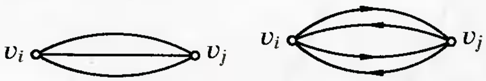
3) 

Плоский граф

4) 

5) 

6) 

7) 

Мультиграф Ориентированный мультиграф

Задача 2. Докажите, что в полном графе с n вершинами $\frac{n(n-1)}{2}$ ребер.

Решение. Каждой вершине в полном графе с n вершинами принадлежит $n-1$ ребро, но в произведении $n(n-1)$ каждое ребро учтено дважды (так как одно ребро инцидентно двум вершинам). Следовательно, число ребер в полном графе с n вершинами равно $\frac{n(n-1)}{2}$.

Задача 3. Девять шахматистов проводят турнир в один круг (каждый из участников должен сыграть с остальными по одному разу). Покажите, что в любой момент найдутся два шахматиста, сыгравшие одинаковое число партий.

Решение. Переведем условие задачи на язык графов. Каждому шахматисту поставим в соответствие вершину графа, соединим ребрами попарно вершины, соответствующие шахматистам, уже сыгравшим между собой партию. Получим граф с девятью вершинами. Степени его вершин равняются числу партий, сыгранных соответствующими игроками. Покажем, что во всяком графе с девятью вершинами всегда найдутся хотя бы две вершины одинаковой степени.

Каждая вершина графа с девятью вершинами может иметь степень 0, 1, 2, 3, 4, 5, 6, 7, 8. Предположим, что существует граф G, все вершины которого имеют разную степень, т. е. каждое из чисел последовательности 0, 1, 2, 3, 4, 5, 6, 7, 8 является степенью одной и только одной из его вершин. Но этого не может быть, так как если в графе есть вершина v_i степени 0, то в нем найдется вершина v_j со степенью 8. Эта вершина v_j должна быть соединена ребрами со всеми остальными вершинами графа, в том числе и с v_i, поэтому степень вершины v_i не может равняться 0. Таким образом, в графе с девятью вершинами не могут быть одновременно вершины степени 0 и 8. Следовательно, найдутся хотя бы две вершины, степени которых равны между собой. Таким образом, доказано, что в любой момент найдутся хотя бы два шахматиста, сыгравшие одинаковое число партий.

Задача 4. (Для самостоятельного решения.)

Девять человек проводят шахматный турнир в один круг. К некоторому моменту выясняется, что двое сыграли одинаковое число партий. Докажите, что тогда либо один участник еще не сыграл ни одной партии, либо один сыграл все партии.

Задача 5. Может ли так случиться, что в одной компании из шести человек каждый знаком с двумя и только с двумя другими?
Решение. Участников этой компании изобразим вершиной графа (рис. 5.1), а отношение знакомства между двумя участниками — ребром. Изобразим графы, которые могут соответствовать такой компании.

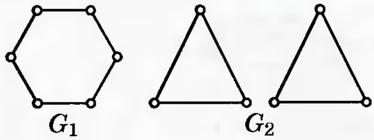


Рис. 5.1

Про граф G_1 говорят, что он связный, так как из каждой вершины по ребрам можно попасть в любую другую. Делаем вывод, что в этом случае каждый через своих знакомых может познакомиться со всеми остальными.

Про граф G_2 говорят, что он несвязный, так как состоит из двух простых циклов. Делаем вывод, что граф соответствует двум компаниям, участники одной из них могут быть не знакомы с участниками другой.

Задача 6. Из пункта A в пункт B выехали пять машин одной марки разного цвета: белая, черная, красная, синяя, зеленая. Черная едет впереди синей, зеленая — впереди белой, но позади синей, красная впереди черной. Какая машина едет первой и какая последней?
Решение. Решаем задачу, построив ориентированный граф для отношения f: «x едет сзади y». На плоскости отметим пять точек, соответствующих каждой машине, и обозначим их первой буквой цвета машины (рис. 5.2).

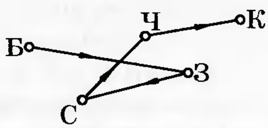


Рис. 5.2

Анализируя граф, получаем следующий порядок движения: красная, черная, синяя, зеленая, белая.

Задача 7. Пусть даны графы $G_1(X, E)$ и $G_2(Y, E)$, изображенные на рис. 5.3.

Установите, изоморфны ли данные графы.
Решение. Для доказательства того, что граф G_1 изоморфен графу G_2 необходимо и достаточно выполнение условия: найти такую подстановку, которая граф G_1 переводит в граф G_2.

Запишем элементы $x \in X$ и $y \in Y$ с соответствующими им парами чисел, где первое число — число исходов из вершины, а второе — число заходов в

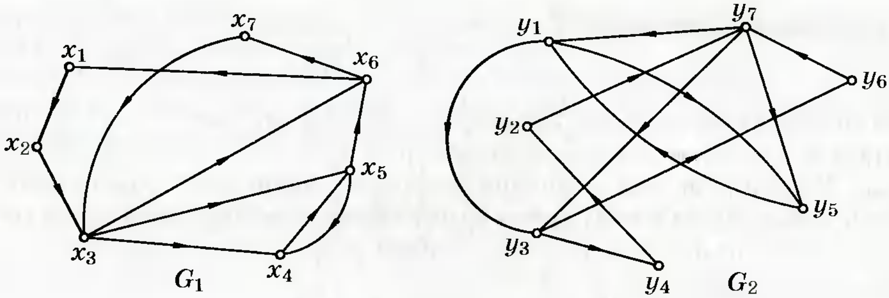


Рис. 5.3

вершину. Далее определим частичную подстановку, соединяя вершины x_i и y_i с одинаковыми числами (рис. 5.4).

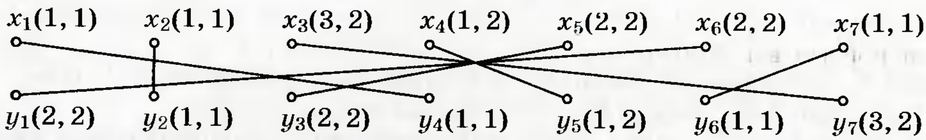


Рис. 5.4

В результате получим подстановку

$$\begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ y_4 & y_2 & y_7 & y_5 & y_3 & y_1 & y_6 \end{pmatrix}.$$

Следовательно, графы G_1 и G_2 изоморфны.

Задача 8. Для неориентированного графа, изображенного на рис. 5.5, постройте матрицу смежности и матрицу инцидентности.

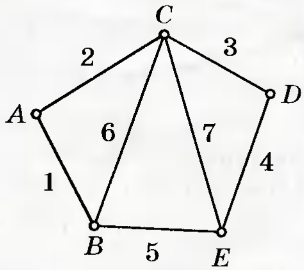


Рис. 5.5

Решение. Матрица смежности

$$\begin{array}{c} A \\ B \\ C \\ D \\ E \\ \end{array}\begin{array}{c}\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} \\ \begin{matrix} A & B & C & D & E \end{matrix}\end{array}$$

Матрица инцидентности

$$\begin{array}{c} A \\ B \\ C \\ D \\ E \\ \end{array}\begin{array}{c}\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \\ \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{matrix}\end{array}$$

Задача 9. Задан граф $G(V, E)$, где $V = \{v_1, v_2, v_3, v_4, v_5\}$; $E_{v_1} = \{v_1, v_3, v_5\}$; $E_{v_2} = \varnothing$; $E_{v_3} = \{v_1, v_2, v_5\}$; $E_{v_4} = \{v_1\}$; $E_{v_5} = \{v_1, v_2, v_3, v_4, v_5\}$.

1. Задайте граф с помощью бинарного отношения, т. е. совокупности множества V и подмножества множества упорядоченных пар $\langle v_i, v_j \rangle \in V \times V$.

2. Изобразите орграф на рисунке.

3. Постройте матрицу смежности.

Решение.

1. $V = \{v_1, v_2, v_3, v_4, v_5\}$.

Множество пар: $\{\langle v_1, v_1\rangle; \langle v_1, v_3\rangle; \langle v_1, v_5\rangle; \langle v_3, v_1\rangle; \langle v_3, v_2\rangle; \langle v_3, v_5\rangle; \langle v_4, v_1\rangle; \langle v_5, v_1\rangle; \langle v_5, v_2\rangle; \langle v_5, v_3\rangle; \langle v_5, v_4\rangle; \langle v_5, v_5\rangle\}$.

2. См. рис. 5.6.

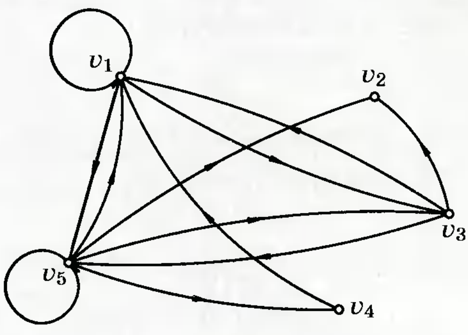


Рис. 5.6

3. $\begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$

Задача 10. Дано множество $V = \{1, 2, 3, 4, 5\}$. На этом множестве задано отношение f: $x > y$. Постройте орграф данного отношения.

Решение. Для того чтобы построить орграф данного отношения f: $x > y$, изобразим все элементы множества V точками на плоскости и проведем стрелку от каждого большего числа к меньшему (рис. 5.7).

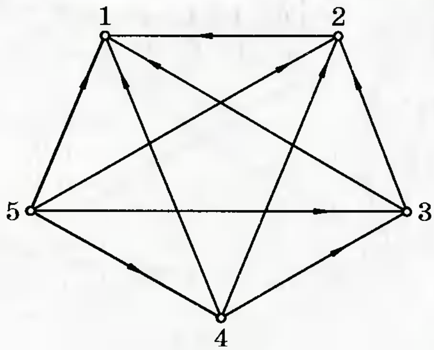


Рис. 5.7

Задача 11. Дана матрица

$$\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ \\ \end{array} \begin{array}{c} \begin{bmatrix} 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 3 & 1 & 0 & 0 \end{bmatrix} \\ \begin{matrix} 1 & 2 & 3 & 4 \end{matrix} \end{array}$$

Постройте орграф, для которого данная матрица является матрицей смежности. Найдите матрицу инцидентности орграфа.

Решение. Для построения орграфа его вершине однозначно сопоставим точку на плоскости. Данная матрица смежности имеет четыре строки и четыре столбца, следовательно, в орграфе четыре вершины: 1, 2, 3, 4.

Проанализируем элементы матрицы:

$a_{11} = 0$ — при вершине 1 нет петель;
$a_{12} = 2$ — из вершины 1 выходят две стрелки к вершине 2;
$a_{13} = 0$ — из 1 не выходит ни одной стрелки к вершине 3;
$a_{14} = 0$ — из 1 не выходит ни одной стрелки к вершине 4;
$a_{21} = 0$ — из 2 не выходит ни одной стрелки к вершине 1;
$a_{22} = 0$ — при 2 не петель;
$a_{23} = 1$ — из 2 выходит одна стрелка к вершине 3;
$a_{24} = 0$ — из 2 не выходит ни одной стрелки к вершине 4;
$a_{31} = 1$ — из 3 выходит одна стрелка к вершине 1;
$a_{32} = 0$ — из 3 не выходит ни одной стрелки к вершине 2;
$a_{33} = 0$ — при 3 нет петель;
$a_{34} = 1$ — из 3 выходит одна стрелка к вершине 4;
$a_{41} = 3$ — из 4 выходит 3 стрелки к вершине 1;
$a_{42} = 1$ — из 4 выходит одна стрелка к вершине 2;

$a_{43} = 0$ — из 4 не выходит ни одной стрелки к вершине 3;
$a_{44} = 0$ — при 4 нет петель.
Строим орграф (рис. 5.8).

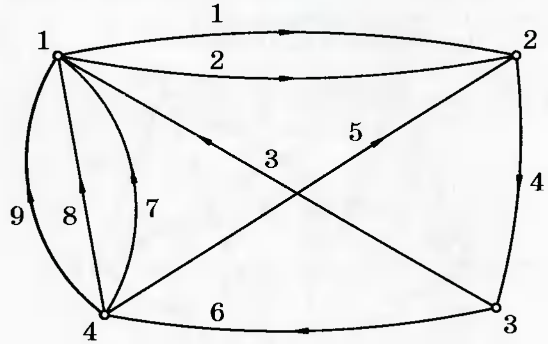


Рис. 5.8

Для построенного графа запишем матрицу инцидентности:

$$\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ \end{array}\begin{array}{c}\begin{bmatrix} 1 & 1 & -1 & 0 & 0 & 0 & -1 & -1 & -1 \\ -1 & -1 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 1 & 1 & 1 \end{bmatrix} \\ \begin{matrix} 1 & \;\;2 & \;\;3 & \;\;4 & \;\;5 & \;\;6 & \;\;7 & \;\;8 & \;\;9 \end{matrix}\end{array}$$

Здесь четыре строки по числу вершин и 9 столбцов по числу дуг.

Задача 12. Пусть заданы два графа $G_1(V_1, E_1)$, $G_2(V_2, E_2)$ (рис. 5.9).

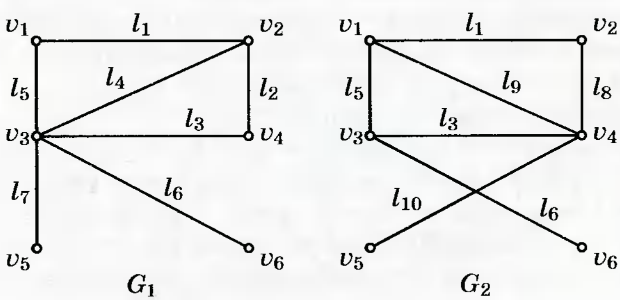


Рис. 5.9

Изобразите геометрически объединение графов $G_1 \cup G_2$; пересечение графов $G_1 \cap G_2$ и сумму по модулю два $G_1 \oplus G_2$.

Решение. Объединение графов G_1 и G_2: $G_1 \cup G_2$ (рис. 5.10).

Пересечение графов G_1 и G_2: $G_1 \cap G_2$ (рис. 5.11).

Сумма по модулю два графов G_1 и G_2: $G_1 \oplus G_2$ (рис. 5.12).

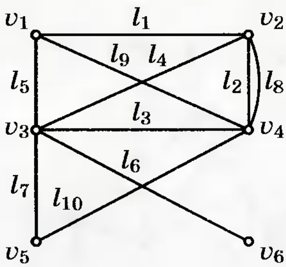


Рис. 5.10

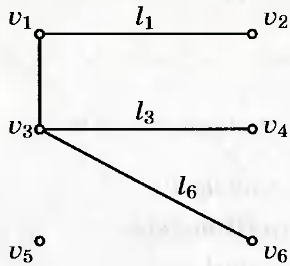


Рис. 5.11

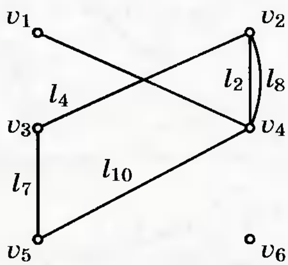


Рис. 5.12

Задача 13. Найдите эйлеров цикл в эйлеровом графе (рис. 5.13).

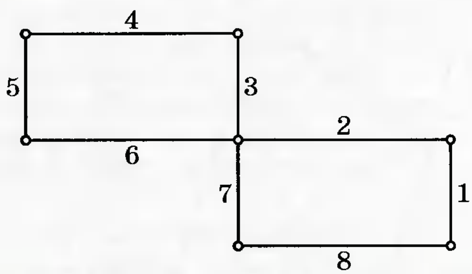


Рис. 5.13

Решение. После выбора вершины *a* и прохождении ребер 1 и 2 имеются три возможности выбора: ребра 3, 6 или 7. Выбираем ребро 3 или 6. Например, ребро 3. Далее обходим оставшиеся ребра и получаем эйлеров цикл 1, 2, 3, 4, 5, 6, 7, 8.

Задача 14. Найдите цикл, содержащий все вершины додекаэдра, причем в точности по одному разу каждую.

Решение. Этот цикл: 1, 2, 3, 4, 5, 6, 19, 18, 14, 15, 16, 17, 7, 8, 9, 10, 11, 12, 13, 20. Этот цикл называется гамильтоновым циклом (рис. 5.14).

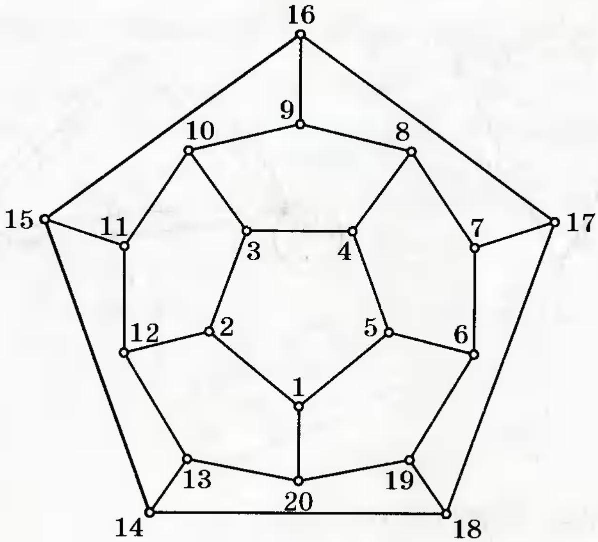


Рис. 5.14

Задача 15. (Для самостоятельного решения.)

Покажите, что в изображенном графе нет гамильтонова пути, но в графе, полученном из него удалением одной из вершин, имеется гамильтонов цикл (рис. 5.15).

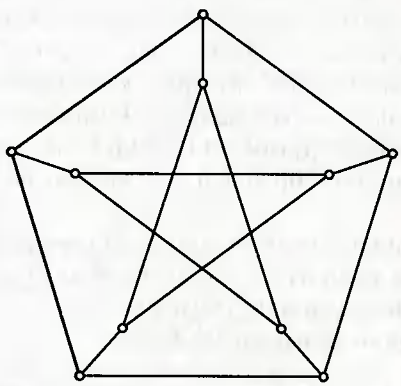
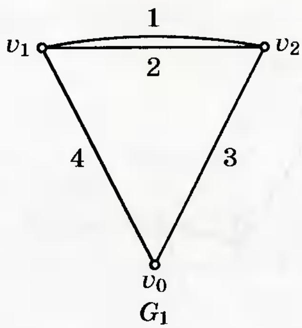


Рис. 5.15

Задача 16. (Для самостоятельного решения.)

Даны графы G_1 и G_2 (рис. 5.16). Постройте матрицы смежности.



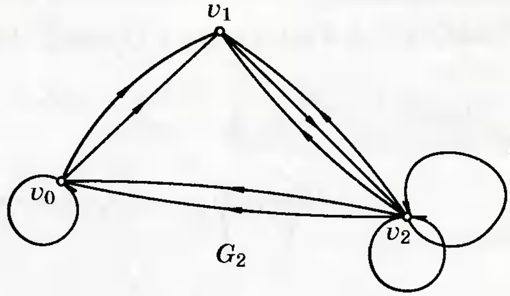


Рис. 5.16

Контрольные вопросы

1. Что называется графом? Ориентированным графом? Приведите примеры.

2. Что такое степень вершины?

3. Перечислите основные понятия, связанные с неориентированными графами.

4. Перечислите основные понятия, связанные с орграфами.

5. Перечислите способы задания графов.

6. В чем состоит аналитический способ задания графа?

7. В чем состоит геометрический способ задания графа?

8. В чем состоит матричный способ задания графа?

9. Какая матрица называется матрицей смежности графа?

10. Какая матрица называется матрицей инцидентности графа?

11. Что называется маршрутом, циклом и цепью графа?

12. Сформулируйте понятие связности графа. Какой граф называют связным?

13. Какие два графа называются изоморфными?

14. Сформулируйте алгоритм изоморфизма двух графов.

15. Перечислите операции над графами.

16. Дайте определение эйлерова графа.

17. Сформулируйте алгоритм построения эйлерова цикла.

18. Какой граф называют гамильтоновым?

Часть 6

Элементы теории кодирования

Лекции 13–15

1. Кодирование как способ представления информации

Теория кодирования представляет собой один из разделов дискретной математики, в котором рассматривается процесс представления информации в определенной стандартной форме и обратный процесс восстановления информации по этому представлению.

Рассмотрим следующие примеры.

Пример 1. Представление натуральных чисел в десятичной системе исчисления. При таком представлении каждому числу $n \in N = \{1, 2, \dots\}$ ставится в соответствие последовательность (слово) $a_{r-1}a_{r-2}\dots a_1a_0$, такая, что $n = a_{r-1}10^{r-1} + a_{r-2}10^{r-2} + \dots + a_1 10 + a_0$: $a \neq 0$; $a_i \in \{0, 1, \dots, 9\}$.

Пример 2. Задание (кодирование) геометрических фигур (объектов) уравнениями в системе координат. Так, уравнение $y = kx + b$ кодирует прямую. Уравнение $x^2 + y^2 = R^2$ кодирует окружность. Уравнение $y = ax^2 + bx + c$, $a \neq 0$ кодирует параболу.

Рассмотренные примеры показывают, насколько широко применяется кодирование информации в деятельности человека. Прежде средства кодирования играли вспомогательную роль и не рассматривались как отдельный предмет математического изучения, но с появлением компьютеров ситуация радикально изменилась. Кодирование является центральным вопросом при решении практически всех задач программирования:

— представления данных произвольной природы в памяти компьютера;

— защита информации от несанкционированного доступа;

— обеспечение помехоустойчивости при передаче данных по каналам связи;

— сжатие информации в базах данных.

2. Кодирование и декодирование

Кодированием называется отображение произвольного множества A в множество конечных последовательностей в некотором алфавите B, а *декодированием* — обратное отображение.

Изучение различных свойств кодирования и декодирования, а также построение кодирований (кодов), обладающих требуемыми свойствами, составляет предмет исследований теории кодирования.

Требуемые свойства связаны обычно с простотой реализации кодирования и декодирования, с обеспечением помехоустойчивости и т. д. Под помехоустойчивостью понимается возможность однозначного декодирования при отсутствии или допустимом уровне искажений в кодовых словах.

3. Помехоустойчивое кодирование

Важная роль помехоустойчивого кодирования обусловлена недопустимостью ошибок в вычислительных системах, также использованием кодов, исправляющих или обнаруживающих ошибки на выходе вычислительных устройств. Таким образом, помехоустойчивость или исправление ошибок является функцией декодирования.

4. Канал связи

Передача информации сводится к передаче по каналу связи слов, которые могут искажаться и поэтому восприниматься неверно. Рассмотрим принципиальную схему цифровой системы связи с использованием кодирования и декодирования (рис. 6.1).

Источником сообщений является, как правило, сообщение, состоящее из двоичных или десятеричных цифр, или же текст, записанный с помощью некоторого алфавита. Кодирующее устройство преобразует эти сообщения в сигналы, которые могут быть переданы по каналу связи. Эти сигналы поступают в канал и искажаются шумом либо злоумышленником. Затем искаженный сигнал поступает в декодирующее устройство, в котором исходное сообщение восстанавливается (исправляются ошибки и декодируются) и затем направляется получателю.

Рис. 6.1

В идеальном случае символы, которые появляются на выходе декодирующего устройства, должны совпадать с символами, которые поступают на вход кодирующего устройства. Но в реальных системах передачи и обработки информации появляются случайные и преднамеренные ошибки.

Поэтому назначение кодов состоит в том, чтобы обнаружить и, быть может, исправить такие ошибки.

5. Криптология

Конфиденциальной (секретной) информацией называется информация, которая по соглашению между ее отправителем и получателем не предназначена для третьих лиц.

Исследованием и разработкой математических методов конфиденциальной информации, передаваемой отправителем получателю по общедоступным каналам связи, занимается *криптология*, одной из составляющих которой является *криптография*. Криптография разрабатывает методы защиты информации, т. е. представляет один из классов кодирования с целенаправленным изменением открытого текста, которое затрудняет получение содержащейся в нем конфиденциальной информации.

6. Алфавитное кодирование

В различных задачах, связанных с обработкой, передачей и хранением информации, а также в связи с потребностями техники возникает необходимость преобразования информации в более удобной форме. Кодирование, применяемое для этих целей, называется *алфавитным кодированием*. Алфавитное кодирование — это представление информации в стандартной форме, при которой элементарным синтаксическим единицам языка сообщений (буквам алфавита) последовательно сопоставляются кодовые комбинации символов из некоторого заданного алфавита. Под информацией здесь понимаем линейную последовательность букв (слова). Примером алфавитного кодирования может служить известный код Морзе (азбука Морзе), в котором слова кодируются побуквенно. А буквам сопоставляются слова в алфавите из трех символов $\{\cdot, -, \wedge\}$, где $\wedge$ — пробел.

Другим примером алфавитного кодирования является десятичная система счисления, рассмотренная в первом примере.

В современных компьютерах в основном применяется двоичное кодирование, при котором любому символу исходного алфавита (буква, цифра, символы арифметических операций, специальные знаки и т. д.) ставится в соответствие их двоичный код. Кроме того, алфавитное кодирование применяется в криптологии для маскировки конфиденциальной информации.

7. Математическое изучение алфавитного кодирования

Пусть задан алфавит сообщений $A = \{a_1, a_2, \ldots, a_n\}$, состоящий из конечного числа букв: конечное множество. Конечная последовательность букв из A $\alpha = a_{i_1}a_{i_2}\ldots a_{i_k}$ называется словом в алфавите A, а число k — длиной слова α; если $k = 0$, то слово называют пустым и обозначают $\wedge$.

Множество всех непустых слов конечной длины в алфавите A обозначим через $A*$.

Если $S \subset A*$ (S — подмножество множества $A*$), то слова из S называются сообщениями, а объект, порождающий слова из S, — источником сообщений. Источником сообщений может быть человек, автомат и т. д.

Пусть, кроме алфавита A, задан еще алфавит $B = \{b_1, b_2, \ldots, b_m\}$. Зададим отображение f, которое каждому слову $\alpha \in S$ ставит в соответствие слово $\beta \in B*$, где $B*$ — множество всех непустых слов конечной длины в алфавите B.

Слово β будем называть кодом сообщения α, а процесс перехода от слова α к слову β — кодированием.

Алфавитное кодирование определяется следующим образом. В множестве $B*$ выбираются некоторым образом r слов $\beta_1, \beta_2, ..., \beta_r$, называемых *элементарными кодами.*

По определению получаем, что $f(a_i) = \beta_i$, $i = 1, 2, ..., r$. Тогда код любого слова $\alpha = a_{i_1} a_{i_2} \ldots a_{i_k} \in A*$ есть следующее слово:

$$\beta(\alpha) = f(a_{i_1}) f(a_{i_2}) \ldots f(a_{i_k}) = \beta_{i_1} \beta_{i_2} \ldots \beta_{i_p}.$$

Схема, определяющая отображение f на буквах алфавита A, называется схемой кодирования, обозначается Σ и оформляется в виде таблицы:

a_1	a_2	...	a_{r-1}	a_r
β_1	β_2	...	β_r	β_r

Множество кодовых слов $\{\beta_1, \beta_2, ..., \beta_r\}$ будем обозначать $C(\Sigma)$.

8. Проблема взаимной однозначности

Одним из основных вопросов в кодировании является проблема взаимной однозначности, т. е. возможность по коду β сообщения α однозначно восстановить α.

Возникает вопрос: возможно ли по схеме алфавитного кодирования узнать, обладает ли алфавитное кодирование свойством однозначности или нет? Если решать эту задачу, исходя из определения, то нужно перебрать бесконечное число слов, так как для каждого кода нужно установить, допускает или не допускает этот код однозначное кодирование.

9. Достаточный признак взаимной однозначности алфавитного кодирования

Прежде, чем устанавливать общий критерий взаимной однозначности алфавитного кодирования, приведем простой достаточный признак взаимной однозначности.

Введем следующие понятия.

Если $\beta = \beta'\beta''$, то β' называется началом, или *префиксом*, слова β, а β'' — окончанием, или *постфиксом*, слова β.

Если $\beta' \neq \Lambda$, то β' называется *собственным началом.*

Если $\beta'' \neq \Lambda$, то β'' называется собственным окончанием слова β.

Определение. Схема алфавитного кодирования обладает свойствами префикса, если ни один элементарный код *не является* префиксом другого элементарного кода.

Теорема 1. Если схема Σ алфавитного кодирования обладает свойством префикса, то алфавитное кодирование взаимно однозначно.

Доказательство. Допустим, что некоторое слово $\beta \in B*$ допускает два декодирования. Это значит, что β можно представить в двух видах: $\beta_1 = \beta_{i_1}\beta_{i_2}\ldots\beta_{i_k}$; $\beta_2 = \beta_{j_1}\beta_{j_2}\ldots\beta_{j_l}$.

Так как эти представления различны, то существует такое p, что $1 \leqslant p \leqslant \min(k, l)$, для которого $\beta_{i_p} \neq \beta_{j_p}$. Но тогда одно из слов β_1 и β_2 есть префикс другого. Это противоречит условию теоремы. Следовательно, наше утверждение о существовании двух декодирований неверно. Теорема доказана.

Заметим, что условие *префиксности* не является необходимым, т. е. другими словами, схема алфавитного кодирования может не обладать свойствами префиксности, но алфавитное кодирование, задаваемое этой схемой, будет взаимно однозначным.

Определение. Слово $b_{i_n}b_{i_{n-1}}\ldots b_{i_1}$ будем называть обратным к слову $\beta = b_{i_1}b_{i_2}\ldots b_{i_n}$ и обозначать β^{-1}.

Определение. Схему кодирования, полученную из схемы кодирования $\Sigma\{\beta_1, \ldots, \beta_r\}$ заменой каждого элементарного кода β_i на β_i^{-1}, будем называть обратной к схеме Σ и обозначать Σ^{-1}. Введенное понятие позволяет усилить теорему 1.

Теорема 2. Если схема Σ или схема Σ^{-1} обладает свойством префикса, то тогда алфавитное кодирование, определяемое схемой $\Sigma(\Sigma^{-1})$, будет взаимно однозначным.

Доказательство. Очевидно, что алфавитные кодирования, задаваемые схемами Σ и Σ^{-1}, одновременно или обладают, или не обладают свойством взаимной однозначности. Отсюда немедленно следует справедливость теоремы 2.

Заметим, что условие теоремы 2 не является необходимым, так как существуют схемы кодирования Σ, со свойством взаимной однозначности, но такие, что ни Σ, ни Σ^{-1} не обладают свойством префиксности.

10. Общий критерий взаимной однозначности

Для вывода общего критерия взаимной однозначности понадобится следующее понятие.

Представление элементарного кода β_i схемы алфавитного кодирования Σ в виде

$$\beta_i = \beta'\beta_{i_1}\ldots\beta_{i_k}\beta'',$$

где слово β' не может оканчиваться на элементарный код, а слово β'' начинаться с элементарного кода, будем называть *нетривиальным разложением* элементарного кода. При этом одно из слов β' или β'' может быть пустым ($\wedge$).

Для определения однозначности или неоднозначности схемы алфавитного кодирования существует эффективный алгоритм, использующий понятие нетривиального разложения элементарных кодов.

Опишем этот алгоритм.

1. Для каждого элементарного кода выписываем все нетривиальные разложения.

2. Выписываем множество M_1, состоящее из слов β', которые входят в качестве начал в нетривиальные разложения элементарных кодов.

3. Выписываем множество M_2, состоящее из всех слов β'', которые являются окончанием нетривиальных разложений элементарных кодов.

4. Составляем множество $M = M_1 \cap M_2 \cup \{\wedge\}$, т. е. множество слов, встречающихся как в качестве начала, так и в качестве окончания в нетривиальных разложениях элементарных кодов.

5. Выписываем все разложения элементарных кодов, связанных с множеством M, т. е. все разложения элементарных кодов вида:

$$\beta_i = \beta'\beta_{i_1}\cdots\beta_{i_k}\beta'',$$

где $\beta', \beta'' \in M$, а k может быть равно 0.

6. По разложениям, полученным в пункте 5, строится ориентированный граф G_Σ следующим образом. Вершины графа отождествляют с элементами множества M. Пара вершин β' и β'' соединяются ориентированными ребрами в том и только в том случае, если существует разложение $\beta'\beta_{i_1}\ldots\beta_{i_k}\beta''$. При этом ребру (β', β'') приписывается слово $\beta_{i_1}\ldots\beta_{i_k}$, соответствующее этому разложению.

7. По полученному графу G_Σ легко проверить, обладает или нет исходная схема кодирования свойством взаимной однозначности.

Справедлива следующая теорема.

Теорема 3 (А. А. Маркова). Алфавитное кодирование со схемой Σ *не обладает* свойством взаимной однозначности тогда и только тогда, когда граф G_Σ содержит ориентированный цикл (контур), проходящий через вершину $\wedge$. Т. е. алфавитное кодирование является взаимно однозначным тогда и только тогда, когда в графе G_Σ отсутствуют контуры и петли, проходящие через вершины $\wedge$.

Замечание. Напомним, что если схема алфавитного кодирования Σ не обладает свойством взаимной однозначности, то это означает, что существуют слова из $B*$, допускающие два кодирования. Одно из таких слов β легко находится по графу G_Σ.

Для записи слова β нужно посмотреть ориентированный цикл, проходящий через вершину $\wedge$, начиная с $\wedge$, и выписать последовательно все слова, приписанные ребрам и вершинам, входящим в этот цикл.

Восьмое практическое занятие по теме «Алфавитное кодирование»

Задача 1. Задано алфавитное кодирование, для которого $A = \{a_1, a_2\}$; $B = \{b_1, b_2\}$ и схема Σ задана таблицей

a_1	a_2
b_1	b_1b_2

Выясните, обладает ли эта схема кодирования свойством однозначности.

Решение. Процесс декодирования осуществляется следующим образом: код β слова α разбивается на элементарные коды. Для этого в слове β последовательно находятся все буквы b_2 и затем выделяются пары b_1b_2, каждая такая пара соответствует букве a_2. Оставшиеся буквы b_1 соответствуют букве a_1.

Так код $\beta = b_1b_1b_1b_2b_1b_2b_1b_2b_1b_1b_2$ разбивается однозначно на элементарные коды: $\beta = (b_1)(b_1)(b_1b_2)(b_1b_2)(b_1b_2)(b_1)(b_1b_2)$.

Следовательно, исходное сообщение есть слово $\alpha = a_1a_1a_2a_2a_2a_1a_2$.

Задача 2. Пусть схема Σ задана следующей таблицей

a_1	a_2	a_3
b_1b_1	$b_2b_1b_1$	$b_1b_1b_2$

Покажите, что эта схема не обладает свойством однозначности.

Решение. Рассмотрим слово $\beta = b_1b_1b_2b_1b_1b_2b_1b_1$.

Это слово допускает два декодирования.

$$\beta_1 = (b_1b_1)(b_2b_1b_1)(b_2b_1b_1), \quad \beta_2 = (b_1b_1b_2)(b_1b_1b_2)(b_1b_1),$$

тогда $\alpha_1 = a_1a_2a_2$; $\alpha_2 = a_3a_3a_1$.

Следовательно, схема Σ не обладает свойством однозначности.

Задача 3. Выясните, обладает ли код $C(\Sigma)$ свойством префикса:

1) $C(\Sigma) = \{a, ba, bb, bbba\}$,
2) $C(\Sigma) = \{ab, bb, ba, aab\}$,
3) $C(\Sigma) = \{ac, c, bb, abc, bac, abb, abcb\}$,
4) $C(\Sigma) = \{a, ba, cfb, acb\}$,
5) $C(\Sigma) = \{a, ba, bba, \ldots, (b^n a, \ldots)\}$,
6) $C(\Sigma) = \{a, ba, \ldots, b(a)^n, \ldots\}$.

Ответ: (1); (3); (4); (6) — $C(\Sigma)$ не обладает свойством префикса. (2); (5) — $C(\Sigma)$ обладает свойством префикса.

Задача 4. Схема алфавитного кодирования Σ задана таблицей

a_1	a_2	a_3
b_1	b_1b_2	b_3b_1

Покажите, что ни Σ, ни Σ^{-1} не обладают свойством префикса, но тем ни менее алфавитное кодирование, задаваемое этой схемой, является взаимно-однозначным.

Решение. Схема Σ не обладает свойством префикса, так как b_1 является префиксом слова b_1b_2. С другой стороны, в схеме $C(\Sigma^{-1}) = \{b_1, b_2, b_1, b_1, b_3\}$ b_1 является префиксом слова b_1b_3.

По коду β любого слова $\alpha \in A*$, $A = \{a_1, a_2, a_3\}$, однозначно восстанавливается это слово. Алгоритм восстановления следующий:

1. В кодовом слове β находим все буквы b_2 и выделяем все пары b_1b_2.
2. Находим в кодовом слове β все буквы b_2 и выделяем все пары b_3b_1.
3. Каждую пару b_1b_2 заменяем буквой a_2, пару b_3b_1 заменяем буквой a_3, оставшиеся буквы b_1 заменяем на a_1. В результате получим исходное сообщение (слово) α.

Действительно пусть

$$\beta = b_1b_1b_2b_3b_1b_1b_1b_1b_2b_1b_3b_1,$$
$$\beta = (b_1)(b_1b_2)(b_3b_1)(b_1)(b_1)(b_1b_2)(b_1)(b_3b_1),$$
$$\alpha = a_1a_2a_3a_1a_1a_2a_1a_3.$$

Задача 5. Выясните, является ли код $C(\Sigma)$ с кодирующим алфавитом $\{0, 1, 2\}$ однозначно декодируемым:

1) $\{01, 201, 112, 122, 0112\}$;
2) $\{001, 021, 102, 201, 001121, 01012101\}$;
3) $\{0, 01, 0010001001\}$;
4) $\{20, 01202, 22, 2001, 2012010, 10201121, 1112\}$;
5) $\{01, 011, 100, 2100, 101210, 001210\}$;
6) $\{01, 011, 100, 2100, 10110, 00112\}$;
7) $\{01, 12, 021, 0102, 10112\}$;
8) $\{01, 12, 012, 111, 0102, 10112, 01112\}$;
9) $\{01, 12, 012, 0102, 020112\}$;
10) $\{01, 10, 210, 121, 0210, 0112\}$;

11) $\{01, 10, 210, 201, 0210, 011022, 2221\}$;
12) $\{01, 10, 210, 201, 0210, 011022, 221\}$;
13) $\{01, 10, 210, 201, 0210, 011022\}$;
14) $\{01, 12, 011, 01210, 20120, 2011220\}$;
15) $\{01, 12, 011, 01210, 201120, 2011220\}$;
16) $\{000, 0100, 10, 1001, 0010010\}$;
17) $\{01, 12, 01121, 21201\}$.

Задача 6. Алфавитное кодирование задано схемой Σ в виде таблицы

a_1	a_2	a_3	a_4	a_5
b_1b_2	$b_1b_3b_2$	b_2b_3	$b_1b_2b_1b_3$	$b_2b_1b_2b_2b_3$

С помощью общего критерия взаимной однозначности выясните, обладает ли эта схема кодирования свойством взаимной однозначности или нет.

Решение. Алгоритм решения.

1. Выпишем все нетривиальные разложения каждого элементарного кода:

$$\beta_1 = (b_1)(b_2),$$
$$\beta_2 = (b_1)(b_3b_2) = (b_1b_3)(b_2),$$
$$\beta_3 = (b_2)(b_3),$$
$$\beta_4 = (b_1)(b_2b_1b_3) = (b_1b_2)(b_1b_3) = (b_1b_2b_1)(b_3),$$
$$\beta_5 = (b_2)(b_1b_2)(b_2b_3) = (b_2b_1)(b_2b_2b_3) = (b_2b_1b_2b_2)(b_3).$$

2. Выписываем множество M_1, состоящее из слов, входящих в качестве начал в нетривиальные разложения элементарных кодов.

$$M_1 = \{b_1; b_1b_3; b_2; b_1b_2; b_1b_2b_1; b_2b_1; b_2b_1b_2b_2\}.$$

3. Выписываем множество M_2, состоящее из слов, которые являются окончанием нетривиальных разложений элементарных кодов:

$$M_2 = \{b_2; b_3b_2; b_3; b_2b_1b_3; b_1b_3; b_2b_3; b_2b_2b_3\}.$$

4. Составляем множество $M = M_1 \cap M_2 \cup \{\wedge\}$.

$$M = \{\wedge, b_2, b_1b_3\}.$$

5. Выписываем все разложения элементарных кодов, связанных с элементами множества M.

$$\beta_2 = (b_1b_3) \wedge (b_2),$$
$$\beta_4 = (b_1b_2)(b_1b_3) = \wedge\beta_1(b_1b_3),$$
$$\beta_5 = (b_2)(b_1b_2)(b_2b_3) = b_2\beta_1\beta_3 \wedge .$$

6. По полученным разложениям строим ориентированный граф: вершины графа — элементы множества $M = \{\wedge, b_2, b_1b_3\}$. Пара вершин соединяется ориентированными ребрами в том и только в том случае, если существует разложение, где вершины графа входят в разложение в качестве начала и конца (рис. 6.2).

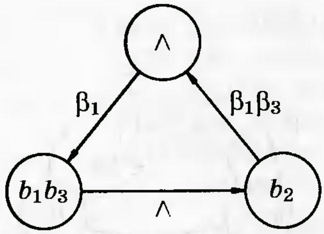


Рис. 6.2

7. В графе G_Σ присутствует ориентированный цикл, проходящий через вершину Λ, следовательно, согласно теореме Маркова, алфавитное кодирование с заданной схемой не является взаимно-однозначным.

8. Выписывая слова, приписанные вершинам и дугам ориентированного цикла графа G_Σ, получаем слово декодируемое неоднозначно:

$$\beta = b_1b_2b_1b_3b_2b_1b_2b_2b_3,$$
$$\beta_1 = (b_1b_2)(b_1b_3b_2)(b_1b_2)(b_2b_3),$$
$$\alpha_1 = a_1a_2a_1a_3,$$
$$\beta_2 = (b_1b_2b_1b_3)(b_2b_1b_2b_2b_3),$$
$$\alpha_2 = a_4a_5.$$

Задача 7. Схема алфавитного кодирования задана таблицей

a_1	a_2	a_3	a_4	a_5
b_1	b_2b_1	$b_1b_2b_2$	$b_2b_1b_2b_2$	$b_2b_2b_2b_2$

Выясните, обладает ли эта схема свойством взаимной однозначности.

Решение.

1. Выписываем все нетривиальные разложения элементарных кодов.

$$\beta_2 = (b_2)(b_1),$$
$$\beta_3 = (b_1)(b_2b_2) = (b_1b_2)(b_2),$$
$$\beta_4 = (b_2)(b_1b_2b_2) = (b_2b_1b_2)(b_2) = (b_2)(b_1)(b_2b_2) = (b_2b_1)(b_2b_2),$$
$$\beta_5 = (b_2)(b_2b_2b_2) = (b_2b_2)(b_2b_2) = (b_1b_2b_2)(b_2).$$

2. $M_1 = \{b_2; b_1; b_1b_2; b_2b_1b_2; b_2b_1; b_2b_2; b_2b_2b_2\}$.
3. $M_2 = \{b_1; b_2b_2; b_2; b_1b_2b_2; b_2b_2b_2\}$.
4. $M = \{b_2; b_2b_2; b_2b_2b_2; \Lambda\}$.

5.
$$\beta_2 = (b_2)\beta_1\Lambda,$$
$$\beta_3 = \Lambda\beta_1(b_2b_2),$$
$$\beta_4 = (b_2)\beta_3\Lambda = (b_2)\beta_1(b_2b_2) = \Lambda\beta_2(b_2b_2),$$
$$\beta_5 = (b_2)\wedge(b_2b_2b_2) = (b_2b_2)\wedge(b_2b_2) = (b_1b_2b_2)\wedge(b_2).$$

6. Строим граф G_Σ (рис. 6.3).

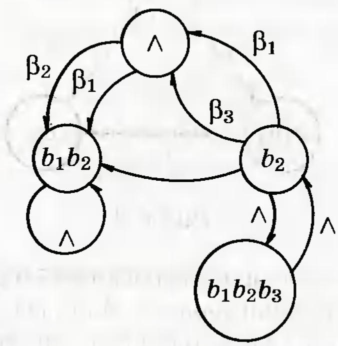


Рис. 6.3

7. В графе G_Σ отсутствуют контуры и петли, проходящие через вершину Λ, следовательно, алфавитное кодирование с заданной схемой Σ однозначно декодируемо.

Задача 8. Выясните, является ли кодирование $C(\Sigma) = \{a, ab, cab, baac\}$ взаимнооднозначным.

Решение.

1.
$$\beta_2 = (a)(b),$$
$$\beta_3 = (c)(ab) = (ca)(b),$$
$$\beta_4 = (b)(aac) = (ba)(ac) = (baa)(c) = (b)(a)(a)(c).$$

2. $M_1 = \{a; c; ca; b; ba; baa\}$.
3. $M_2 = \{b; ab; aac; ac; c\}$.
4. $M = \{b; c; \Lambda\}$.

5.
$$\beta_2 = \Lambda\beta_1(b),$$
$$\beta_3 = c\beta_2\Lambda,$$
$$\beta_4 = (b)\beta_1\beta_1(c).$$

6. Строим граф C_Σ (рис. 6.4).

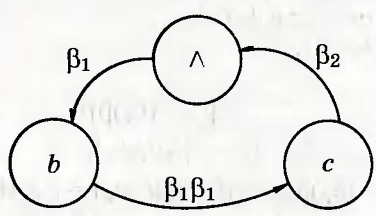


Рис. 6.4

7. Кодирование не является взаимно-однозначным, так как существует контур, проходящий через вершину Λ.

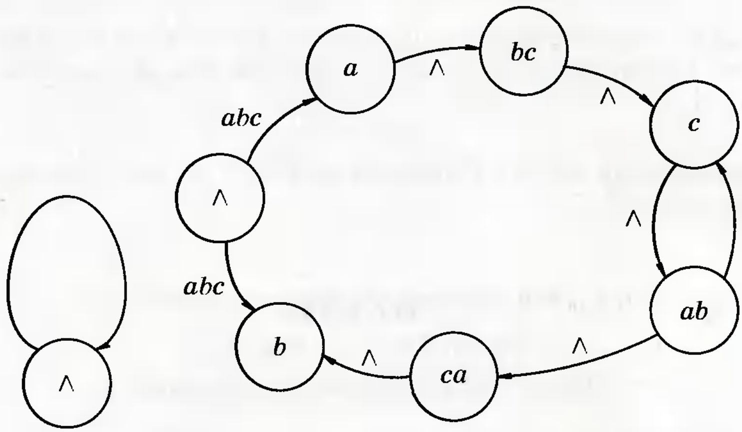
8. Получаем слово, декодируемое неоднозначно:

$$\beta = abaacab,$$
$$\beta_1 = (a)(baac)(ab),$$
$$\beta_2 = (ab)(a)(a)(cab).$$

Задача 9. Выясните, является ли алфавитное кодирование со схемой Σ взаимно-однозначным, если:

а) $C(\Sigma) = \{a, b, aab\}$,

б) $C(\Sigma) = \{cab, abc, dcc, abca, abcb\}$.



а б

Рис. 6.5

Решение.

а) граф G_Σ содержит петлю $(\Lambda\Lambda)$ (рис. 6.5*а*). Код не является однозначно кодируемым. Слово декодируемое неоднозначно:

$$\beta = (aab) = (a)(a)(b).$$

б) граф G_Σ контуров и петель, проходящих через Λ, не содержит (рис. 6.5*б*). Кодирование является взаимно-однозначным.

Задача 10. Покажите, что алфавитное кодирование с кодирующим алфавитом $\{0, 1, 2\}$ и множеством кодирующих слов $C(\Sigma) = \{01, 201, 112, 122, 0112\}$ не является взаимно-однозначным.

Решение. Построим граф G_Σ (рис. 6.6). Он содержит контур, проходящий через вершину Λ, следовательно, кодирование не является взаимно-однозначным, причем декодируемое слово неоднозначно $\beta = 0112201$.

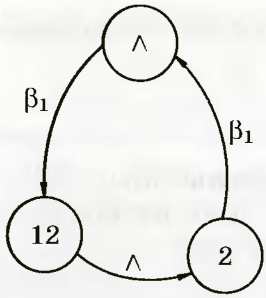


Рис. 6.6

Задача 11. (Для самостоятельного решения.)

Покажите, что алфавитное кодирование с кодирующим алфавитом $\{0, 1, 2\}$ и множеством кодирующих слов:

а) $C(\Sigma) = \{001, 021, 102, 201, 001121, 01012101\}$ является однозначно кодируемым;

б) $C(\Sigma) = \{0, 01, 0010001001\}$ не является однозначно декодируемым.

11. Двоичный алфавит

В современных компьютерах любая информация представляется в виде двоичных кодов, т. е. упорядоченных наборов двух различных символов, которые обычно обозначаются через 0 и 1. Таким образом, алфавит, в котором записываются сообщения, считаем состоящим из двух символов $\{0, 1\}$. Он называется двоичным алфавитом.

Тогда сообщение есть конечная последовательность символов этого алфавита.

Это простейшее алфавитное кодирование, которое является взаимно-однозначным, так как каждому символу исходного алфавита ставится в соответствие определенное двоичное число.

Например, натуральному числу 23 соответствует двоичное число 10111, поскольку

$$23 = 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0.$$

Полезно запомнить запись в двоичной системе первых шестнадцати натуральных чисел

0	1	2	3	4	5	6	7
0	1	10	11	100	101	110	111

8	9	10	11	12	13	14	15
1000	1001	1010	1011	1100	1101	1110	1111

12. Самокорректирующиеся коды

Пусть мы имеем множество всех двоичных слов длины m. Эти слова передаются по каналу связи, в котором действует источник помех. Этот источник помех при передаче двоичного слова длины m может выдавать ошибки не более чем в p символах.

Это означает, что двоичная последовательность, полученная на выходе канала, отличается от исходной не более чем в p позициях.

Очевидно, что если исходное слово передавать без предварительного кодирования, то установить на выходе истинное сообщение практически невозможно. Поэтому возникает задача построения по исходному, любому слову $a_1a_2...a_m$ его самокорректирующегося кода $b_1b_2...b_l$ ($l > m$), позволяющего по полученному на выходе канала кода $b'_1b'_2...b'_l$ однозначно восстановить передаваемый код $b_1b_2...b_l$, а значит, и исходное сообщение $a_1a_2...a_m$. Напомним, что при передаче кода $b_1b_2...b_l$ по каналу связи код, возможно, исказился и, следовательно, на выходе канала будет слово $b'_1b'_2...b'_l$, вообще говоря, отличающееся от $b_1b_2...b_l$ не более чем в p позициях.

Коды, обладающие выше указанными свойствами, называют *самокорректирующимися кодами* относительно источника помех или кодами, исправляющими p ошибок.

13. Коды Хемминга

Перейдем к построению самокорректирующихся кодов для случая $p = 1$, которые называются *кодами Хемминга*. Будем считать, что в канале связи при передаче сообщения может произойти не более одной ошибки. Это означает, что если исходное сообщение $a_1a_2...a_m$ кодируется набором $b_1b_2...b_l$ ($l = m + k$), то на выходе возможны следующие варианты кодов: $b_1b_2...b_l$; $\overline{b_1}b_2...b_l$; $b_1\overline{b_2}...b_l$; ...; $b_1b_2...\overline{b_l}$.

Таким образом, число вариантов равно $l+1$. Поясним, что ошибка может не произойти, либо она произойдет в одном из l разрядов, и символ b_i заменится на противоположный $\overline{b_i}$. Число дополнительных разрядов для построения кодов Хемминга нужно выбрать так, чтобы их хватило для кодирования перечисленных $l + 1$ случаев. Следовательно, необходимо, чтобы

$$2^k \geqslant l+1 \quad \text{или} \quad 2^m \leqslant \frac{2^l}{l+1}.$$

Поэтому, зная m, l выбираем как наименьшее целое число, удовлетворяющее условию:

$$2^m \leqslant \frac{2^l}{l+1}.$$

Число l называется длиной кода Хемминга. Число m — число информационных символов.

Замечание. Учитывая, что $l = m + k$, можно выбирать не l, а число k, которое называется числом контрольных символов и является наименьшим целым числом, удовлетворяющим условию: $2^k \geqslant k + m + 1$.

Например, если $m = 4$, то $l = 7$, $k = 3$;

если $m = 9$, то $l = 13$, $k = 4$.

Таким образом, при построении кодов Хемминга, первое, что нужно сделать: это по числу m определить числа k и l.

14. Алгоритм построения кода Хемминга

Пусть для сообщения $\alpha = a_1a_2...a_m$ длины m необходимо построить код Хемминга. Возьмем $m = 9$; исходное сообщение

$$\alpha = 101110111 = a_1a_2a_3a_4a_5a_6a_7a_8a_9.$$

Тогда $l = 13$, $k = 4$; код Хемминга $\beta = b_1b_2b_3b_4b_5b_6b_7b_8b_9b_{10}b_{11}b_{12}b_{13}$.

(1) Представим каждое число i из множества $L = \{1, 2, ..., l\}$ в виде k-разрядного двоичного числа $\tau = V_{k-1}V_{k-2}...V_1V_0$. Результаты запишем в виде таблицы

$\tau \backslash i$	1	2	3	4	5	6	7	8	9	10	11	12	13
V_0	1	0	1	0	1	0	1	0	1	0	1	0	1
V_1	0	1	1	0	0	1	1	0	0	1	1	0	0
V_2	0	0	0	1	1	1	1	0	0	0	0	1	1
V_3	0	0	0	0	0	0	0	1	1	1	1	1	1

(2) Разберем множество L на k подмножеств следующим образом:

$$L_0 = \{i \in L_0 : V_0 = 1\}; \quad L_0 = \{1, 3, 5, 7, 9, 11, 13\},$$
$$L_1 = \{i \in L_1 : V_1 = 1\}; \quad L_1 = \{2, 3, 6, 7, 10, 11\},$$
$$L_2 = \{i \in L_2 : V_2 = 1\}; \quad L_2 = \{4, 5, 6, 7, 12, 13\},$$
$$L_3 = \{i \in L_3 : V_3 = 1\}; \quad L_3 = \{8, 9, 10, 11, 12, 13\}.$$

(3) Первые элементы (их ровно k) этих множеств есть степени числа 2; они определяют номера контрольных разрядов кода Хемминга. Остальные элементы множества L определяют номера информационных разрядов. Следовательно, в коде Хемминга разряды $b_1b_2b_4b_8$ — контрольные, остальные разряды $b_3b_5b_6b_7b_9b_{10}b_{11}b_{12}b_{13}$ — информационные.

(4) Формирование значений информационных символов.

Информационные символы кода Хемминга формируются естественным образом из символов исходного сообщения $a_1a_2...a_m$. А именно: $b_3 = a_1$; $b_5 = a_2$; $b_6 = a_3$; $b_7 = a_4$; $b_9 = a_5$; $b_{10} = a_6$; $b_{11} = a_7$; $b_{12} = a_8$; $b_{13} = a_9$.

Так как исходное сообщение $\alpha = 101110111$, то $b_3 = 1$; $b_5 = 0$; $b_6 = 1$; $b_7 = 1$; $b_9 = 1$; $b_{10} = 0$; $b_{11} = b_{12} = b_{13} = 1$.

(5) Формирование значений контрольных символов.

После определения информационных символов контрольные символы определяются следующим образом:

$$b_1 = \oplus\sum b_j; \quad j \in L_0; \quad j \neq 1,$$
$$b_2 = \oplus\sum b_j; \quad j \in L_1; \quad j \neq 2,$$
$$b_4 = \oplus\sum b_j; \quad j \in L_2; \quad j \neq 4,$$
$$b_8 = \oplus\sum b_j; \quad j \in L_3; \quad j \neq 8.$$

Здесь $\oplus\sum$ — сумма по модулю два, b_j — разряды, имеющие номера из соответствующих множеств L_i.

В примере будем иметь:

$$b_1 = b_3 \oplus b_5 \oplus b_7 \oplus b_9 \oplus b_{11} \oplus b_{13} = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 1,$$
$$b_2 = b_3 \oplus b_6 \oplus b_7 \oplus b_{10} \oplus b_{11} = 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 0,$$
$$b_4 = b_5 \oplus b_6 \oplus b_7 \oplus b_{12} \oplus b_{13} = 0 \oplus 1 \oplus 1 \oplus 1 = 0,$$
$$b_8 = b_9 \oplus b_{10} \oplus b_{11} \oplus b_{12} \oplus b_{13} = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 = 0.$$

(6) Окончательно, для сообщения $\alpha = 101110111$ код Хемминга β будет следующим: $\beta = b_1b_2b_3b_4b_5b_6b_7b_8b_9b_{10}b_{11}b_{12}b_{13} = 1010011010111$.

Таким образом, можно построить код Хемминга для сообщения с любым набором длины m.

15. Обнаружение ошибки в кодах Хемминга

Пусть при передаче кода $\beta = b_1b_2...b_l$ произошла ошибка в разряде с номером t, т. е. на выходе канала получено слово $\beta' = b_1b_2...b_{t-1}b_tb_{t+1}...b_l$.

Представим t в виде k-разрядного двоичного числа: $t = V_{k-1}...V_1V_0$. Покажем, как по коду β' найти разряды V_i числа t.

Рассмотрим $t' = V'_{k-1}...V'_1V'_0$, где:

$$V'_0 = \oplus \sum b'_j; \quad j \in L_0,$$
$$V'_1 = \oplus \sum b'_j; \quad j \in L_1,$$
$$\dots$$
$$V'_{k-1} = \oplus \sum b'_j; \quad j \in L_{k-1}.$$

Покажем, что $t' = t$, т. е. $V'_0 = V_0$; $V'_1 = V_1$; ...; $V'_{k-1} = V_{k-1}$.

Рассмотрим ситуации:

1. Пусть $V_i = 0$; это значит, что $t \notin L_i = \{j \in L_i : V_i = 1\}$.

Следовательно, все разряды с номерами из L_i получены на выходе канала без искажения, т. е. $b'_t = b_t$, $t \in L_i$. Так как $b_{2^i} = \oplus \sum b_j$, $j \in L_i$, $j \neq 2^i$ а это равносильно равенству $\oplus \sum b_j = \oplus \sum b'_j = V'_i = 0 = V_i$ $(j \in L_i)$.

2. Пусть $V_i = 1$, тогда $t \in L_i = \{j \in L_i : V_i = 1\}$, и некоторый разряд с номером из L_i получен на выходе канала с искажением, т. е. для некоторого q из L_i $b'_q = \overline{b_q} = b_q \oplus 1$, а для всех $j \in L_i$, $j \neq q$, $b'_j = b_j$.

Отсюда получаем $V'_i = \oplus \sum b'_j = (\oplus \sum b_j) \oplus 1 = 0 \oplus 1 = 1$. Следовательно, и в этом случае $V_i = V'_i$.

Пусть в рассмотренной выше задаче ошибка при передаче кодового слова $\beta = b_1b_2b_3b_4b_5b_6b_7b_8b_9b_{10}b_{11}b_{12}b_{13} = 1010011010111$ произошла в 11 разряде ($t = 11$). Т. е. на выходе канала получено сообщение

$$\beta' = b'_1b'_2b'_3b'_4b'_5b'_6b'_7b'_8b'_9b'_{10}b'_{11}b'_{12}b'_{13} = 1010011010011.$$

Для этого кодового сообщения получаем:

$$V_0 = b'_1 \oplus b'_3 \oplus b'_5 \oplus b'_7 \oplus b'_9 \oplus b'_{11} \oplus b'_{13} = 1 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 1,$$
$$V_1 = b'_2 \oplus b'_3 \oplus b'_6 \oplus b'_7 \oplus b'_{10} \oplus b'_{11} = 0 \oplus 1 \oplus 1 \oplus 1 \oplus 0 \oplus 0 = 1,$$
$$V_2 = b'_4 \oplus b'_5 \oplus b'_6 \oplus b'_7 \oplus b'_{12} \oplus b'_{13} = 0 \oplus 0 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 0,$$
$$V_3 = b'_8 \oplus b'_9 \oplus b'_{10} \oplus b'_{11} \oplus b'_{12} \oplus b'_{13} = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 \oplus 1 = 1.$$

Таким образом, двоичное представление номера разряда, в котором произошла ошибка, есть 1011. Но это не что иное, как двоичное представление числа 11. Следовательно, ошибочный разряд 11.

16. Декодирование (получение исходного сообщения)

Этот шаг осуществляется так: после исправления ошибки выписать последовательно слева направо из кода сообщения информационные символы, т. е. $a_1a_2...a_m = b_3b_5b_6b_7b_9b_{10}b_{11}b_{12}b_{13}$. В нашем примере из кода $\beta = 1010011010111$ выписываем $\alpha = 101110111$. Это и есть исходное сообщение.

Девятое практическое занятие по теме «Коды Хемминга»

Задача 1. По методу Хемминга постройте кодовое слово для сообщения $\alpha = 1011$.

Решение. $m = 4$; $2^4 \leqslant \frac{2^l}{l+1}$; $l = 7$, $k = 3$, $L = \{1, 2, 3, 4, 5, 6, 7\}$.

1.

$\tau \backslash i$	1	2	3	4	5	6	7
V_0	1	0	1	0	1	0	1
V_1	0	1	1	0	0	1	1
V_2	0	0	0	1	1	1	1

2.
$$L_0 = \{i \in L : V_0 = 1\} = \{1, 3, 5, 7\},$$
$$L_1 = \{i \in L : V_1 = 1\} = \{2, 3, 6, 7\},$$
$$L_2 = \{i \in L : V_2 = 1\} = \{4, 5, 6, 7\}.$$

3. b_1, b_2, b_4 — контрольные разряды.

4. b_3, b_5, b_6, b_7 — информационные разряды, причем
$$b_3 = a_1, \quad b_5 = a_2, \quad b_6 = a_3, \quad b_7 = a_4,$$
$$b_3 = 1, \quad b_5 = 0, \quad b_6 = 1, \quad b_7 = 1.$$

5.
$$b_1 = b_3 \oplus b_5 \oplus b_7 = 1 \oplus 0 \oplus 1 = 0,$$
$$b_2 = b_3 \oplus b_6 \oplus b_7 = 1 \oplus 1 \oplus 1 = 1,$$
$$b_4 = b_5 \oplus b_6 \oplus b_7 = 0 \oplus 1 \oplus 1 = 0.$$

6. $\beta = b_1b_2b_3b_4b_5b_6b_7 = 0110011$.

Таким образом, кодовым словом для $\alpha = 1011$ является слово $\beta = 0110011$.

Задача 2. Постройте по методу Хемминга кодовое слово для сообщений:

1) $\alpha = 010$; 2) $\alpha = 011$; 3) $\alpha = 1001$; 4) $\alpha = 1101$; 5) $\alpha = 10101011$; 6) $\alpha = 111001111$; 7) $\alpha = 100010011$; 8) $\alpha = 01110111011$.

Ответ: 1) 100110; 2) 110011; 3) 00110014; 4) 1010101; 5) 1110010110114; 6) 0010110001111; 7) 1110000110011; 8) 000011110111011.

Задача 3. Декодируйте слово $\beta' = 1001110$, где произошла ошибка не более чем в одном разряде.

Решение. Имеем $l = 7$; $2^m \leqslant \dfrac{2^l}{l+1} = \dfrac{2^7}{8} = 2^4$; $m = 4$; $k = l - m = 7 - 4 = 3$; $\beta' = b'_1b'_2b'_3b'_4b'_5b'_6b'_7 = 1001110$.

Вычислим $V = \{V_0, V_1, V_2\}$. Имеем

$$L_0 = \{1,3,5,7\}; \quad V_0 = b'_1 \oplus b'_3 \oplus b'_5 \oplus b'_7 = 1 \oplus 0 \oplus 1 \oplus 0 = 0,$$
$$L_1 = \{2,3,6,7\}; \quad V_1 = b'_2 \oplus b'_3 \oplus b'_6 \oplus b'_7 = 0 \oplus 0 \oplus 1 \oplus 0 = 1,$$
$$L_2 = \{4,5,6,7\}; \quad V_2 = b'_4 \oplus b'_5 \oplus b'_6 \oplus b'_7 = 1 \oplus 1 \oplus 1 \oplus 0 = 1.$$

Получили 110 — это двоичное представление числа 6. Следовательно, ошибка произошла в шестом разряде, значит, $\beta = 1001100$. Вычеркивая контрольные разряды 1, 2, 4, получим исходное сообщение $\alpha = 0100$.

Задача 4. Декодируйте слово $\beta' = 001011110111111$.
Решение. $\beta' = b'_1b'_2b'_3b'_4b'_5b'_6b'_7b'_8b'_9b'_{10}b'_{11}b'_{12}b'_{13}b'_{14}b'_{15} = 001011110111111$,

$$l = 15; \quad 2^m \leqslant \frac{2^{15}}{16} \leqslant \frac{2^{15}}{2^4} = 2^{11} \Rightarrow m = 11, \quad k = l - m = 15 - 11 = 4.$$

$V_i \backslash i$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
V_0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
V_1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
V_2	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
V_3	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

$$L_0 = \{1,3,5,7,9,11,13,15\};$$
$$L_1 = \{2,3,6,7,10,11,14,15\};$$
$$L_2 = \{4,5,6,7,12,13,14,15\};$$
$$L_3 = \{8,9,10,11,12,13,14,15\};$$
$$V_0 = b'_1 \oplus b'_3 \oplus b'_5 \oplus b'_7 \oplus b'_9 \oplus b'_{11} \oplus b'_{13} \oplus b'_{15} = 0 \oplus 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 = 0;$$
$$V_1 = b'_2 \oplus b'_3 \oplus b'_6 \oplus b'_7 \oplus b'_{10} \oplus b'_{11} \oplus b'_{14} \oplus b'_{15} = 0 \oplus 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 1;$$
$$V_2 = b'_4 \oplus b'_5 \oplus b'_6 \oplus b'_7 \oplus b'_{12} \oplus b'_{13} \oplus b'_{14} \oplus b'_{15} = 0 \oplus 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 1;$$
$$V_3 = b'_8 \oplus b'_9 \oplus b'_{10} \oplus b'_{11} \oplus b'_{12} \oplus b'_{13} \oplus b'_{14} \oplus b'_{15} =$$
$$= 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 1.$$

Получили 1110. Это двоичное представление числа 14: ошибка в 14-м разряде $\beta' = b'_1b'_2b'_3b'_4b'_5b'_6b'_7b'_8b'_9b'_{10}b'_{11}b'_{12}b'_{13}b'_{14}b'_{15} = 001011110111101$.

Вычеркивая разряды 1, 2, 4, 8, получим $\alpha = 11110111101$.

Задача 5. По каналу связи передавалось кодовое слово α, построенное по методу Хемминга. Канал связи искажал слово не более чем в одном разряде, в результате было получено слово β'.

Восстановите исходное сообщение.

1) $\beta' = 110$; 2) $\beta' = 101110$; 3) $\beta' = 011110$; 4) $\beta' = 1001011$; 5) $\beta' = 0101101$; 6) $\beta' = 1011101$; 7) $\beta' = 1100011$; 8) $\beta' = 11011100110$; 9) $\beta' = 1010101010100$.

Ответ: 1) 1; 2) 110; 3) 110; 4) 0011; 5) 0101; 6) 1101; 7) 0011; 8) 0111110; 9) 110110100.

Контрольные вопросы

1. В чем состоит процесс кодирования?
2. Что такое кодирование и декодирование?
3. Что такое канал связи?
4. Чем занимается криптология? Что такое криптография?
5. Что такое алфавитное кодирование?
6. Дайте объяснение математического понятия алфавитного кодирования.
7. Как определяется схема алфавитного кодирования и как определяются кодовые слова?
8. В чем состоит проблема взаимной однозначности алфавитного кодирования?
9. Что называется префиксом слова и когда схема алфавитного кодирования обладает свойствами префикса?
10. Сформулируйте первый достаточный признак взаимной однозначности алфавитного кодирования.
11. Какая схема кодирования называется обратной к схеме алфавитного кодирования и как она обозначается?
12. Сформулируйте второй достаточный признак взаимной однозначности алфавитного кодирования.
13. Что называется нетривиальным разложением элементарного кода?
14. Сформулируйте алгоритм критерия взаимной однозначности алфавитного кодирования.
15. Сформулируйте теорему Маркова.
16. Как по ориентированному циклу написать слово схемы алфавитного кодирования?
17. Как строится двоичное алфавитное кодирование?
18. Каковы свойства самокорректирующихся кодов?
19. Какие коды называются кодами Хемминга? Что называется длиной кода Хемминга? Что называется числом информационных и контрольных символов? Каким соотношением эти символы связаны?
20. Опишите процесс кодирования Хемминга.
21. Опишите процесс декодирования Хемминга, исправляющего не более чем одну ошибку.

Часть 7

Элементы теории автоматов

Лекции 16–18

1. Понятие конечного автомата

Теория автоматов представляет собой раздел дискретной математики, изучающий модели преобразователей дискретной информации. Такими преобразователями являются как реальные устройства (компьютеры, живые организмы), так и воображаемые устройства (аксиоматические теории, математические машины). По сути конечный автомат можно охарактеризовать как устройство M, имеющее входной и выходной каналы при этом в каждый из дискретных моментов времени, называемых тактовыми моментами, оно находится в одном из конечных состояний.

По входному каналу в каждый момент времени $t = 1, 2, \dots$ в устройство M поступают входные сигналы (из некоторого конечного множества сигналов). Задается закон изменения состояния к следующему моменту времени в зависимости от входного сигнала и состояния устройства в текущий момент времени. Выходной сигнал зависит от состояния и входного сигнала в текущий момент времени (рис. 7.1).

Рис. 7.1

2. Определение конечного автомата

Конечный автомат является математической моделью реальных дискретных устройств по переработке информации.

Конечным автоматом называется система $A = (X; Q; Y; \varphi; \psi)$, где $X; Q; Y$ — произвольные непустые конечные множества.

Множество $X = \{a_1, ..., a_m\}$ называется входным алфавитом, а его элементы — входными сигналами, их последовательности — входными словами. Множество $Q = \{q_1, ..., q_n\}$ называется множеством состояний автомата, а его элементы — состояниями. Множество $Y = \{b_1, ..., b_p\}$ называется выходным алфавитом, его элементы — выходными сигналами, их последовательности — выходными словами.

Функция $\varphi : X \times Q \to Q$ называется функцией переходов. Функция $\psi : X \times Q \to Y$ называется функцией выходов, т. е. $\varphi(x, q) \in Q$; $\psi(x, q) \in Y$ для $\forall x \in X$; $\forall q \in Q$.

С конечным автоматом ассоциируется воображаемое устройство, которое работает следующим образом. Оно может находиться в состоянии из множества Q, воспринимать сигналы из множества X и выдавать сигналы из множества Y.

3. Способы задания конечного автомата

Существует несколько эквивалентных способов задания абстрактных автоматов, среди которых можно назвать три: табличный, геометрический и функциональный.

Табличное задание автомата

Из определения автомата следует, что его всегда можно задать таблицей с двумя входами, содержащей m строк и n столбцов, где на пересечении столбца q и строки a стоят значения функций $\varphi(a_i; q_j)$; $\psi(a_i; q_j)$.

$a \diagdown q$	q_1	...	q_j	...	q_n
a_1	$\varphi(a_1; q_1)$; $\psi(a_1; q_1)$	...	$\varphi(a_1; q_j)$; $\psi(a_1; q_j)$	...	$\varphi(a_1; q_n)$; $\psi(a_1; q_n)$
...	...	...	...	...	...
a_i	$\varphi(a_i; q_1)$; $\psi(a_i; q_1)$	...	$\varphi(a_i; q_j)$; $\psi(a_i; q_j)$	...	$\varphi(a_i; q_n)$; $\psi(a_i; q_n)$
...	...	...	...	...	...
a_m	$\varphi(a_m; q_1)$; $\psi(a_m; q_1)$	...	$\varphi(a_m; q_j)$; $\psi(a_m; q_j)$	...	$\varphi(a_m; q_n)$; $\psi(a_m; q_n)$

Задание автомата диаграммой Мура

Другой способ задания конечного автомата — графический. При этом способе состояния автомата изображают кружками, в которые вписывают символы состояний q_j ($j = 1, ..., n$). Из каждого кружка проводится

m стрелок (ориентированных ребер) взаимно-однозначно соответствующих символам входного алфавита $X\{a_1, ..., a_m\}$. Стрелке, соответствующей букве $a_i \in X$ и выходящей из кружка $q_j \in Q$, приписывается пара $(a_i, \psi(a_i; q_j))$, причем эта стрелка ведет в кружок, соответствующий $\varphi(a_i, q_j)$.

Полученный рисунок называется графом автомата или, диаграммой Мура. Для не очень сложных автоматов этот способ более нагляден, чем табличный.

Задание конечного автомата системой булевых функций

Третий способ задания конечного автомата $A = (X; Q; Y; \varphi; \psi)$, заданного таблицей или диаграммой Мура, состоит в определении системы булевых функций.

Изложим алгоритм этого способа задания.

(1) Находим числа k, r, s, удовлетворяющие условиям $2^{k-1} < m < 2^k$; $2^{r-1} < n \leqslant 2^r$; $2^{s-1} < p \leqslant 2^s$, где $m = |X|$; $n = |Q|$; $p = |Y|$.

Очевидно, что k, r, s соответственно равны числу разрядов в двоичном представлении чисел m, n, p. Например, если $m = 5$, $n = 17$, $p = 3$, то $k = 3$, $r = 5$, $s = 2$.

(2) Кодирование состояний входных и выходных символов исходного автомата.

Каждому $q_j \in Q$ взаимно-однозначно ставим в соответствие двоичную последовательность длины r — двоичный код $\alpha(q) = z_1 z_2 ... z_r$. Аналогично каждому $a_i \in X$ и каждому $b_k \in Y$ ставим взаимно однозначно в соответствие двоичные последовательности $\beta(\alpha) = x_1 x_2 ... x_k$; $\gamma(b) = y_1 y_2 ... y_s$.

Отметим, что кодирование состояний, входных и выходных символов можно провести многими способами. При этом некоторые последовательности (коды) могут оказаться неиспользованными.

(3) Составляем следующую таблицу:

Код входного символа				Код текущего состояния				Код следующего состояния				Код выходного символа			
x_1	x_2	...	x_k	z_1	z_2	...	z_r	φ_1	φ_2	...	φ_r	ψ_1	ψ_2	...	ψ_s
0	0	...	0	0	0	...	0								
β_1	β_2		β_k	α_1	α_2		α_r	α'_1	α'_2		α'_r	γ_1	γ_2		γ_s
1	1	...	1	1	1	...	1								

Эта таблица содержит $k + r + r + s$ столбцов и 2^{k+r} строк. В первых $k + r$ столбцах выписаны все наборы длины $k + r$. Каждый такой набор соответствует паре (β, α), где α — возможный код некоторого состояния, β — код входного символа.

(4) Заполнение последних столбцов в таблице (предыдущий шаг).

Для каждой пары (a_i, q_j), где $a_i \in X$; $q_j \in Q$, находим код $\beta(a)$ и $\alpha(q)$. По таблице автомата (или диаграмме Мура) определяем $\varphi(a;q) = q'$ и $\psi(a;q) = b$. Затем находим код $\alpha(q') = \alpha'_1\alpha'_2...\alpha'_r$ и код $\gamma(\beta) = \gamma_1\gamma_2...\gamma_s$. В строку таблицы, соответствующую набору

$$\beta_1, \beta_2, ..., \beta_k, \alpha_1, \alpha_2, ..., \alpha_r,$$

дописываем набор

$$\alpha'_1, \alpha'_2, ..., \alpha'_r, \gamma_1, \gamma_2, ..., \gamma_s.$$

(5) Определение системы булевых функций.

После выполнения предыдущего шага может оказаться, что не все строки в таблице заполнены. Это произойдет в том случае, если хотя бы одно из чисел m, n не является степенью 2. Таким образом, функции $\varphi_1, \varphi_2, ..., \varphi_r, \psi_1, \psi_2, ..., \psi_s$ окажутся не полностью определенными — на некоторых наборах их значения не определены. Тогда мы их доопределяем произвольным образом. Как правило, доопределение функций производят так, чтобы получившиеся полностью определенные функции удовлетворяли тем или иным условиям оптимальности, например представлялись минимальными ДНФ.

После выполнения этого шага исходный автомат будет задаваться системой полностью определенных булевых функций

$$\{\varphi_1, \varphi_2, ..., \varphi_r, \psi_1, \psi_2, ..., \psi_s\}.$$

4. Примеры конечных автоматов

Одним из критериев сложности конечного автомата является число его состояний. Чем меньше это число, тем проще дискретное устройство, реализующее данный автомат. Поэтому одной из важных задач теории конечных автоматов является построение автомата с наименьшим числом состояний.

Поскольку в современных компьютерах любая информация представляется в виде двоичных кодов, то для построения автомата можно использовать элементы, имеющие лишь два различных устойчивых состояния, одно из которых соответствует цифре 0, а другое цифре 1.

Приведем несколько примеров конечных автоматов.

Пример 1. Элемент задержки (элемент памяти).

Элементы задержки представляют собой устройство, имеющее один вход и один выход. Причем значение выходного сигнала в момент времени t совпадает со значением сигнала в предыдущий момент. Схематично элемент задержки можно изобразить следующим образом (рис. 7.2).

Рис. 7.2

Предположим, что входной и, следовательно, выходной алфавит есть $X = \{0, 1\}$; $Y = \{0, 1\}$. Тогда $Q = \{0, 1\}$. Под состоянием элемента задержки в момент времени t понимается содержание элемента памяти в данный момент. Таким образом $q(t) = X(t-1)$, а $Y(t) = q(t) = X(t-1)$.

Зададим элемент задержки таблицей, где $a_1 = 0, a_2 = 1, q_1 = 0, q_2 = 1$,

$$\varphi(a_1; q_1) = \varphi(0, 0) = 0; \quad \psi(a_1, q_1) = \psi(0, 0) = 0;$$
$$\varphi(a_1; q_2) = \varphi(0, 1) = 0; \quad \psi(a_1, q_2) = \psi(0, 1) = 1;$$
$$\varphi(a_2; q_1) = \varphi(1, 0) = 1; \quad \psi(a_2, q_1) = \psi(1, 0) = 0;$$
$$\varphi(a_2; q_2) = \varphi(1, 1) = 1; \quad \psi(a_2, q_2) = \psi(1, 1) = 1,$$

$a \backslash q$	0	1
0	$\varphi = 0; \psi = 0$	$\varphi = 0; \psi = 1$
1	$\varphi = 1; \psi = 0$	$\varphi = 1; \psi = 1$

Диаграмма Мура изображена на рис. 7.3.

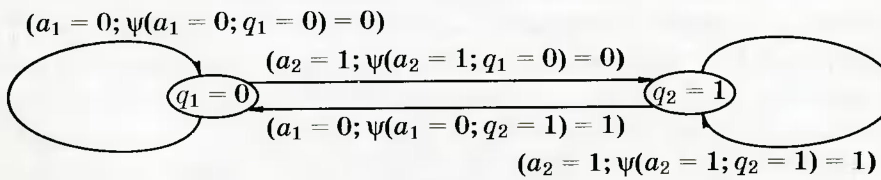


Рис. 7.3

Для представления этого автомата системой булевых функций используем таблицу автомата и вышеизложенный алгоритм. При этом кодирование производить не нужно, так как входной и выходной алфавиты и состояния уже закодированы.

Обратим внимание на то, что $m = n = p = 2$. Тогда $k = r = s = 1$, и поэтому элемент задержки задается двумя функциями φ и ψ. Таблица истинности этих функций содержит $2^{k+r} = 2^2 = 4$ строки и $k+r+r+s = 4$ столбца:

x	z	φ	ψ
0	0	0	0
0	1	0	1
1	0	1	0
1	1	1	1

Пример 2. Двоичный сумматор последовательного действия.

Данный сумматор последовательного действия представляет собой устройство, осуществляющее сложение двух чисел в двоичной системе исчисления. На входы сумматора подаются числа x_1 и x_2, начиная с младших разрядов. На выходе формируется последовательность, соответствующая записи числа $x_1 + x_2$ в двоичной системе исчисления (рис. 7.4).

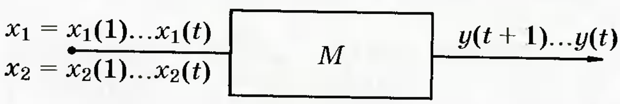


Рис. 7.4

Входной и выходной алфавиты определены однозначно: $X = \{00; 01; 10; 11\}$; $Y = \{0, 1\}$. Множество состояний определяется значением переноса при сложении соответствующих разрядов чисел x_1 и x_2. Если при сложении некоторых разрядов образовался перенос, то будем считать, что сумматор перешел в состояние q_1. При отсутствии переноса будем считать, что сумматор находится в состоянии q_0.

Сумматор задается таблицей.

$a \backslash q$	q_0	q_1
00	q_0; 0	q_0; 1
01	q_0; 1	q_1; 0
10	q_0; 1	q_1; 0
11	q_1; 0	q_1; 1

Диаграмма Мура сумматора последовательного действия изображена на рис. 7.5.

Заметим, что входные и выходные символы уже закодированы. Состояния закодируем следующим образом: $\alpha(q_0) = 0$; $\alpha(q_1) = 1$. Поэтому

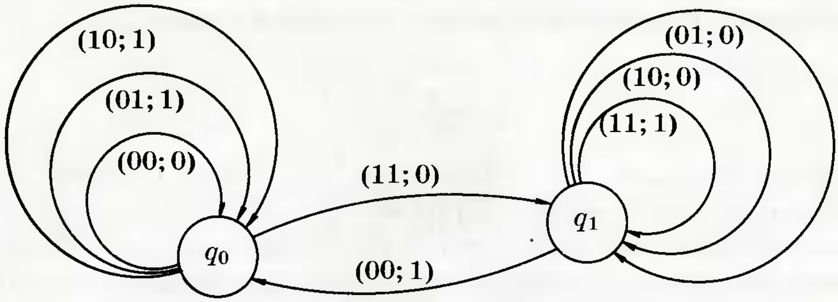


Рис. 7.5

сумматор последовательного действия задается двумя булевыми функциями, таблица истинности которых следующая:

x_1	x_2	z	φ	ψ
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Пример 3. Схема сравнения на равенство.

Схема сравнения на равенство представляет собой устройство, сравнивающее два числа x_1 и x_2, заданные в двоичной системе исчисления. Это устройство работает следующим образом. На вход устройства последовательно, начиная со старших, подаются разряды чисел x_1 и x_2. Эти разряды сравниваются. При совпадении разрядов на выходе схемы формируется выходной сигнал 0, в противном случае на выходе появляется сигнал 1. Ясно, что появление 1 в выходной последовательности означает, что сравниваемые числа x_1 и x_2 различны. Если же выходная последовательность является нулевой и ее длина совпадает с числом разрядов сравниваемых чисел, то $x_1 = x_2$.

Для этого автомата $X = \{00; 01; 10; 11\}$; $Y = \{0, 1\}$.

Функционирование схемы определяется двумя состояниями. Состояние q_0 соответствует равенству сравниваемых в данный момент разрядов. При этом автомат остается в этом же состоянии. Если в следующий момент сравниваемые разряды будут различны, то автомат перейдет в новое состояние q_1 и в нем остается. Так как это означает, что числа различны.

Таким образом, схему сравнения можно задать таблицей:

$x \backslash q$	q_0	q_1
00	$q_0; 0$	$q_1; 1$
01	$q_1; 1$	$q_1; 1$
10	$q_1; 1$	$q_1; 1$
11	$q_0; 0$	$q_1; 1$

Диаграмма Мура схемы сравнения на равенство изображена на рис. 7.6.

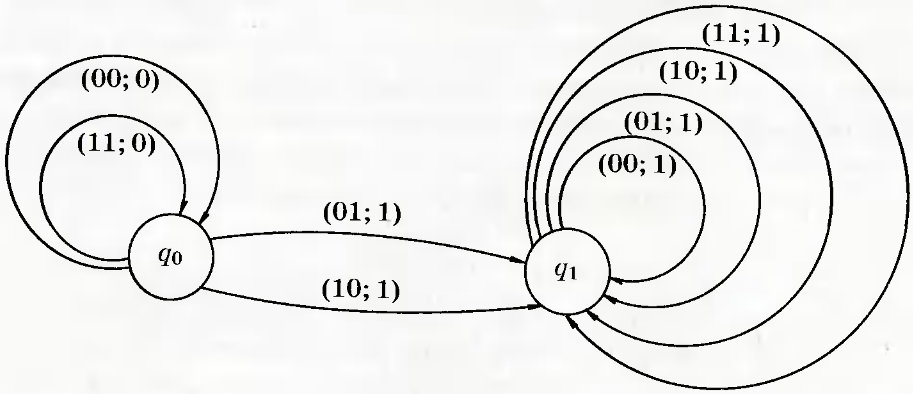


Рис. 7.6

Кодирование состояний произведем следующим образом: $\alpha(q_0) = 0$; $\alpha(q_1) = 1$. Автомат будет задаваться двумя функциями.

x_1	x_2	z	φ	ψ
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	1	1

Пример 4. Схема сравнения на неравенство.

Схема сравнения на неравенство представляет собой устройство, позволяющее выяснить, равны ли сравниваемые x_1 и x_2, и если они не равны, выяснить, какое из них больше другого. Это устройство имеет два входа и два выхода. Выходные сигналы $y_1(t)$ и $y_2(t)$ определяются по следующим правилам:

$$y_1(t) = y_2(t) = 0, \text{ если } x_1(t) = x_2(t);$$

$$y_1(t) = 1; \quad y_2(t) = 0, \text{ если } x_1(t) > x_2(t); \quad \text{т. е. } x_1(t) = 1; \quad x_2(t) = 0;$$
$$y_1(t) = 0; \quad y_2(t) = 1, \text{ если } x_1(t) < x_2(t); \quad \text{т. е. } x_1(t) = 0; \quad x_2(t) = 1.$$

Таким образом, при подаче на вход схемы сравнения на неравенство чисел $x_1 = x_1(1)...x_1(t)$ и $x_2 = x_2(1)...x_2(t)$ последовательно сравниваются разряды этих чисел, начиная со старших. Выходные сигналы формулируются согласно вышеуказанным правилам. При этом, если выходная последовательность состоит из нулевых пар, то $x_1 = x_2$. Если первая, отличная от нулевой, пара имеет вид $\begin{pmatrix}1\\0\end{pmatrix}$, ($\begin{pmatrix}0\\1\end{pmatrix}$) то $x_1 > x_2$ ($x_1 < x_2$).

Из описания схемы следует, что

$$X = \{00; 01; 10; 11\}; \quad Y = \{00; 01; 10\}.$$

Состояние схемы определяется следующим образом. Предположим, что в начальный момент времени $t = 1$ автомат находится в состоянии q_1. Если сравниваемые разряды чисел x_1 и x_2 совпадают, то автомат остается в этом состоянии. Заметим, что на выходе при этом появится сигнал 00. Если же разряд числа x_1 будет меньше (больше) соответствующего разряда числа x_2, то автомат перейдет в состояние q_2 (q_3). При этом на выходе появится сигнал 01 (10). В дальнейшем при подаче оставшихся разрядов чисел x_1 и x_2 на входы автомата автомат будет оставаться в состоянии q_2 (q_3) и вырабатывать выходной символ 10 (01). Из вышеизложенного следует, что схему сравнения на неравенство можно задать таблицей:

$x \backslash q$	q_1	q_2	q_3
00	q_1; 00	q_2; 01	q_3; 10
01	q_2; 01	q_2; 01	q_3; 10
10	q_3; 10	q_2; 01	q_3; 10
11	q_1; 00	q_2; 01	q_3; 10

Соответствующая диаграмма Мура изображена на рис. 7.7.

Входной и выходной алфавиты здесь уже закодированы. Состояния q_1, q_2 и q_3 закодируем: $\alpha_1(q_1) = 00$; $\alpha(q_2) = 01$; $\alpha(q_3) = 10$.

Следовательно, данную схему можно задать системой, состоящей из четырех булевых функций, которые зависят от четырех переменных. Эти

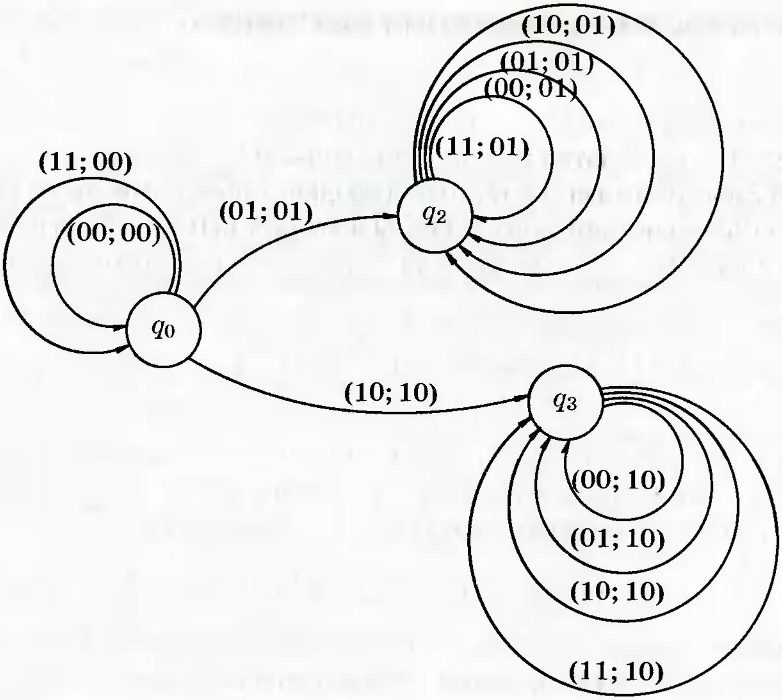


Рис. 7.7

функции частично определены и задаются таблицей истинности

	x_1	x_2	z_1	z_2	φ_1	φ_2	ψ_1	ψ_2
	0	0	0	0	0	0	0	0
	0	0	0	1	0	1	0	1
	0	0	1	0	1	0	1	0
*	0	0	1	1	1	1	1	1
	0	1	0	0	0	1	0	1
	0	1	0	1	0	1	0	1
	0	1	1	0	0	1	0	1
*	0	1	1	1	1	1	1	1
	1	0	0	0	1	0	1	0
	1	0	0	1	0	1	0	1
	1	0	1	0	1	0	1	0
*	1	0	1	1	1	1	1	1
	1	1	0	0	0	0	0	0
	1	1	0	1	0	1	0	1
	1	1	1	0	1	0	1	0
*	1	1	1	1	1	1	1	1

В таблице символами * отмечены наборы переменных x_1, x_2, z_1, z_2, на которых функции φ_1, φ_2, ψ_1, ψ_2 не определены. Положим значения функций φ_1, φ_2, ψ_1, ψ_2, на этих наборах равными 1.

5. Канонические уравнения автомата

Если в момент времени $t = 1, 2, \ldots$ на вход автомата $A = (X; Q; Y; \varphi; \psi)$ последовательно подаются входные символы $x(t) \in X$ и при этом автомат находится в состоянии $q(t) \in Q$, то под воздействием символа $x(t)$ автомат перейдет в новое состояние $q(t+1) \in Q$ и выдаст выходной сигнал $y(t)$.

Величины $x(t)$, $y(t)$, $q(t)$, $q(t+1)$ связаны между собой следующими уравнениями:

$$\begin{cases} q(t+1) = \varphi\,(x(t); q(t))\,, \\ y(t) = \psi\,(x(t); q(t))\,, \end{cases} \qquad t = 1, 2, \ldots, n, \ldots$$

Эти уравнения называются каноническими уравнениями автомата A. При задании автомата системой булевых функций эти уравнения записываются в координатной форме:

$$\begin{gathered} z_1(t+1) = \varphi_1\,(x_1(t), \ldots, x_k(t), z_1(t), \ldots, z_r(t)) \\ \ldots \\ z_r(t+1) = \varphi_r\,(x_1(t), \ldots, x_k(t), z_1(t), \ldots, z_r(t)) \\ y_1(t) = \psi_1\,(x_1(t), \ldots, x_k(t), z_1(t), \ldots, z_r(t)) \\ \ldots \\ y_s(t) = \psi_s\,(x_1(t), \ldots, x_k(t), z_1(t), \ldots, z_r(t)) \\ t = 1, 2, \ldots \end{gathered}$$

Для построения канонических уравнений автомата A необходимо для данной булевой функции найти минимальную ДНФ (дизъюнктивную нормальную форму), которая, вообще говоря, определяется неоднозначно. Аналитический алгоритм построения этой ДНФ следующий:

1. Для данной функции $f(x_1, \ldots, x_n)$ строим совершенную конъюнктивную нормальную форму (СКНФ).

2. В построенной СКНФ раскрываем скобки, используя правило:

$$(A \vee B) \wedge (C \vee D) = A \wedge C \vee B \wedge C \vee A \wedge D \vee B \wedge D.$$

3. Полученное выражение упрощаем, применяя тождества вида:

$$K_1 \wedge K_2 \vee K_1 = K_1; \quad K \vee K = K; \quad K \vee 0 = K; \quad K \wedge \overline{K} = 0; \quad K \vee 1 = 1;$$
$$K \wedge 1 = K; \quad K \wedge K = K; \quad K \vee \overline{K} = 1.$$

В результате получим сокращенную ДНФ, являющуюся дизъюнкцией всех простых импликат данной функции $f(x_1, \ldots, x_n)$.

Для рассмотренных выше примеров автоматов канонические уравнения задаются следующими формулами:

пример 1: $\begin{cases} z(t+1) = x(t), \\ y(t) = z(t), \end{cases} \quad t = 1, 2, \ldots$

пример 2: $\begin{cases} z(t+1) = x_1(t) \wedge x_2(t) \vee x_1(t) \wedge z(t) \vee x_2(t) \wedge z(t), \\ y(t) = x_1(t) \oplus x_2(t) \oplus z(t), \end{cases} \quad t = 1, 2, \ldots$

пример 3: $\begin{cases} z(t+1) = \overline{x_1}(t) \wedge x_2(t) \vee x_1(t) \wedge \overline{x_2}(t) \vee z(t), \\ y(t) = \overline{x_1}(t) \wedge x_2(t) \vee x_1(t) \wedge \overline{x_2}(t) \vee z(t), \end{cases} \quad t = 1, 2, \ldots$

пример 4: $\begin{cases} z_1(t+1) = z_1(t) \vee x_1(t) \wedge \overline{x_2}(t) \wedge \overline{z_2}(t), \\ z_1(t+1) = z_2(t) \vee \overline{x_1}(t) \wedge x_2(t) \wedge \overline{z_1}(t), \\ y_1(t) = z_1(t) \vee x_1(t) \wedge \overline{x_2}(t) \wedge \overline{z_2}(t), \\ y_2(t) = z_2(t) \vee \overline{x_1}(t) \wedge x_2(t) \wedge \overline{z_1}(t), \end{cases} \quad t = 1, 2, \ldots$

В качестве иллюстрации изложенного выше алгоритма рассмотрим пример 3.

Таблица истинности системы булевых функций следующая:

x_1	x_2	z	φ	ψ
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	1	1

1. Строим СКНФ функции $\varphi(x_1, x_2, z)$. Так как эта функция задана набором своих значений $\overline{\varphi} = 01111101$, то ее СКНФ будет иметь вид $(x_1 \vee x_2 \vee z) \wedge (\overline{x_1} \vee \overline{x_2} \vee z)$.

2. Раскрываем скобки

$$(x_1 \vee x_2 \vee z) \wedge \overline{x_1} \vee (x_1 \vee x_2 \vee z) \wedge \overline{x_2} \vee (x_1 \vee x_2 \vee z) \wedge z =$$
$$= x_1 \wedge \overline{x_1} \vee x_2 \wedge \overline{x_1} \vee z \wedge \overline{x_1} \vee x_1 \wedge \overline{x_2} \vee x_2 \wedge \overline{x_2} \vee z \wedge \overline{x_2} \vee x_1 \wedge z \vee x_2 \wedge z \vee z \wedge z.$$

Упрощаем последнее выражение:

$$0 \vee \overline{x_1} \wedge x_2 \wedge \overline{x_1} \wedge z \vee x_1 \wedge \overline{x_2} \vee 0 \vee \overline{x_2} \wedge z \vee x_1 \wedge z \vee x_2 \wedge z \vee z =$$
$$= \overline{x_1} \wedge x_2 \vee x_1 \wedge \overline{x_2} \vee z(\overline{x_2} \vee x_2) \vee z(\overline{x_1} \vee x_1) \vee z = \overline{x_1} \wedge x_2 \vee x_1 \wedge \overline{x_2} \vee z.$$

Таким образом, получим $z(t+1) = \overline{x_1}(t) \wedge x_2(t) \vee x_1(t) \wedge \overline{x_2}(t) \vee z(t)$.

Аналогично строится функция $y(t)$. При этом из таблицы истинности выписываем набор значений функции $\psi(x_1, x_2, z)$ $\overline{\psi} = 01111101$, который совпадает с набором значений функции $\varphi(x_1, x_2, z)$.

Десятое практическое занятие по теме «Конечные автоматы»

Задача 1. Для автомата, заданного таблицей, постройте диаграмму Мура. Задайте этот автомат системой булевых функций.

а)

$\alpha \backslash q$	0	1	2	3
0	(1;1)	(3;0)	(2;0)	(2;0)
1	(2;1)	(2;0)	(3;0)	(3;0)

б)

$\alpha \backslash q$	0	1	2	3
0	(1;0)	(3;1)	(2;0)	(1;0)
1	(3;0)	(1;1)	(0;1)	(3;1)

в)

$\alpha \backslash q$	0	1	2	3
0	(0;0)	(1;1)	(3;1)	(2;0)
1	(2;0)	(0;1)	(3;1)	(1;0)

г)

$\alpha \backslash q$	0	1	2	3
0	(2;0)	(0;0)	(3;1)	(1;0)
1	(1;0)	(0;0)	(0;0)	(3;0)

д)

$\alpha \backslash q$	0	1	2	3
0	(3;0)	(2;0)	(1;1)	(0;1)
1	(0;1)	(1;1)	(2;0)	(3;0)

е)

$\alpha \backslash q$	0	1	2	3
0	(2;1)	(2;1)	(2;1)	(2;1)
1	(1;1)	(3;1)	(0;0)	(1;0)

Задача 2. Для автомата, заданного диаграммой Мура, выпишите соответственную таблицу и систему булевых функций.

а) См. рис. 7.8.

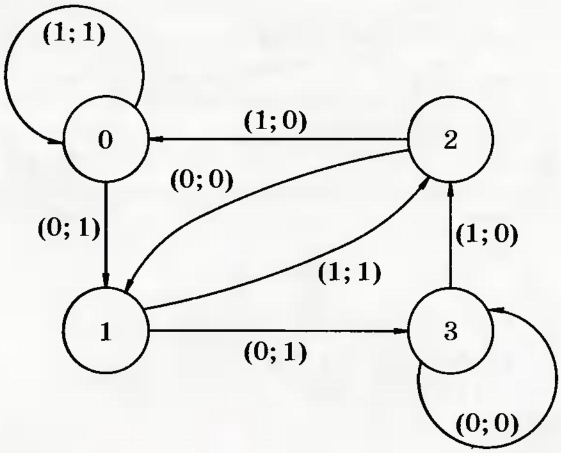


Рис. 7.8

б) См. рис. 7.9.
в) См. рис. 7.10.

Задача 3. Для автомата, заданного каноническими уравнениями, постройте диаграмму Мура и соответствующую таблицу:

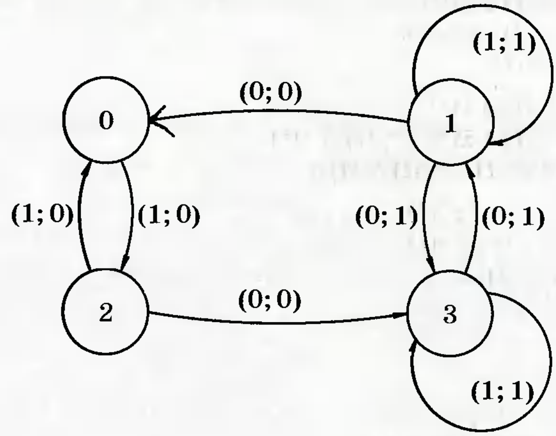


Рис. 7.9

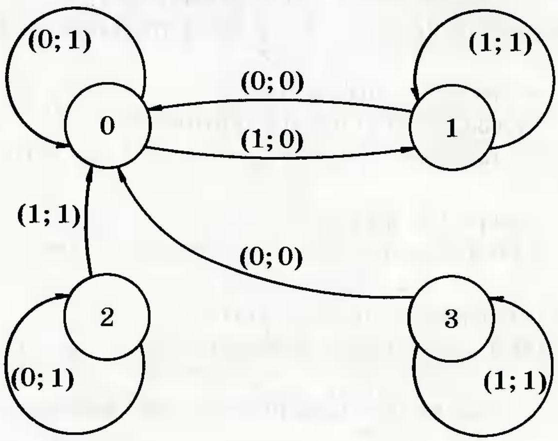


Рис. 7.10

а) $\begin{cases} z(t+1) = \overline{x_1}(t) \wedge \overline{z}(t) \vee \overline{x_2}(t) \wedge z(t), \\ y(t) = \overline{x_2}(t) \wedge z(t) \vee x_1(t) \wedge x_2(t); \end{cases}$

б) $\begin{cases} z(t+1) = x(t), \\ y(t) = x(t) \leftrightarrow z(t); \end{cases}$

в) $\begin{cases} z_1(t+1) = \overline{x}(t) \wedge (z_1(t) \leftrightarrow z_2(t)) \vee z_2(t) \wedge x(t), \\ z_2(t+1) = \overline{x}(t) \wedge z_2(t) \vee x(t) \wedge z_1(t), \\ y_1(t) = \overline{x}(t) \wedge (z_1(t) \vee z_2(t)) \vee x(t) \wedge z_1(t), \\ y_2(t) = \overline{x}(t) \wedge z_1(t) \wedge z_2(t) \vee x(t) \wedge z_1(t) \wedge \overline{z_2}(t); \end{cases}$

г) $\begin{cases} z_1(t+1) = x(t) \wedge (z_2(t) \vee z_1(t)) \wedge \overline{z_2}(t), \\ z_2(t+1) = z_1(t) \wedge [\overline{z_2}(t) \wedge x(t) \vee \overline{x}(t) \wedge z_2(t)] \vee z_1(t) \wedge x(t), \\ y(t) = z_1(t) \wedge x(t) \vee z_1(t) \wedge z_2(t) \wedge \overline{x}(t); \end{cases}$

д) $\begin{cases} z_1(t+1) = z_1(t) \downarrow x(t), \\ z_2(t+1) = z_1(t) \to z_2(t), \\ y(t) = x(t) \oplus z(t); \end{cases}$

е) $\begin{cases} z_1(t+1) = z_2(t) \wedge x(t), \\ z_2(t+1) = \overline{z_1}(t) \wedge \overline{x}(t) \vee z_1(t) \wedge x(t), \\ y(t) = \overline{z_1}(t) \wedge \overline{z_2}(t) \vee z_1(t) \wedge \overline{x}(t); \end{cases}$

ж) $\begin{cases} z_1(t+1) = z_1(t) \oplus z_2(t) \oplus x(t), \\ z_2(t+1) = z_2(t) \to x(t), \\ y(t) = z_2(t) \leftrightarrow x(t). \end{cases}$

Контрольные вопросы

1. Что включает в себя понятие «конечный автомат»?

2. Какие основные термины связаны с введением понятия конечного автомата?

3. Дайте определение конечного автомата.

4. Укажите способы задания конечного автомата.

5. Каково табличное задание конечного автомата? Укажите способы задания конечного автомата.

6. Как строится диаграмма Мура?

7. Изложите алгоритм задания конечного автомата системой булевых функций.

8. Приведите примеры конечных автоматов.

9. Какие уравнения называются каноническими уравнениями конечного автомата?

10. Как построить каноническое уравнение конечного автомата?

Часть 8

Элементы теории алгоритмов

Лекции 19–23

Этот раздел дискретной математики связан с решением различных вычислительных задач. Что значит вычислить? Какими вопросами для этого мы располагаем? Всегда ли можно решить задачу? Что такое задача? Что такое процесс ее решения?

На все поставленные вопросы может дать ответ теория алгоритмов, представляющая собой теоретические основы всей дискретной математики, особенно ее компьютерной части.

I. Вычислимые функции и алгоритмы

Одним из центральных понятий современной математики является понятие вычислимой функции и алгоритма.

1. Основные определения

Определение. *Вычислимая функция* — это такая функция, для которой существует вычисляющий ее значения алгоритм, т. е. функция $y = f(x_1, x_2, \ldots, x_n)$ называется вычислимой, если существует алгоритм, позволяющий определить значения функции при любых значениях переменных $x_1, x_2, \ldots, x_n$.

Определение. *Алгоритмом* называется точное предписание, определяющее вычислительный процесс, который ведет от варьируемых исходных данных к искомому результату, т. е. алгоритм — это совокупность правил, определяющих данный вычислительный процесс.

Для каждого алгоритма существует некоторая совокупность возможных исходных данных — объектов, к которым имеет смысл применять рассматриваемый алгоритм. Для каждого алгоритма выделяется *область применимости*: если процесс применения алгоритма к какому-либо объекту заканчивается выдачей результатов, то говорят, что он применим к этому объекту.

Алгоритм задает функцию, определенную на его области применимости и ставящую в соответствие каждому элементу этой области результат применения к нему алгоритма.

Не все объекты, встречающиеся в математике, могут служить исходными данными, результатами или промежуточными данными алгоритма.

В алгоритмических процессах могут участвовать лишь *конструктивные объекты*.

Определение. *Конструктивными объектами* будем называть натуральные и рациональные числа, полиномы с натуральными или рациональными коэффициентами, матрицы с натуральными или рациональными элементами, слова в некотором алфавите и т. д., т. е. такие объекты, которые могут быть построены целиком и представлены для рассмотрения.

Поскольку возможными исходными данными и результатами алгоритма могут быть лишь конструктивные объекты, то лишь конструктивные объекты могут быть аргументами и значениями вычислимой функции.

2. Свойства алгоритмов

Анализ известных в математике алгоритмов дал возможность выявить характерные его свойства.

Свойство 1. *Дискретность.* Алгоритм описывает процесс последовательного построения величин, идущий в дискретном времени. Необходимый для вычисления интервал времени разбит на малые отрезки — такты. Система величин в конце каждого такта получается в результате осуществления элементарного шага алгоритма (определенной программы преобразования) из системы величин, имеющейся к началу такта.

Свойство 2. *Детерминированность (определенность)* требуется, чтобы метод действия (вычисления) был настолько точен и общепонятен, чтобы не оставалось места произволу. Этот метод можно сообщить другому лицу в виде конечного числа указаний, т. е. программа преобразований в каждом такте однозначно определена.

Свойство 3. *Результативность.* Это свойство, называемое иногда еще направленностью алгоритма, требует, чтобы алгоритмическая процедура, примененная к любой задаче данного типа, через конечное число шагов останавливалась и после остановки можно было бы прочесть искомый результат.

Свойство 4. *Массовость.* Алгоритм служит не для решения какой-либо одной конкретной задачи, а для решения целого класса задач. Процедуру для решения одной индивидуальной задачи не называют алгоритмом.

3. Понятия разрешимого предиката, разрешимого множества, перечислимого множества

На основе понятий вычислимой функции и алгоритма определяются понятия разрешимого предиката, разрешимого множества и перечислимого множества.

Определение. Предикат $P(x_1, x_2, ..., x_n)$, определенный на множестве целых чисел, называется алгоритмически разрешимым или просто разрешимым, если существует алгоритм для определения значения предиката P при любых значениях переменных $x_1, x_2, ..., x_n$.

Определение. Множество называется разрешимым, если существует алгоритм, распознающий принадлежность произвольного элемента к этому множеству.

Определение. Множество называется перечислимым, если оно есть множество значений какой-нибудь вычислимой функции, определенной на всем натуральном ряду.

4. Пример алгоритма

Прежде, чем перейти к точным определениям, рассмотрим пример вычислительного алгоритма: алгоритм вычисления числа π, основанный на формуле удвоения.

Исторически первый и в течение длительного времени единственный алгоритм вычисления числа π был основан на подсчете периметров правильных вписанных и описанных многоугольников с помощью формул удвоения. Эта формула связывает длины сторон правильных n- и $2n$-угольников, вписанных в окружность радиуса R. Положим диаметр рассматриваемой окружности равным единице: $d = 2R = 1$ (длина такой окружности равна π), тогда формула удвоения примет вид:

$$a_{2n} = \frac{1}{2}\sqrt{2 - 2\sqrt{1 - a_n^2}}.$$

Умножим и разделим выражение под знаком общего корня на сопряженное. В результате получим:

$$a_{2n} = \frac{\sqrt{4 - 4(1 - a_n^2)}}{2\sqrt{2 + 2\sqrt{1 - a_n^2}}} = \frac{a_n}{\sqrt{2 + 2\sqrt{1 - a_n^2}}}.$$

Введем периметр многоугольника $p_n = na_n$ и подставим выражения сторон a_n и a_{2n} через периметры в формулу, тогда она преобразуется к виду

$$p_{2n} = \frac{p_n}{\sqrt{\frac{1}{2} + \frac{1}{2}\sqrt{1 - \frac{p_n^2}{n^2}}}}.$$

Мы получили рекуррентную формулу.

Сторона правильного описанного n-угольника b_n при $d = 2R = 1$ выражается через сторону вписанного n-угольника a_n с помощью соотношения

$$b_n = \frac{a_n}{\sqrt{1 - a_n^2}}.$$

Перейдем в нем от сторон a_n и b_n к соответствующим периметрам, обозначая периметр вписанного многоугольника через q_n: $q_n = nb_n$. В результате будем иметь:

$$q_n = \frac{p_n}{\sqrt{1 - \frac{p_n^2}{n^2}}}.$$

Но $\lim\limits_{n\to\infty} p_n = \lim\limits_{n\to\infty} q_n = \pi$.

Вычисление числа π с помощью данного метода можно начать с какого-нибудь простого правильного многоугольника, например с шестиугольника, для которого

$$p_6 = 3, \quad q_6 = \frac{3}{\sqrt{1 - \frac{1}{4}}} = 2\sqrt{3} = 3{,}464101\ldots$$

Далее процесс строится следующим образом: по рекуррентной формуле находятся последовательно $p_{12}, p_{24}, p_{48}, p_{96}, p_{192}, \ldots$. По ним с помощью формулы вычисляются соответствующие значения q_n.

Двухстороннее неравенство $p_n < \pi < q_n$ позволяет утверждать, что найденные на некотором шаге числа дают приближенные значения π с недостатком и избытком с ошибкой $\varepsilon_n < \Delta = q_n - p_n$, которая стремится к нулю при $n \to \infty$.

Используя описанные идеи и проводя сложнейшие для своего времени вычисления, древнегреческий ученый Архимед дошел до правильного 96-угольника и получил для π двухстороннюю оценку:

$$3\frac{10}{71} < \pi < 3\frac{1}{7}, \quad \left(3\frac{10}{71} = 3{,}14084\ldots, \quad 3\frac{1}{7} = 3{,}14285\ldots, \quad \Delta \approx 0{,}002\right).$$

В Европе французский математик Ф. Виет нашел 9 правильных цифр числа π после запятой с помощью $3 \cdot 2^{17}$-угольника (16 удвоений числа сторон).

Приведем таблицу 16 удвоений числа сторон: повторение результата Ф. Виета.

k	$n = 6 \cdot 2^k$	p_n	q_n
0	6	3,000 000 000 000	3,464 101 615 138
1	12	3,105 828 541 230	3,630 002 002 236
2	24	3,132 628 613 281	3,245 155 564 194
3	48	3,139 350 203 047	3,166 557 423 678
4	96	3,141 031 950 890	3,147 778 817 495
5	192	3,141 452 472 285	3,143 135 797 312
6	384	3,141 557 607 912	3,141 978 227 840
7	768	3,141 583 892 148	3,141 689 033 932
8	1536	3,141 590 463 228	3,141 616 747 849
9	3072	3,141 592 105 999	3,141 598 677 103
10	6144	3,141 592 516 692	3,141 594 159 465
11	12288	3,141 592 619 365	3,141 593 030 059
12	24576	3,141 592 645 034	3,141 592 747 706
13	49152	3,141 592 651 034	3,141 592 677 119
14	98304	3,141 592 653 055	3,141 592 659 472
15	196608	3,141 592 653 456	3,141 592 655 060
16	393216	3,141 592 653 556	3,141 592 653 957

Для оценки точности определения π с помощью этих расчетов составим разность периметров q_n и p_n, приведенных в последней строке ($n = 6 \cdot 2^{16} = 39321$):

$$\varepsilon_n < \Delta_n = q_n - p_n = 0{,}000\,000\,000\,401.$$

Первые 10 знаков у чисел p_n и q_n совпадают, т. е. $\pi = 3{,}141\,592\,653\ldots$.

5. Интуитивное понятие алгоритма

Приведенные выше определения и свойства алгоритмов являются эмпирическими, подмеченными для всех известных алгоритмов. Таким образом, понятие алгоритма и понятие вычислимой функции выводятся непосредственно из опыта и могут быть усвоены лишь на примерах, а поэтому эти понятия являются *интуитивными* и не могут быть положены в основу *математической* теории алгоритмов.

После того как было сформулировано интуитивное понятие алгоритма, были предприняты усилия дать его точное математическое определение.

В результате появились несколько вариантов определения алгоритма, которые трактуются как самостоятельные математические понятия. Каждое из них имеет свое название: «Теория рекурсивных функций», «Нормальные алгоритмы Маркова», «Машины Тьюринга».

II. Теория рекурсивных функций

1. Простейшие функции

Прежде всего рассмотрим следующие простейшие функции, которые называются базисными.

(a) нуль — функция 0 ($0(x) = 0$ для всех x);

(b) функция следования $x' = x + 1$, которая означает переход к следующему элементу заданного множества;

(c) функция тождества, или функция выделения аргумента: для каждых $n \geqslant 1$ и $1 \leqslant i \leqslant n$ функция J_i^n, определяемая посредством $J_i^n(x_1, x_2, \ldots, x_n) = x_i$.

Эти функции вычислимы, и на их основе построим более сложные вычислимые функции — с помощью преобразований, которые называются операторами.

2. Операторы

а) *Оператор суперпозиции (подстановки).* В основе этого оператора лежит способ вычисления, который состоит в том, что если мы умеем вычислять функции f и g, то мы сумеем вычислить функцию $h = f(g)$.

Пусть f — m-местная функция, $g_1, \ldots, g_n$ — n-местные частичные функции на множестве N. Оператор S, ставящий в соответствие функциям f и $g_1, \ldots, g_n$ — n-местную функцию h, которая удовлетворяет тождеству

$$h(x_1, \ldots, x_n) = f\,(g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n))\,,$$

называется оператором суперпозиции (подстановки). При этом определенная всюду функция $h = S(f, g_1, \ldots)$ является суперпозицией функций f и $g_1, \ldots, g_m$.

б) *Оператор примитивной рекурсии.* В основе этого оператора лежит широко распространенный способ вычисления функции натурального аргумента, состоящий в том, что задается $f(0)$ и так называемая рекуррентная формула $f(n+1) = h(f(n))$, позволяющая вычислить $f(n+1)$, если известно $f(n)$. С помощью этой формулы последовательно вычисляются $f(1)$, $f(2)$ и т. д. Такой процесс последовательного вычисления значений функции называется рекурсией.

Пусть заданы какие-либо числовые частичные функции: n-местная g и $(n+2)$-местная h. $(n+1)$-местная частичная функция f возникает из функций g и h с помощью оператора примитивной рекурсии, если она может быть задана *схемой примитивной рекурсии*:

$$\begin{cases} f(x_1, \ldots, x_n, 0) = g(x_1, \ldots, x_n), \\ f(x_1, \ldots, x_n, 1) = h\,(x_1, \ldots, x_n, 0, f(x_1, \ldots, x_n, 0))\,, \\ f(x_1, \ldots, x_n, 2) = h\,(x_1, \ldots, x_n, 1, f(x_1, \ldots, x_n, 1))\,, \\ \ldots \\ f(x_1, \ldots, x_n, y+1) = h\,(x_1, \ldots, x_n, y, f(x_1, \ldots, x_n, y))\,. \end{cases}$$

Совокупность этих равенств для любых функций g и h однозначно определяет значения функции f. Таким образом, для каждых двух частичных числовых функций g от n переменных и h от $(n+2)$ переменных существует одна и только одна функция f от $(n+1)$ переменной, возникающая в результате примитивной рекурсии. Оператор примитивной рекурсии будем обозначать через $f = R(g, h)$.

Итак, схема примитивной рекурсии образует некоторый, индуктивный процесс построения функции h, при котором на нулевом шаге используется функция g, а на каждом последующем шаге — значение функции f от аргументов $x_1, \ldots, x_n$, номер y предыдущего шага и значение функции h, вычисленного на предыдущем шаге. Заметим, что оператор примитивной рекурсии можно применять по любым переменным, входящим в функции f, g и h, но всегда нужно указывать, по каким переменным этот оператор проводится.

в) *Оператор минимизации.* Существует третья важная операция, порождающая новые вычислимые функции, а именно неограниченная минимизация, или просто минимизация.

Пусть $f(x, y)$ — функция, и мы хотим определить функцию $g(x)$, положив $g(x)$ = наименьшее y такое, что $f(x, y) = 0$. При этом из вычислимости f должна следовать вычислимость g. Для этого введем следующее определение оператора минимизации μ, который дает вычислимые функции из вычислимых функций.

Для каждой функции $f(x, y)$

$$\mu f(x,y) = 0 = \begin{cases} \text{наименьший } y\text{, такой, что } f(x,y) \text{ определенно для всех} \\ z \leqslant y \text{ и } f(x,y) = 0\text{, если такой } y \text{ существует,} \\ \text{не определено в противном случае.} \end{cases}$$

$\mu y(\ldots)$ читается как «наименьший y, такой, что...».

Этот оператор называют иногда μ-оператором.

Сформулируем следующую теорему: Пусть $f(x, y)$ вычислима; тогда вычислима и функция $g(x) = \mu y\,(f(x, y) = 0)$.

3. Примитивно-рекурсивные функции

Определение. Функция f называется примитивно-рекурсивной, если существует последовательность функций $f_0, f_1, \ldots, f_n$, в которой $f_n = f$ и всякая функция f_i является либо простейшей (базисной), либо получается из предыдущих функций с помощью конечного числа применений оператора суперпозиции S или оператора примитивной рекурсии R (простейшие функции являются примитивно-рекурсивными).

Итак, примитивно-рекурсивные функции — это арифметические функции, которые сопоставляются по определенным правилам примитивно-рекурсивным описанием.

Всякая примитивно-рекурсивная функция является вычислимой, так как процесс ее определения с помощью примитивно-рекурсивного описания дает одновременно и способ вычисления значений функции для любых числовых значений ее аргументов.

4. Частично-рекурсивные функции

Определение. Функция $f(x_1, \ldots, x_n)$ называется частично-рекурсивной, если она может быть получена из простейших примитивно-рекурсивных функций с помощью конечного числа применений операторов суперпозиции, примитивной рекурсии и минимизации.

Из определения вытекает следующие свойства частично-рекурсивных функций.

1. Каждая примитивно-рекурсивная функция является также и частично-рекурсивной.

2. Класс частично-рекурсивных функций шире класса примитивно-рекурсивных функций, так как все примитивно-рекурсивные функции всюду определенные, а среди частично-рекурсивных функций есть функции, не всюду определенные.

Понятие частично-рекурсивной функции является одним из основных понятий теории алгоритмов.

Какие бы классы алгоритмов до сих пор не строились, во всех случаях оказывалось, что числовые функции, вычисляемые посредством этих алгоритмов, были частично-рекурсивными. Поэтому общепринятым является тезис Черча относительно частично-рекурсивных функций: *класс алгоритмически вычислимых частичных числовых функций совпадает с классом всех частично-рекурсивных функций.*

Для каждой частично-рекурсивной функции f существует вычислительный процесс, посредством которого любое натуральное число x перерабатывается в значение $f(x)$ функции f. Этот процесс не дает определенного результата тогда и только тогда, когда значение функции f в точке x не определено.

5. Примитивно-рекурсивные предикаты

Наряду с примитивно-рекурсивными функциями в рассмотрение вводятся примитивно-рекурсивные предикаты.

Предикаты в математической логике вводятся там, где необходимо символически отобразить соотношение между несколькими предметами. В общем случае предикат определен на некотором множестве предметов и может принимать два значения: истинно или ложно (1 или 0).

Мы будем рассматривать арифметические предикаты, определенные на множестве натуральных чисел $N_0 = N \cup \{0\} = \{0, 1, 2, \dots\}$.

Определение. Пусть $P(x_1, \dots, x_n)$ — n-местный предикат (зависит от n переменных) на множестве натуральных чисел N_0. Функция $\varphi_p(x_1, \dots, x_n)$ называется характеристической функцией для предиката P, если эта функция удовлетворяет условию

$$\varphi_P(x_1, \dots, x_n) = \begin{cases} 0, & \text{если } P(x_1, \dots, x_n) \text{ истинно;} \\ 1, & \text{если } P(x_1, \dots, x_n) \text{ ложно.} \end{cases}$$

Таким образом, характеристическая функция, или еще ее называют представляющей функцией предиката $P(x_1, ..., x_n)$, $\varphi_P(x_1, ..., x_n)$, обращается в нуль лишь для тех и только для тех $x_1, ..., x_n$, для которых $P(x_1, ..., x_n)$ истинно. Тогда истинность $P(x_1, ..., x_n)$ соответствует равенству $\varphi_P(x_1, ..., x_n) = 0$.

Очевидно, что один и тот же предикат может иметь несколько характеристических функций, нули которых должны совпадать.

Определение. Предикат называется примитивно-рекурсивным, если существует примитивно-рекурсивная функция, представляющая этот предикат, т. е. его характеристическая функция примитивно-рекурсивна.

Если предикаты $P_1, ..., P_n$ примитивно-рекурсивны, то и любой предикат, полученный из них с помощью логических операций, примитивно-рекурсивен. Операции исчисления высказываний (отрицание, конъюнкция, дизъюнкция, импликация) сохраняют примитивную рекурсивность предикатов.

Определение. Предикат $P(x_1, ..., x_n)$ разрешим, если характеристическая функция $\varphi(x_1, ..., x_n)$ вычислима; предикат $P(x_1, ..., x_n)$ неразрешим, если функция $\varphi_P(x_1, ..., x_n)$ невычислима.

Одиннадцатое практическое занятие по теме «Рекурсивные функции»

Задача 1. Найдите функции g и h в рекурсивной формуле для двухместной функции $f(x, y) = x \cdot y$, если рекурсия проводится по переменной x.

Решение. Алгоритм решения:

1. Найдите функцию $g(x) = f(0, y)$ (т. к. рекурсия проводится по x).

$$g(y) = f(0, y) = 0 \cdot y = 0.$$

2. Подставить в функцию $f(x, y)$ вместо x значение $x + 1$.

$$f(x + 1, y) = (x + 1) \cdot y = x \cdot y + y.$$

3. Выразить функцию $f(x + 1, v)$ через $f(x, y)$, x, y

$$f(x + 1, y) = f(x, y) + y.$$

4. Найти функцию $h(x, y, z)$ из формулы $f(x + 1, y) = h\,(x, y, f(x, y))$, заменив $f(x, y)$ на z.

$$h(x, y, z) = z + y.$$

Задача 2. Найдите функции g и h в рекурсивной формуле для трехместной функции $f(x, y, z) = x \cdot y + z$, если рекурсия проводится по переменной y.
Решение. Алгоритм решения.

1. Найти функцию g: $g(x, z) = f(x, 0, z)$ (так как рекурсия проводится по y)

$$g(x, z) = f(x, 0) = x \cdot 0 + z = z.$$

2. Подставить в функцию $f(x, y, z)$ вместо у значение $y + 1$.

$$f(x, y + 1, z) = x(y + 1) + z = xy + z + x.$$

3. Выразить функцию $f(x, y + 1, z)$ через $f(x, y + 1, z)$, x, y, z.

$$f(x, y + 1, z) = (xy + z) + x = f(x, y, z) + x.$$

4. Найти функцию $h(x, y, z, t)$ из формулы

$$f(x, y + 1, z) = h\,(x, y, z, f(x, y, z))\,.$$

Заменив $f(x, y, z)$ на t, получим $h(x, y, z, t) = t + x$.

Задача 3. Примените оператор примитивной рекурсии к простейшим функциям $g(x) = J_1^1(x) = x$ и $h(x, y, z) = z' = z + 1$, постройте функцию $f'(x, y) = R(g, h)$, если рекурсия проводится по переменной y.
Решение. Для построения функции $f(x, y) = R(g, h)$ составим примитивно-рекурсивное описание функции $f(x, y)$ по переменной y, применяя оператор примитивной рекурсии к функциям $g(x) = x$; $h(x, y, z) = z + 1$ по переменной y.

$$\begin{gathered}
f(0, x) = g(x) = x,\\
f(1, x) = h\,(x, 1, f(0, x)) = f(0, x) + 1 = x + 1,\\
f(2, x) = h\,(x, 2, f(1, x)) = f(1, x) + 1 = x + 1 + 1 = x + 2,\\
f(3, x) = h\,(x, 3, f(2, x)) = f(2, x) + 1 = x + 2 + 1 = x + 3,\\
f(4, x) = h\,(x, 4, f(3, x)) = f(3, x) + 1 = x + 3 + 1 = x + 4,\\
\ldots\\
f(y, x) = h\,(x, y, f(y - 1, x)) = f(y - 1, x) + 1 = x + (y - 1) + 1 = x + y.
\end{gathered}$$

Получим двухместную функцию сложения $f(x, y) = x + y$, которая примитивно-рекурсивная, так как получена из простейших функций с помощью конечного числа применений оператора примитивной рекурсии.

Задача 4. Примените оператор примитивной рекурсии к простейшей функции $g(x) = 0$ и к примитивно-рекурсивной функции $h(x, y, z) = x + y$ по переменной y, постройте функцию $f(x, y) = R(g, h)$, записав ее в аналитической форме.
Решение. Примитивно-рекурсивное описание функции $f(x, y)$ по переменной y:

$$\begin{gathered}
f(0, x) = g(x) = 0,\\
f(1, x) = x + f(0, x) = x + 0 = x,
\end{gathered}$$

$$f(2, x) = x + f(1, x) = x + x = 2x,$$
$$f(3, x) = x + f(2, x) = x + 2x = 3x,$$
$$f(4, x) = x + f(3, x) = x + 3x = 4x,$$
$$\ldots$$
$$f(y, x) = x + f(y - 1, x) = x + (y - 1)x = xy.$$

Получили двухместную примитивно-рекурсивную функцию умножения $f(x, y) = xy$.

Задача 5. Применив оператор примитивной рекурсии к функциям $g(x) = 1$ и $h(x, y, z) = xy$ по переменной y, постройте функцию $f(x, y) = R(g, h)$, записав ее в аналитической форме.
Решение. Примитивно-рекурсивное описание функции $f(x, y)$ по переменной y:

$$f(0, x) = g(x) = 0,$$
$$f(1, x) = f(0, x) \cdot x = 1 \cdot x = x,$$
$$f(2, x) = f(1, x) \cdot x = x \cdot x = x^2,$$
$$f(3, x) = f(2, x) \cdot x = x^2 \cdot x = x^3,$$
$$\ldots$$
$$f(y, x) = f(y - 1, x) \cdot x = x^{y-1} \cdot x = x^y.$$

Получили двухместную примитивно-рекурсивную функцию $f(x, y) = x^y$.

Задача 6. Докажите, что одноместная функция $x!$ (где $0! = 1$) — примитивно-рекурсивная.
Решение. Пусть $g(x) = 1$; $h(x) = h(x) \cdot x'$,

$$f(0) = g = 1!,$$
$$f(1) = h(f(0)) \cdot 1' = 1 \cdot 1' = 1(1 + 1) = 1 \cdot 2 = 2!,$$
$$f(2) = h(f(1)) \cdot 2' = 1 \cdot 2(2 + 1) = 1 \cdot 2 \cdot 3 = 3!,$$
$$f(3) = h(f(2)) \cdot 3' = 1 \cdot 2 \cdot 3(3 + 1) = 1 \cdot 2 \cdot 3 \cdot 4 = 4!,$$
$$\ldots$$
$$f(x - 1) = h(f(x - 2)) \cdot (x - 1) = 1 \cdot 2 \cdot 3 \cdot 4 \cdot \ldots \cdot (x - 1)(x - 1 + 1) =$$
$$= 1 \cdot 2 \cdot 3 \cdot \ldots \cdot x = x!,$$

т. е. функция $x!$ — примитивно-рекурсивная.

Задача 7. Примените оператор минимизации к функции

$$f(x) = \begin{cases} 3x + 2, \text{ если } x \neq 2, \\ \text{не определено, если } x = 2. \end{cases}$$

Решение. Для каждого $x_0 \in N$ ищем минимальное решение уравнения $f(y) = x_0$.

Так как множеством значений функции $f(x)$ является множество

$$\{3n+2:\ n \neq 2\} = \{2, 5\} \cup \{11, 14, \ldots, 3n+2, \ldots\},$$

то уравнение $f(y) = x_0$ имеет решение лишь при $x_0 = 2, 5, 11, 14, \ldots$. Для всякого такого x_0 решение единственное (оно равно $\frac{x_0 - 2}{3}$).

Принимая во внимание, что функция $f(x)$ при $x = 2$ не определена, заключаем: найденные решения, превосходящие 2, т. е. 3, 4, …, не являются допустимыми в силу определения минимизации. Итак, функция $g(x) = \mu_x f(x)$ определена только при $x = 2$ и $x = 5$; $g(2) = 0$, $g(5) = 1$.

Задача 8. Докажите, что если функция $f(x_1, \ldots, x_n)$ примитивно-рекурсивна, то следующие функции примитивно-рекурсивные:

а) $f_1(x_1, x_2, \ldots, x_n) = f(x_2, x_1, \ldots, x_n)$, б) $f_2(x_1, x_2, \ldots, x_n) = f(x_2, \ldots, x_n, x_1)$,

в) $f_3(x_1, \ldots, x_n, x_{n+1}) = f(x_1, \ldots, x_n)$, г) $f_4(x_1, \ldots, x_{n-1}) = f(x_1, x_1, \ldots, x_{n-1})$.

Решение. Функции f_1, f_2, f_3, f_4 получаются суперпозициями из f и J_m^n.

$$\text{а) } f_1(x_1, x_2, x_3, \ldots, x_n) =$$
$$= f\left(J_2^n(x_1, \ldots, x_n), J_1^n(x_1, \ldots, x_n), J_3^n(x_1, \ldots, x_n), \ldots, J_n^n(x_1, \ldots, x_n)\right) =$$
$$= f(x_2, x_1, x_3, \ldots, x_n),$$

что и требовалось доказать.

$$\text{б) } f_2(x_1, x_2, \ldots, x_n) =$$
$$= f\left(J_2^n(x_1, \ldots, x_n), J_3^n(x_1, \ldots, x_n), J_n^n(x_1, \ldots, x_n), \ldots, J_1^n(x_1, \ldots, x_n)\right)$$

$$\text{в) } f_3(x_1, x_2, \ldots, x_{n+1}) = f\left(J_1^{n+1}(x_1, \ldots, x_{n+1}), \ldots, J_n^{n+1}(x_1, \ldots, x_{n+1})\right),$$

$$\text{г) } f_4(x_1, \ldots, x_{n-1}) = f\left(J_1^n(x_1, \ldots, x_{n-1}), \ldots, J_{n-1}^{n-1}(x_1, \ldots, x_{n-1})\right).$$

Задача 9. Пусть g^{n+1} — примитивно-рекурсивная функция. Доказать, что функция $f^{n+1}(x_1, \ldots, x_n, x_{n+1}) = \sum\limits_{i=0}^{x_{n+1}} g(x_1, \ldots, x_n, i)$ примитивно-рекурсивная.

Решение. $f^{n+1}(x_1, \ldots, x_n, 0) = g(x_1, \ldots, x_n, 0)$,

$$f^{n+1}(x_1, \ldots, x_n, y+1) = g(x_1, \ldots, x_n, y+1) + f(x_1, \ldots, x_n, y).$$

Задачи для самостоятельного решения

Задача 10. Найдите функции g и h в рекурсивной формуле для двухместной функции $f(x, y)$, если рекурсия проводится: а) по переменной x, б) по переменной y. Составьте примитивно-рекурсивное описание функции $f(x, y)$ и докажите, что функция $f(x, y)$ принадлежит классу примитивно-рекурсивных функций.

1) $f(x, y) = 2xy$; 2) $f(x, y) = y + 3x$; 3) $f(x, y) = xy + y + x$;

4) $f(x, y) = x + 2y + 1$; 5) $f(x, y) = x^2 + y$; 6) $f(x, y) = 3^{x+y}$; 7) $f(x, y) = (x+y)^2$;

8) $f(x, y) = x^2 + y^2$.

Задача 11. Найдите функции g и h в рекурсивной формуле для трехместной функции $f(x, y, z)$, если рекурсия проводится по переменной x:

1) $f(x, y, z) = xy^2z$;   2) $f(x, y, z) = x^2(y + z)$;

если рекурсия проводится по переменной y:

3) $f(x, y, z) = x^2y + y^2z$;

если рекурсия проводится по переменной z:

4) $f(x, y, z) = xy + z$;   5) $f(x, y, z) = 2^{x+y+z}$.

Задача 12. Найдите рекурсивные формулы для трехместной функции $f(x, y, z)$ и составьте примитивно-рекурсивное описание, если рекурсия проводится по переменной x:

1) $f(x, y, z) = xy^2z$; 2) $f(x, y, z) = x^2(y + z)$, если рекурсия проводится по переменной y;

3) $f(x, y, z) = x^2y + y^2z$, если рекурсия проводится по переменной z;

4) $f(x, y, z) = xy + z$; 5) $f(x, y, z) = 2^{x+y+z}$.

Задача 13. Применив оператор примитивной рекурсии к функциям g и h по переменной x, постройте функцию $f(x) = R(g, h)$ и запишите ее в аналитической форме:

а) $g = 0$; $h(x, y) = 2x + y$;

б) $g = 0$; $h(x, y) = x + 2y$;

Ответ: а) $f(x + 1) = x(x + 1)$; б) $f(x + 1) = 2^{x+1} - x - 2$.

III. Нормальный алгоритм Маркова

Наряду с рекурсивными функциями нормальный алгоритм Маркова получил известность в качестве одного из уточнений общего интуитивного представления об алгоритме. Исходными данными и возможными результатами применения нормального алгоритма являются конструктивные объекты, достаточно общего типа — слова, и это обстоятельство определяет его роль как алгоритма в алфавите в математике.

1. Основные определения

Определение. Алфавитом называется любая конечная система различных символов.

Определение. Буквами называются символы, составляющие алфавит.

Определение. Словом в алфавите называется любая конечная последовательность букв в этом алфавите.

Определение. Пустым словом называется слово, не содержащее ни одной буквы и обозначаемое символом $\wedge$.

Примеры.

1) $\{$а, ъ, ?, 7, $*\}$ — алфавит; а, ъ, ?, 7, $*$ — буквы.

2) $\{a, b, c\}$ — алфавит; ac, a, $abbca$, $bbbbb$, $bbacab$ — слова этого алфавита.

3) Алфавит исчисления высказываний состоит из букв A, B, Q, R, P и других, возможно с индексами, логических связок — $\wedge$, $\vee$, $\rightarrow$, а также вспомогательных символов (,).

Основная операция на словах — операция приписывания слова к слову: если дано слово, имеющее вид $a_0a_1...a_n$, и слово вида $b_0b_1...b_m$, то можно образовать новое слово $a_0a_1...a_nb_0b_1...b_m$, полученное приписыванием или соединением слов.

Если $\wedge$ — пустое слово, а a — слово, то $\wedge a = a\wedge = a$.

Определение. Два конкретных слова $a_0a_1...a_n$ и $b_0b_1...b_m$ и в алфавите A равны, т. е. $a_0a_1...a_n = b_0b_1...b_m$, если $n = m$ и $a_1 = b_1, ..., a_n = b_n$. Все пустые слова считаются равными.

Определение. Если $a_0a_1...a_n$ — слово, состоящее из n букв, где $a_1...a_n \in A$, то n называется длиной этого слова. Длиной пустого слова будет число 0.

Рассмотрим два слова: слово P и слово Q в некотором алфавите A. Если слово P является частью слова Q, то говорят, что слово P входит в слово Q.

В качестве примера можно привести два слова: $P = ac$; $Q = bbacab$. В общем случае слово P может входить в слово Q несколько раз. Так, слово bcb два раза входит в слово $abcbcbab$.

Определение. Ассоциативным исчислением называется совокупность всех слов в данном алфавите A вместе с конечной системой допустимых подставок, т. е. чтобы задать ассоциативное исчисление достаточно задать алфавит и систему подстановок.

Опишем процесс преобразования слов, позволяющий из заданного слова получить новые слова. Зададим в некотором алфавите A конечную систему допустимых подстановок, т. е. пар слов в этом алфавите: $P - Q$; $L - M$; ..., $S - T$.

Любую подстановку $L - M$ можно применять к некоторому слову R этого алфавита следующим способом: если в слове R имеется одно или несколько включений слова L, то любое из этих включений может быть заменено словом M и наоборот, если имеется включение слова M, то его можно заменить словом L.

К полученным с помощью допустимых подставок словам можно снова применить допустимые подстановки: так будут получены новые слова.

Определение. Два слова P_1 и P_2 в некотором ассоциативном исчислении называются смежными, если одно из них может быть преобразовано в другое при помощи однократного применения некоторой допустимой подстановки.

Определение. Последовательность слов $P_1, P_2, P_3, \ldots, Q$ называется дедуктивной цепочкой, ведущей от слова P к слову Q, если каждые из двух рядом стоящих слов этой цепочки являются смежными.

Определение. Два слова P и Q называются эквивалентными, если существует дедуктивная цепочка, ведущая от слова P к слову Q. Отношение эквивалентности обозначается $P \leftrightarrow Q$. Очевидно, что если $P \leftrightarrow Q$, то, поскольку допустимые подстановки можно применять в обе стороны, $Q \leftrightarrow P$.

Иногда рассматривается специальный вид ассоциативного исчисления, которое задается алфавитом и системой ориентированных подстановок вида $P \to Q$. Это означает, что подстановку разрешается проводить лишь слева направо, т. е. заменять вхождение слова P на слово Q, но не наоборот.

Ясно, что в таком ассоциативном исчислении из эквивалентности $P \leftrightarrow Q$ не следует, что $Q \leftrightarrow P$.

2. Проблема слов в ассоциативном исчислении

Проблема слов в ассоциативном исчислении заключается в следующем: для любых двух слов в данном исчислении требуется узнать, эквивалентны они или нет.

Поскольку в любом ассоциативном исчислении содержится бесчисленное множество различных слов, проблема эквивалентности представляет собой бесконечную серию однотипных задач, а решение мыслится в виде алгоритма, распознающего эквивалентность или неэквивалентность любой пары слов. Можно легко построить алгоритм, решающий так называемую *ограниченную проблему слов*: требуется установить, можно ли одно из заданных слов преобразовать в другое применением допустимых подстановок не более чем k раз, где k — произвольное, но фиксированное число.

Применительно к подстановкам это означает, что сначала надо построить все слова, смежные с одним из заданных слов, затем для каждого из полученных слов построить все слова, смежные с ним, и т. д., всего k раз. В итоге мы будем иметь список слов, которые можно получить из заданного слова с помощью применения допустимых подстановок не более k раз. Если второе заданное слово окажется в этом списке, то, следовательно, ответ на поставленный вопрос будет положительным, если его в списке нет, ответ будет отрицательным.

Однако успешное решение ограниченной проблемы слов не приближает нас к решению основной, «неограниченной» проблемы. Поскольку длина дедуктивной цепочки, ведущей от слова P к слову Q, может оказаться сколь угодно большой, то неизвестно, когда следует считать законченным процесс переработки.

Возникает проблема: для произвольного ассоциативного исчисления требуется построить алгоритм, который для любой пары слов в алфавите этого исчисления позволял бы за конечное число шагов выяснить, эквивалентны ли в данном исчислении слова, составляющие эту пару, т. е. построить алгоритм в некотором алфавите A.

3. Алгоритм в некотором алфавите A

Определение. Алгоритмом в алфавите A называется всякое общепонятное точное предписание, которое определяет потенциально возможный процесс над словами из A, допускающий любое слово в качестве исходного и последовательно определяющий новые слова в этом алфавите.

Уточнение понятия алгоритма в алфавите A связано с использованием аппарата подстановок, т. е. с построением ассоциативного исчисления.

Определение. Алгоритм в алфавите A задается в виде некоторой системы допустимых подстановок, дополненной общепонятным точным предписанием о том, в каком порядке и как нужно применять эти допустимые подстановки, и когда наступает остановка. Таким образом, схема подстановок вместе с указанием, как ими пользоваться, определяет алгоритм в алфавите A.

Определение. Два алгоритма в некотором алфавите называются эквивалентными, если области их применимости совпадают, и результаты переработки ими любого слова из их области применимости также совпадают.

4. Нормальный алгоритм Маркова

А. А. Марковым было дано точное математическое определение *нормального алгоритма*.

Задается алфавит A и фиксируется схема подстановок. Алгоритм предписывает, исходя из произвольного слова P в алфавите A, просмотреть формулы подстановок в том порядке, в каком они заданы в схеме, разыскивая формулу с левой частью, входящей в P. Если такой формулы не найдется, то процесс обрывается. В противном случае берется первая из таких формул и делается подстановка ее правой части вместо первого вхождения ее левой части в P, что дает новое слово P_1 в алфавите A. После выполнения первого шага приступают ко второму шагу, отличающемуся от первого только тем, что роль P играет P_1. Далее делают аналогичный третий шаг и т. д. до тех пор, пока не придется оборвать процесс. Оборваться же он может лишь двумя способами: во-первых, когда мы получим такое слово P_n, что ни одна из левых частей формул схемы подстановок не будет в него входить; во-вторых, когда при получении слова P_n нам придется применить последнюю формулу. В обоих случаях мы считаем, что наш алгоритм перерабатывает слово P в слово P_n.

Различные нормальные алгоритмы отличаются друг от друга лишь алфавитами и системами допустимых подстановок. Чтобы задать какой-либо нормальный алгоритм достаточно задать алфавит и систему подстановок.

Понятие алгоритма в некотором алфавите было уточнено Марковым следующим образом: всякий алгоритм в алфавите A эквивалентен нормальному алгоритму в этом же алфавите.

5. Нормально вычислимая функция

Определение. Пусть $A = \{a_1, ..., a_n\}$ — алфавит. Если P, Q — слова в алфавите A, то выражения $P \to Q$, $P \to \circ Q$ называются формулами подстановки в алфавите A. При этом формула $P \to Q$ называется простой, а формула $P \to \circ Q$ — заключительной.

Определение. Произвольная конечная последовательность формул называется схемой S нормального алгоритма U. Таким образом, схема нормального алгоритма имеет вид:

$$P_1 \to \circ Q_1,\ P_2 \to \circ Q_2,\ \ldots,\ P_n \to \circ Q_m.$$

Схема S определяет алгоритм U, перерабатывающий слова в алфавите A, т. е. вычисляющий словарную функцию на словах в алфавите A.

Нормальный алгоритм U со схемой S вычисляет словарную функцию

$$F : A^* \to A^*,$$

где A^* — множество слов в алфавите A, если для любых слов $P, Q \in A^*$ имеем $F_S(P) = Q$.

Таким образом, вычислительный процесс в алгоритме Маркова сводится к преобразованию слов в соответствии с заданной программой с помощью словарной функции.

Оказалось, что класс функций, вычислимых с помощью нормальных алгоритмов, совпадает с классом частично-рекурсивных функций.

Двенадцатое практическое занятие по теме «Нормальные алгоритмы»

Задача 1. Задано слово $abcbcbab$, примените к нему подстановку $ab - bcb$.
Решение. Замена каждого из двух вхождений bcb дает слова $aabcbab$, $abcabab$, а замена каждого из двух вхождений ab дает слова $bcbcbcbab$, $abcbcbbcb$.

Задача 2. Покажите, что подстановка $ab - bcb$ не применима к слову $bacb$.
Решение. Эта подстановка не применима к слову $bacb$, так как в это слово не входит ни слово ab, ни слово bcb.

Задача 3. Задано ассоциативное исчисление алфавитом $\{a, b, c, d, e\}$ и системой допустимых подстановок

1) ac – са, **2)** $ad - da$, **3)** $bc - cb$, **4)** $bd - db$, **5)** $abac - abacc$, **6)** $eca - ae$, **7)** $edb - be$.

Покажите, что 1) слова $abcde$ и $acbde$ в этом исчислении являются смежными; 2) слово $aaabb$ не имеет смежных слов; 3) слово $abcde$ эквивалентно слову $cadedb$.
Решение. 1) Слова $abcde$ и $acbde$ являются смежными, так как слово $abcde$ может быть преобразовано в слово $acbde$ с помощью применения подстановки $bc - cb$;

2) слово $aaabb$ не имеет смежных слов, так как к нему не применима ни одна подстановка;

3) слово $abcde$ эквивалентно к слову $cadedb$, так как существует дедуктивная цепочка смежных слов: $a\underbrace{bc}de$, $\underbrace{ac}bde$, $ca\underbrace{bd}e$, $cad\underbrace{be}$, $cadedb$. Здесь последовательно применимы подстановки 3, 1, 4, 5.

Задача 4. Пусть задан нормальный алгоритм Маркова: алфавит $A = \{1, +\}$ и система подстановок S: 1) $1+ \to +1$; 2) $+1 \to 1$; 3) $1 \to \circ 1$.

Стрелками общепринято обозначать тот факт, что задано не ассоциативное исчисление обычного вида, а нормальный алгоритм Маркова. Покажите, во что перерабатывает этот алгоритм слово $1111 + 11 + 111$.
Решение. Последовательно получаем слова:

$$111\underbrace{1+}11+111 \to 11\underbrace{1+}111+111 \to 1\underbrace{1+}1111+111 \to \underbrace{1+}11111+111 \to$$
$$\to 11111\underbrace{1+}111 \to +1111\underbrace{1+}1111 \to +111\underbrace{1+}11111 \to +11\underbrace{1+}111111 \to$$
$$\to +1\underbrace{1+}1111111 \to +\underbrace{1+}11111111 \to +\underbrace{+1}11111111 \to \underbrace{+1}11111111 \to$$
$$\to 111111111 \to \circ 111111111.$$

Процедура заканчивается применением заключительной подстановки $1 \to$ $\to \circ 1$, которая перерабатывает слово 111111111 в себя.

Задача 5. Пусть задан нормальный алгоритм Маркова: алфавит $A = \{1, +\}$; система подстановок S: 1) $+ \to \wedge$; 2) $1 \to \circ 1$ ($\wedge$ — пустое слово). Покажите, что этот алгоритм перерабатывает слово $11 + 111 + 1 + 11$ в слово 11111111.

Решение. $11\underbrace{+}111+1+11 \to 11111\underbrace{+}1+11 \to 111111\underbrace{+}11 \to 11111111 \to$
$\to \circ 11111111.$

Замечание. Нетрудно убедиться, что указанные в задачах 3 и 4 два нормальных алгоритма эквивалентны, и оба суммируют количество единиц, т. е. осуществляют сложение.

Задачи для самостоятельного решения

Задача 6. Покажите, что алгоритм с алфавитом $A = \{1, +\}$ и системой подстановок S: 1) $1+ \to +1$; 2) $++ \to +$; 3) $+ \to \Lambda$ эквивалентен нормальным алгоритмам в задачах 4 и 5.

Задача 7. Покажите, что нормальный алгоритм $A = \{1, *, \vee, ?\}$ и S: 1) $*11 \to \vee*1$; 2) $*1 \to \vee$; 3) $1\vee \to \vee1?$; 4) $?\vee \to \vee?$; 5) $\vee1 \to \vee$; 6) $\vee? \to ?$; 7) $? \to 1$; 8) $1 \to \circ1$ перерабатывает всякое слово вида $\underbrace{111\ldots11}_{m\,\text{раз}} * \underbrace{111\ldots11}_{n\,\text{раз}}$ в слово $\underbrace{111\ldots11}_{m\cdot n\,\text{раз}}$, т. е. осуществляет умножение.

Задача 8. Постройте нормальный алгоритм, вычисляющий функцию x^2, если $A\{0, 1, a, b, c, d, e\}$; система подстановок S: 1) $c1 \to 0ac$; 2) $a0 \to 0a$; 3) $ea \to ae$; 4) $0a \to ae0$; 5) $0c \to d$; 6) $ed \to d1$; 7) $ad \to d$; 8) $0d \to d$; 9) $bc \to \circ1$; 10) $bd \to \circ1$; 11) $1 \to bc$.

Указание: для изображения натурального числа m здесь использована запись $11\ldots1$, где число единиц равно $m+1$, т. е. для $m = 2$ используется слово 111, для $m = 4$ — 11111.

Решение.

$$\underbrace{1}11 \to b\underbrace{c1}1 \to b0a\underbrace{c1} \to b\underbrace{0a}0ac \to b0\underbrace{0a}ac \to b\underbrace{0a}e0ac \to$$

$$\to bae0e\underbrace{0a}c \to bae0\underbrace{ea}e0c \to bae\underbrace{0a}ee0c \to ba\underbrace{ea}e0ee0c \to$$

$$\to baaee0ec\underbrace{0c} \to baaee0c\underbrace{ed} \to baaee0\underbrace{ed}1 \to baaee\underbrace{0d}11 \to bae\underbrace{ed}11 \to$$

$$\to baa\underbrace{ed}111 \to ba\underbrace{ad}1111 \to b\underbrace{ad}1111 \to \underbrace{bd}1111 \to \circ11111.$$

Задача 9. Постройте нормальный алгоритм U над алфавитом A такой, что для любого слова α в алфавите A $U(\alpha) = \alpha\alpha$.

IV. Машины Тьюринга

Определение вычислимости, предложенное А. М. Тьюрингом, основано на анализе осуществления алгоритма человеком, в распоряжении которого имеется карандаш и бумага. Тьюринг рассматривает это как последовательность простых действий следующего вида:

(а) запись или стирание одного символа;

(б) перенесение внимания с одного участка бумаги на другой.

Тьюринг изобрел конечные машины, которые выполняют алгоритмы, представленные таким способом.

Из интуитивного понятия алгоритма ясно, что вычисления в соответствии с некоторым алгоритмом производятся чисто механически и их можно поручить машине.

1. Алгоритмы Тьюринга

Тьюрингскими алгоритмами являются алгоритмы, конструктивные объекты которых можно кодировать в виде слов в некотором алфавите.

Пусть заданы два конечных алфавита A и Q. A назовем внешним алфавитом; Q назовем внутренним алфавитом (или алфавитом состояний).

Внешний алфавит — это конечный алфавит символов, которые подаются на вход машины Тьюринга и выдаются на ее выходе.

Внутренний алфавит — это конечный алфавит символов, комбинации которых определяют внутреннее состояние машины Тьюринга. Элементы A называются внешними символами машины, элементы Q — внутренними состояниями или просто состояниями.

Предположим, что символы $\to$, R, L не входят ни в A, ни в Q.

Командой назовем слово одного из следующих трех видов: $qa \to rb$; $qa \to rbR$; $qa \to rbL$, где $q, r \in Q$; $a, b \in A$. Команды, подобно формулам языка, можно читать по-русски. Команда первого вида читается так: «находясь в состоянии q и наблюдая букву a, следует перейти в состояние r и напечатать букву b». Команда второго вида: «находясь в состоянии q и наблюдая букву b, следует перейти в состояние r, напечатать букву b и затем передвинуться вправо». Команда третьего вида читается так же, как и команда второго вида, за исключением концовки: «...и затем передвинуться влево».

Конечная последовательность команд называется *программой* машины.

Пусть теперь фиксированы алфавиты A и Q, а также две буквы $q_0, q_1 \in Q$ и одна буква $a_0 \in A$. Мы называем q_0 — начальным состоянием, q_1 — финальным или заключительным состоянием, букву a_0 называем бланком, или пустой клеткой, или пробелом.

Алфавит $Q \times A$ назовем алфавитом наблюдаемых букв: про букву $\langle q, a \rangle \in Q \times A$ говорим, что a наблюдается в состоянии q.

Машинным словом, или конфигурацией, называется слово в алфавите $A \cup (Q \times A)$, содержащее в точности одно вхождение наблюдаемой буквы. Таким образом, машинное слово всегда имеет вид: $\alpha \langle q, a \rangle \beta$, где $\alpha, \beta \in A^*$, A^* — множество слов алфавита A, и буква a наблюдается в состоянии q.

Машинное слово называется начальным, если оно имеет вид $\langle q_0, a \rangle \beta$. Здесь q_0 — начальное слово α в данном случае пустое. Машинное слово называется финальным или заключительным, если оно имеет вид $\alpha \langle q_0, a \rangle \beta$, т. е. наблюдаемая буква находится в заключительном состоянии.

Пусть K_1 и K_2 — конфигурации или машинные слова некоторой машины M и k — команда. Определим, что значит команда k переводит машинное слово K_1 в K_2; символически запишем это отношение так $k : K_1 \xrightarrow{M} K_2$.

Пусть K_1 имеет вид $\alpha \langle q, a \rangle \beta$. Если левая часть команды k не имеет вида qa, то $k : K_1 \xrightarrow{M} K_2$ автоматически считается ложным, т. е. команда k не применима к машинному слову K_1.

Пусть левая часть команды k имеет вид qa, тогда разберем три случая в зависимости от строения команды k:

1) k есть $qa \to rb$. В этом случае $k : K_1 \xrightarrow{M} K_2$, и K_2 имеет вид $\alpha \langle r, b \rangle \beta$;

2) k есть $qa \to rbR$. Здесь рассмотрим два подслучая:

a) слово β не пусто, $\beta = c\beta'$, тогда $k : K_1 \xrightarrow{M} K_2$, и K_2 имеет вид $\alpha b \langle r, c \rangle \beta'$;

b) слово β пусто, тогда $k : K_1 \xrightarrow{M} K_2$, и K_2 есть $\alpha b \langle r, a_0 \rangle$.

3) k есть $qa \to rbL$. Здесь рассматриваются два подслучая:

a) слово α не пусто, $a = a'c$, тогда $k : K_1 \xrightarrow{M} K_2$ и K_2 есть $\alpha' \langle r, c \rangle \beta$;

b) слово α пусто, тогда $k : K_1 \xrightarrow{M} K_2$ и K_2 есть $\langle r, a_0 \rangle b\beta$.

Легко видеть, что если $k : K_1 \xrightarrow{M} K_2$, то машинное слово K_2 определяется однозначно по команде k и машинному слову K_1. Перевести K_1 в K_2 командой k — это значит «выполнить команду k, проделать то, что она „требует“ ».

После выполнения команды наблюдаемой может стать уже другая буква, левее или правее исходной.

2. Формализация машины Тьюринга

Определение. Машиной Тьюринга называется набор $M = \langle A, Q, \Pi, q_0, q_1, a_0 \rangle$, где A — внешний алфавит, Q — внутренний алфавит, Π — программа машины, q_0 — начальное состояние, q_1 — заключительное состояние, a_0 — бланк или пробел.

Мы говорим, что машина M переводит машинное слово K_1 в K_2 и имеем $M : K_1 \xrightarrow{M} K_2$, если

1) существует команда k_i программы П машины M такая, что

$$k_i : K_1 \xrightarrow{M} K_2;$$

2) для всех $j < i$ не существует машинного слова K_3 такого, что $k_i : K_1 \xrightarrow{M} K_3$.

Таким образом, если $M : K_1 \xrightarrow{M} K_2$, то машинное слово K_2 определяется однозначно по M и K_1. Конечно, вполне может оказаться, что некоторое машинное слово K_1 не переводится машиной M ни в какое машинное слово K_2.

Протокол вычислений машины Тьюринга M есть конечная последовательность машинных слов $K_0, K_1, \ldots, K_n$ такая, что:

1) K_0 — начальное машинное слово;

2) $M : K_i \xrightarrow{M} K_{i+1}$;

3) если машинное слово K_i входит в протокол $M : K_i \xrightarrow{M} K_{i+1}$, причем K_i — не заключительное машинное слово, то K_{i+1} также входит в протокол вычислений. Т. е. протокол не оканчивается словом K_i, если вычисления можно продолжить дальше;

4) протокол вычислений может содержать не более одного заключительного слова, и если протокол действительно содержит заключительное машинное слово, то этот протокол конечен и заключительным машинным словом является последний его член.

Машину Тьюринга можно представить себе как определенный тип вычислительной машины, обрабатывающую слова в алфавитах и порождающую функцию, которая перерабатывает некоторые слова в алфавите A в слова алфавита A. То есть если $\alpha, \beta \in A^*$, то $M(\alpha) = \beta$.

Заметим, что функция M может быть определена не на всех словах алфавита A. Машина M определена на слове α, и пишем $M(\alpha)$, если M определена на начальном слове.

3. Тезис Черча–Тьюринга

Сформулируем тезис Черча, который теперь звучит так: всякий словарный, дискретный, массовый, детерминированный, замкнутый алгоритм может быть задан в виде машины Тьюринга. Другими словами, для всякой функции, которую можно каким-либо способом вычислить, можно построить вычисляющую ее машину Тьюринга. Это утверждение, высказанное впервые Тьюрингом, называется тезисом Черча–Тьюринга.

В заключение сформулируем следующую теорему. Класс функций, вычислимых по Маркову, совпадает с классом функций, вычислимых по Тьюрингу, и, следовательно, совпадает с классом частично-рекурсивных функций.

Тринадцатое практическое занятие по теме «Машина Тьюринга»

Задача 1. Покажите, что машина Тьюринга со следующей программой команд:

$$q_1 0 \to q_2 0 R; \quad q_1 1 \to q_0 1;$$
$$q_2 0 \to q_2 1; \quad q_2 1 \to q_2 1 R$$

вычисляет функцию $f(x) = x + 1$.

Задача 2. Постройте машину Тьюринга, которая правильно вычисляет функцию $0(x) = 0$.

Решение.

$$q_1 0 \to q_2 0 R,$$
$$q_2 0 \to q_3 0 L, \quad q_2 \to q_2 1 R,$$
$$q_3 0 \to q_0 0, \quad q_3 1 \to q_3 0 L.$$

Задача 3. Пусть $f(x)$ и $g(x)$ — правильно вычислимы по Тьюрингу. Покажите, что функция $h(x) = f(g(x))$ — правильно вычислима по Тьюрингу.

Решение. Пусть F и G — машины, правильно вычисляющие f и g соответственно. Тогда $H = G \circ F$ правильно вычисляет h.

Задача 4. Выясните, применима ли машина Тьюринга T, задаваемая программой П, к слову P. Если применима, то выпишите результат применения машины T к слову P. Считается, что q_1 — начальное состояние.

$$\text{Программа П: } \begin{cases} q_1 0 q_2 1 R \\ q_1 1 q_1 0 L \\ q_2 0 q_3 1 R \\ q_2 1 q_3 0 L \\ q_3 0 q_1 0 R \end{cases}$$

а) $P = 10^3 1$; б) $P = [10]^2 1$.

Решение. а) Исходя из конфигурации $q_1 10^3 1$, получаем последовательно такие конфигурации: $q_1 0^5 1$, $1 q_2 0^4 1$, $1^2 q_3 0^3 1$, $1^2 0 q_1 0^2 1$, $1^2 01 q_2 01$, $1^2 01^2 q_3 1$. Так как команды вида $q_3 1 q_1 \alpha D$ входят в программу П, то последняя конфигурация (т. е.

$1^2 01^2 q_3 1)$ — заключительная. Следовательно, машина T к слову $P = 10^2 1$ применима и $T(P) = 1^2 01^3$.

б) Выписывая конфигурации, имеем $q_1[10]^2 1$, $q_1 0^3 101$, $1q_2 0^2 101$, $1^2 q_3 [01]^2$, $1^2 0 q_1 101$, $1^2 q_1 0^3 1$, $1^3 q_2 0^2 1$, $1^4 q_3 01$, $1^4 0 q_1 1$, $1^4 q_1 0$, $1^5 q_2 0$, $1^6 q_3 0$, $1^6 0 q_1 0$, $1^6 01 q_2 0$, $1^6 01^2 q_3 0$, $1^6 01^2 0 q_1 0$, $1^6 01^2 01 q_2 0$, $1^6 01^2 01^2 q_3 0$ и т. д. Ясно, что этот процесс продолжается неограниченно. Значит, машина T к слову $P = 10^2 1$ не применима.

Задача 5. Выясните, применима ли машина Тьюринга T, задаваемая программой П, к слову T. Если применима, то выпишите результат применения машины T к слову P. q_1 — начальное состояние, q_0 — заключительное состояние.

1.

$$\text{П}: \begin{cases} q_1 0 q_1 0 R \\ q_1 1 q_2 0 R \\ q_2 1 q_2 0 R \\ q_2 0 q_0 1 S \end{cases}$$

а) $P = 1^3 01$; б) $P = 1^2 0^2 1$; в) $P = 1^6$.

Ответ: а) $T(P) = 1^3 01$; б) $T(P) = 1$; в) к слову 1^6 не применима.

2.

$$\text{П}: \begin{cases} q_1 0 q_2 1 L \\ q_1 1 q_2 1 R \\ q_2 1 q_1 1 R \end{cases}$$

а) $P = 1^2 0^2 1$; б) $P = 1^6$; в) $P = 1^2 01^3$.

Ответ: а) $T(P) = 1^3 01$; б) $T(P) = 1^7$.

3.

$$\text{П}: \begin{cases} q_1 0 q_2 1 R \\ q_1 1 q_2 0 L \\ q_2 0 q_3 1 R \\ q_2 1 q_3 0 R \\ q_3 1 q_1 1 R \end{cases}$$

а) $P = 1^3 01^2$; б) $P = 1^5$; в) $P = 1^2 [01]^2$.

Ответ: а) $T(P) = 10^2 1^2 01^3$; б) $T(P) = 10^4 1^3$; в) $T(P) = 101^2 0^2 1$.

4.

$$\text{П}: \begin{cases} q_0 0 q_1 1 R \\ q_1 1 q_2 0 R \\ q_2 0 q_1 1 R \\ q_2 1 q_3 1 L \\ q_3 0 q_1 1 L \end{cases}$$

а) $P = 10^3 1$; б) $P = 10^2 1^2$; в) $P = 10^3 1$.

Ответ: К данным словам не применима.

Задача 6. Постройте в алфавите $\{0; 1\}$ машину Тьюринга, которая применима к словам вида $1^{2m+1} 01^{2n+1}$ ($m \geqslant 0$ и $n \geqslant 0$) и $1^{2m} 01^{2n}$ ($m \geqslant 1$ и $n \geqslant 1$), но не применима к словам вида $1^{2m} 01^{2n}$ и $1^{2n-1} 01^{2m}$ ($m \geqslant 1$ и $n \geqslant 1$).

Решение. Подходящая машина задается программой

$$q_1 1 q_2 1 R, \quad q_3 1 q_3 1 R,$$

$$q_2 1 q_1 1 R, \quad q_4 1 q_3 1 R,$$
$$q_1 0 q_3 0 R, \quad q_4 0 q_4 0 S,$$
$$q_2 0 q_4 0 R.$$

Задача 7. Постройте машину Тьюринга для вычисления функции $x + y$.
Решение.

$$q_1 0 \to q_2 0 R, \quad q_2 1 \to q_2 1 R,$$
$$q_2 0 \to q_1 1 R, \quad q_3 1 \to q_3 1 R,$$
$$q_3 0 \to q_4 0 L, \quad q_4 1 \to q_5 0 L,$$
$$q_5 0 \to q_0 0, \quad q_5 1 \to q_5 1 L.$$

Контрольные вопросы

1. Какая функция называется вычислимой?
2. Что называется алгоритмом?
3. Какие объекты называются конструктивными?
4. Сформулируйте характерные свойства алгоритмов.
5. Приведите пример вычислимого алгоритма: вычисление числа π.
6. Перечислите основные варианты математического определения алгоритма.
7. Какие функции называются простейшими?
8. Дайте определение операций суперпозиции, примитивной рекурсии и минимизации.
9. Какая функция называется примитивно-рекурсивной, частично-рекурсивной?
10. Дайте определение примитивно-рекурсивных предикатов. Какая функция называется характеристической для предиката P?
11. Сформулируйте тезис Черча.
12. Что называется алфавитом, буквами алфавита, словом в алфавите A?
13. Что называется ассоциативным исчислением в данном алфавите A?
14. Сформулируйте понятие алгоритма в алфавите A?
15. Дайте математическое определение нормального алгоритма Маркова.
16. Какая функция называется нормально вычислимой?
17. Что собой представляет алгоритмическая система Маркова?
18. Сформулируйте понятие машины Тьюринга.
19. Дайте математическое определение алгоритма Тьюринга. Что собой представляет алгоритмическая система Тьюринга?
20. Какие команды выполняются в алгоритмической системе Тьюринга?
21. Дайте определение функции вычислимой по Тьюрингу, определение рекурсивного предиката по Тьюрингу.
22. Сформулируйте тезис Черча–Тьюринга.

Часть 9

Задачи для контрольных и самостоятельных работ

I. Функции алгебры логики. Многочлены Жегалкина

Для заданной булевой функции трех переменных

а) Постройте таблицу истинности, найти двоичную форму булевой функции и привести функцию к СДНФ и СКНФ,

б) Найдите двумя способами многочлен Жегалкина и ответить на вопрос, является ли данная булева функция линейной,

в) С помощью эквивалентных преобразований приведите функцию к ДНФ, КНФ. СДНФ, СКНФ.

1. $(x \vee \bar{y}) \to (\bar{z} \oplus \bar{x})$; **2.** $\overline{(x \vee \bar{y})} \to (z \oplus \bar{x})$; **3.** $(\bar{x} \vee \bar{y}) \to \overline{(z \oplus x)}$;
4. $(x \vee \bar{y}) \to \overline{(z \leftrightarrow \bar{x})}$; **5.** $\overline{(x \vee \bar{y}) \to (z \leftrightarrow \bar{x})}$; **6.** $\overline{(x \mid \bar{y}) \oplus (z \to \bar{x})}$;
7. $\overline{(z \to x) \leftrightarrow (y \mid x)}$; **8.** $(x \mid \bar{y}) \oplus (\bar{z} \to x)$; **9.** $(\bar{z} \to x) \leftrightarrow (\bar{x} \mid y)$;
10. $(z \to x) \oplus (x \mid \bar{y})$; **11.** $((x \downarrow y) \to z) \oplus y$; **12.** $\left(\overline{(x \leftrightarrow \bar{y})} \to \bar{z}\right) \mid y$;
13. $\overline{(x \vee \bar{y})} \to (\bar{z} \leftrightarrow y)$; **14.** $\overline{((x \downarrow y) \to \bar{z}) \leftrightarrow y}$; **15.** $\overline{(x \downarrow y)} \to (z \leftrightarrow \bar{y})$;
16. $\overline{\left((x \leftrightarrow y) \mid \bar{z}\right) \oplus y}$; **17.** $(\bar{x} \vee y) \to \overline{(\bar{z} \leftrightarrow x)}$; **18.** $\overline{((x \downarrow y) \to z) \leftrightarrow x}$;
19. $\overline{(x \mid y) \oplus (\bar{z} \to y)}$; **20.** $\overline{(x \vee y) \to (\bar{z} \leftrightarrow y)}$; **21.** $\left(\overline{(x \downarrow y)} \to \bar{z}\right) \oplus y$;
22. $\overline{((x \downarrow y) \to \bar{z}) \leftrightarrow y}$; **23.** $\left(\overline{(x \downarrow y)} \to \bar{z}\right) \leftrightarrow y$; **24.** $\overline{((x \downarrow y) \to \bar{z}) \oplus y}$;
25. $\overline{\left((x \mid y) \to z\right)} \oplus y$.

II. Множества и основные операции над ними

Докажите тождества двумя способами:

а) используя определения равенства множеств и операций над множествами;

б) с помощью алгебры логики.

1. $A \backslash (B \cup C) = (A \backslash B) \cap (A \backslash C)$; **2.** $A \backslash (B \cap C) = (A \backslash B) \cup (A \backslash C)$;
3. $A \cap (B \backslash C) = (A \cap B) \backslash (A \cap C)$; **4.** $(A \backslash B) \backslash C = (A \backslash C) \backslash (B \backslash C)$;
5. $(A \cup B) \backslash C = (A \backslash C) \cup (B \backslash C)$; **6.** $A \backslash (B \backslash C) = (A \backslash B) \cup (A \cap C)$;
7. $A \backslash (B \cup C) = (A \backslash B) \backslash C$; **8.** $A \backslash (B \cup C) = (A \backslash B) \cap (A \backslash C)$;
9. $A \cup (B \backslash C) = (A \cup B) \backslash (C \backslash A)$; **10.** $(A \backslash B) \cup (A \backslash C) = A \backslash (B \cap C)$;
11. $(A \cap B) \backslash (A \cap C) = A \cap (B \backslash C)$; **12.** $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$;
13. $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$; **14.** $A \cap (B \backslash C) = (A \cap B) \backslash C$;

15. $A\backslash(B\cup C)=A\cap\overline{(B\backslash C)}$; **16.** $A\cap(\overline{B}\cap\overline{C})=(A\backslash B)\cap(A\backslash C)$;
17. $(A\backslash\overline{B})\cap(A\cap C)=A\backslash(B\cup C)$; **18.** $(A\backslash C)\backslash(B\backslash C)=(A\backslash C)\backslash B$; **19.** $A\cap B=\overline{(\overline{A}\cup\overline{B})}$;
20. $(\overline{A}\cup B)=\overline{(A\cap\overline{B})}$; **21.** $(A\cap B)\backslash C=(A\backslash C)\cap(B\backslash C)$; **22.** $(A\backslash B)\backslash C=A\backslash(B\cup C)$;
23. $(A\cap B)\cup(A\cap\overline{B})=(A\cup B)\cap(A\cup\overline{B})$; **24.** $A\cap B=(\overline{A}\cup B)\cap A$;
25. $\big((A\backslash B)\cup(B\backslash A)\big)\cup(A\cap B)=A\cup B$.

III. Комбинаторные формулы

Решите уравнение или систему уравнений.

1. $C_{x+8}^{x+3}=5A_{x+6}^{3}$; **2.** $C_{4x+9}^{4(x+1)}=5A_{4x+7}^{3}$; **3.** $C_{x+1}^{x-4}=\dfrac{7}{15}A_{x+1}^{3}$; **4.** $C_{2x+3}^{2(x-1)}=4A_{2(x+1)}^{3}$;

5. $12C_{x+3}^{x-1}=55A_{x+1}^{2}$; **6.** $30C_{x-3}^{x-9}=19A_{x-4}^{4}$; **7.** $\dfrac{A_{x+2}^{n+2}\cdot P_{x-n}}{P_x}=110$;

8. $\dfrac{P_{x+2}}{A_x^n\cdot P_{x-n}}=132$; **9.** $\dfrac{A_x^4\cdot P_{x-4}}{P_{x-2}}=42$; **10.** $\dfrac{A_{x+1}^{n+1}\cdot P_{x-n}}{P_{x-1}}=90$;

11. $12C_x^1+C_{x+4}^2=162$; **12.** $C_{x+1}^5=\dfrac{3A_x^3}{6}$; **13.** $\dfrac{A_x^5+A_x^3}{A_x^3}=43$; **14.** $\dfrac{A_x^7+A_x^5}{A_x^5}=89$;

15. $C_x^3+C_x^2=15(x-1)$; **16.** $C_x^4=\dfrac{15A_x^2}{4}$; **17.** $C_{x-3}^2=21$; **18.** $C_{x+1}^5=\dfrac{3A_x^3}{8}$;

19. $\begin{cases}A_x^y=10A_x^{y-1},\\ C_x^y=\dfrac{5}{3}C_x^{y-1};\end{cases}$ **20.** $\begin{cases}C_x^{y+1}=2{,}5x,\\ C_{x-1}^y=10;\end{cases}$ **21.** $\begin{cases}C_x^y=C_x^{y+2},\\ C_x^2=153;\end{cases}$

22. $\begin{cases}A_x^{n-3}=\dfrac{1}{8}A_x^{n-2},\\ C_x^{n-3}=\dfrac{5}{8}C_x^{n-2};\end{cases}$ **23.** $\begin{cases}A_{2n}^{3x}=8A_{2n}^{3x-1},\\ C_{2n}^{3x}=\dfrac{8}{9}C_{2n}^{3x-1};\end{cases}$ **24.** $\begin{cases}A_{2x}^{n-2}=8A_{2x}^{n-3},\\ C_{2x}^{n-2}=\dfrac{8}{3}C_{2x}^{n-3};\end{cases}$

25. $\begin{cases}A_{5x}^{n-3}=\dfrac{1}{7}A_{5x}^{n-2},\\ C_{5x}^{n-2}=\dfrac{7}{4}C_{5x}^{n-3}.\end{cases}$

IV. Бином Ньютона

1. Найдите четвертый член разложения $(a+3)^7$.
2. Найдите девятый член разложения $\left(a+\sqrt{b}\right)^{12}$.
3. Найдите шестой член разложения $(a^2+b^3)^{13}$.
4. Найдите средний член разложения $(\sqrt{a}-\sqrt{b})^8$.
5. Найдите средний член разложения $(x\sqrt{x}-1)^{14}$.
6. Найдите два средних члена разложения $(\sqrt{a}-\sqrt[3]{b})^{13}$.
7. Найдите два средних члена разложения $(\sqrt[3]{a}-2x\sqrt{x})^{19}$.
8. Найдите пятый член разложения

$$\left(\sqrt[4]{2\sqrt{\frac{2}{11}}}-\sqrt{\frac{1}{3}}\right)^{12}.$$

9. Найдите член разложения $(x+y)^9$, содержащий x^7.

10. Найдите член разложения $(\sqrt{a}+b)^9$, содержащий a^3.

11. Найдите член разложения $(\sqrt{a}+\sqrt[4]{a})^{20}$, содержащий a^7.

12. Найдите член разложения

$$\left(\sqrt[3]{x^2}-\sqrt[5]{x^2}\right)^{12},$$

содержащий $x^{\frac{22}{3}}$.

13. Найдите член разложения

$$\left(\frac{\sqrt{x}}{b}+\frac{b}{\sqrt[3]{x}}\right)^{18},$$

содержащий x^4.

14. Найдите член разложения

$$\left(\frac{1}{\sqrt[3]{a^2}}+\sqrt[4]{a^3}\right)^{17},$$

не содержащий q.

15. Найдите член разложения

$$\left(\frac{\sqrt[3]{x}}{b}+\frac{b}{\sqrt[4]{x}}\right)^{18},$$

содержащий x^{-1}.

16. Найдите член разложения

$$\left(\sqrt[3]{a}-\frac{1}{\sqrt{a}}\right)^{15},$$

не содержащий a.

17. Найдите член разложения

$$\left(\sqrt[9]{\frac{1}{z^8}}+\sqrt[3]{z^2}\right)^{7},$$

не содержащий z.

18. Найдите член разложения

$$\left(z\sqrt[3]{z^{-1}}+\frac{1}{\sqrt[7]{z^2}}\right)^{10},$$

не содержащий z.

19. Найдите член разложения

$$\left(\frac{2x}{y}-\frac{\sqrt{y}}{2x}\right)^{12},$$

не содержащий x.

20. Найдите член разложения $\left(\sqrt[-3]{c^2}+\sqrt[5]{c^3}\right)^{20}$, не содержащий c.

21. Вычислите коэффициент при x^5 в разложении $(1+x)^7$.

22. Вычислите коэффициент при x^{17} в разложении $(1+x^5)^7$.

23. Определите x из условия, что пятый член разложения $\left(\sqrt{x}+\frac{1}{x}\right)^6$ равен $\frac{5}{9}$.

24. Определите x из условия, что третий член разложения $\left(x+x^{\lg x}\right)^5$ равен 1 000 000.

25. Определите z из условия, что разность между пятым и третьим членами разложения $(z+\sqrt{5})^6$ равна 300.

V. Предикаты

Дано множество $M=\{a,b\}$. Предикат $P(x,y)$, где x и $y\in M$ задан следующей таблицей:

x	y	$P(x,y)$
a	a	0
a	b	1
b	a	1
b	b	1

Определить значение следующих высказываний:

1. $(\forall x)P(x,y)$
2. $(\exists x)P(x,a)$
3. $(\forall y)P(a,y)$
4. $(\exists x)P(a,y)$
5. $(\forall x)(\forall y)P(x,y)$
6. $(\forall y)(\forall x)P(x,y)$
7. $(\exists x)(\exists y)P(x,a)$
8. $(\exists y)(\exists x)P(x,a)$
9. $(\exists x)(\forall y)P(x,y)$
10. $(\forall x)(\exists y)P(x,y)$

Пусть $S(x,y,z)$ и $\Pi(x,y,z)$ — соответственно предикаты сложения (z является суммой x и y) и умножения (z является произведением x и y), рассматриваемые на множестве Z всех целых чисел и на множестве $N_0=N\cup\{0\}$ целых неотрицательных чисел.

Какой смысл имеют следующие формулы и на каком множестве Z или N_0 они истинны?

11. $(\exists y)(\forall x)S(x,y,x)$
12. $(\exists y)(\forall x)\Pi(x,y,x)$
13. $(\forall z)(\forall x)(\exists y)S(x,y,z)$
14. $(\forall z)(\forall x)(\exists y)\Pi(x,y,z)$
15. $(\forall y)(\exists x)S(x,y,0)$
16. $(\exists y)(\forall x)\Pi(x,y,-x)$
17. $(\forall y)(\exists x)\Pi(x,y,0)$
18. $(\forall y)(\exists x)S(x,y,-5)$
19. $(\exists x)(\exists y)\Pi(x,y,-5)$
20. $(\exists x)(\exists y)S(x,y,-12)$

Являются ли тождественно-истинными следующие формулы.

21. $(\exists x)P(x)\to(\forall x)P(x)$
22. $(\forall x)P(x)\to(\exists x)P(x)$
23. $(\exists x)P(x)\vee(\exists x)Q(x)\leftrightarrow(\exists x)(P(x)\vee Q(x))$
24. $(\exists x)(\forall y)Q(x,y)\to(\forall y)(\exists x)Q(x,y)$
25. $P(x)\to(\forall y)P(y)$

VI. Графы

Для графа, представленного следующей матрицей инцидентности, определите матрицу смежности графа и изобразите ее графически.

1. $\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$; 2. $\begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$;

3. $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$; 4. $\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$;

5. $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$.

Для графа, представленного следующей матрицей смежности, определите матрицу инцидентности графа и изобразите его графически.

6. $\begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ -3 & 0 & 0 & 0 & 0 \end{bmatrix}$; 7. $\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$; 8. $\begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$;

9. $\begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}$; 10. $\begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}$.

Представьте в виде ориентированного графа отношение $\rho = \langle X, R\rangle$.

11. $X = \{2, 4, 6, 8\}$; $R = \{\langle x, y\rangle : x < y\}$;
12. $X = \{1, 3, 5\}$; $R = \{\langle x, y\rangle : x \leqslant y\}$;
13. $X = P(\{a, b, c\})$; $R = \{\langle A, B\rangle : A \subset B\}$;
14. $X = P(\{a, b\})$; $R = \{\langle A, B\rangle : A \subseteq B\}$;
15. $X = \{2, 4, 8, 10\}$, $R = \{\langle x, y\rangle : x \geqslant y\}$;
16. $X = \{2, 4, 16, 22\}$, $R = \{\langle x, y\rangle : y \vdots x$ (x является делителем y)$\}$;
17. $X = \{2, 4, 16, 22\}$, $R = \{\langle x, y\rangle : x + y \vdots 6$ ($x + y$ делится на 6)$\}$;
18. $X = \{2, 4, 16, 22\}$, $R = \{\langle x, y\rangle : x/y$ четно$\}$

19. $X = \{2, 4, 8, 10\}$, $R = \{\langle x, y\rangle : x - y \vdots 3$ ($x - y$ делится на 3)$\}$;
20. $X = P(\{a, b, c\})$, $R = \{\langle A, B\rangle : A \cap B \neq \varnothing\}$.
Найдите матрицы смежности и инцидентности графов $G_1 \cap G_2$ и $G_1 \cup G_2$, если
21. См. рис. 9.1.

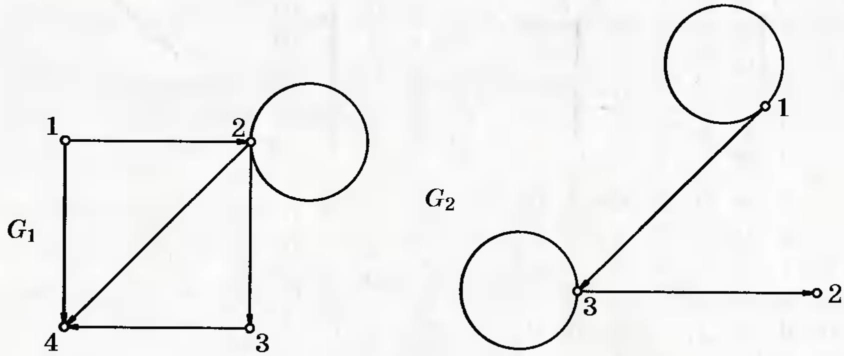


Рис. 9.1

22. См. рис. 9.2.

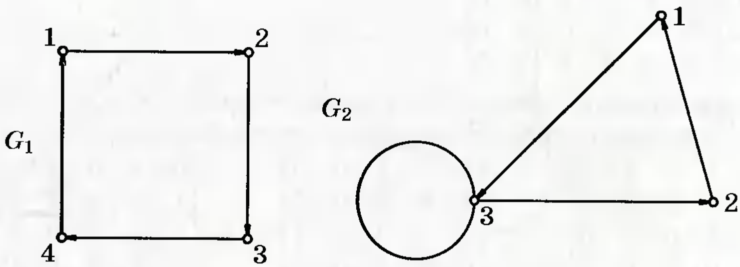


Рис. 9.2

23. См. рис. 9.3.

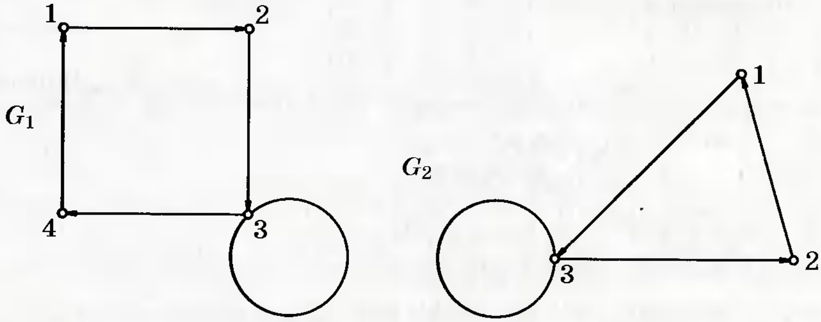


Рис. 9.3

24. См. рис. 9.4.

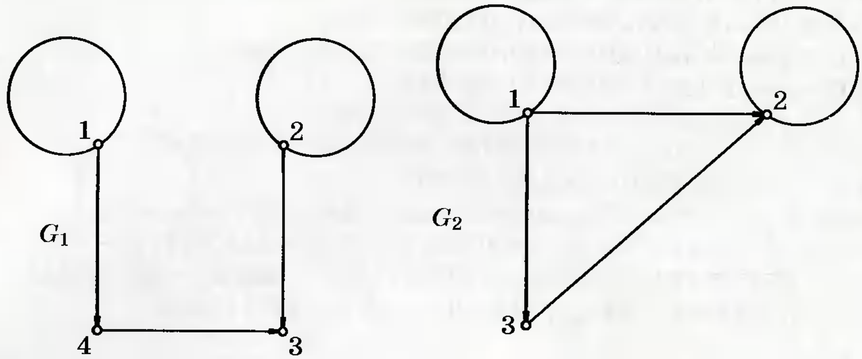


Рис. 9.4

25. См. рис. 9.5.

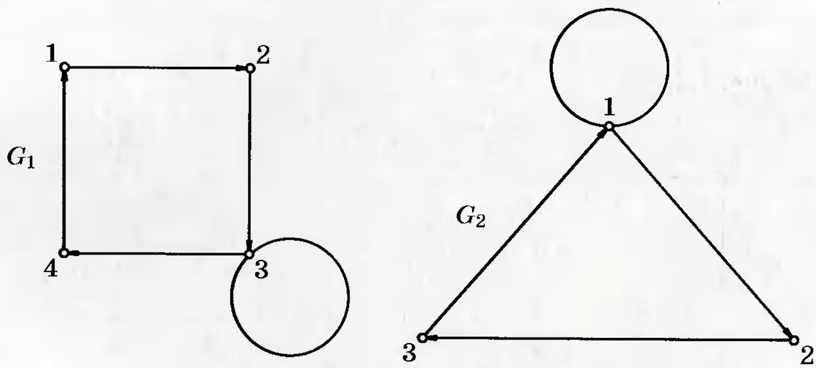


Рис. 9.5

VII. Кодирование

Выясните, является ли код C с кодирующим алфавитом $\{0, 1, 2\}$ однозначно декодируемым:

1. $C = \{01, 201, 112, 122, 0112\}$;

2. $C = \{001, 021, 102, 201, 001121, 01012101\}$;

3. $C = \{0, 01, 0010001001\}$;

4. $C = \{20, 01202, 22, 2001, 2012010, 10201121, 1112\}$;

5. $C = \{01, 011, 100, 2100, 101210, 001210\}$;

6. $C = \{01, 011, 100, 2100, 10110, 00112\}$;

7. $C = \{01, 12, 021, 0102, 10112\}$;

8. $C = \{01, 12, 012, 111, 0102, 10112, 01112\}$;

9. $C = \{01, 12, 012, 0102, 020112\}$;
10. $C = \{01, 10, 210, 121, 0210, 0112\}$;
11. $C = \{01, 10, 210, 201, 0210, 011022, 2221\}$;
12. $C = \{01, 10, 210, 201, 0210, 011022, 221\}$;
13. $C = \{01, 10, 210, 201, 0210, 011022\}$;
14. $C = \{01, 12, 011, 01210, 20120, 2011220\}$;
15. $C = \{01, 12, 011, 01210, 201120, 2011220\}$;
16. $C = \{000, 0100, 10, 1001, 0010010\}$.

Постройте по методу Хэмминга кодовое слово для сообщения:

17. $\alpha = 10101011$; **18.** $\alpha = 100010011$; **19.** $\alpha = 111001111$;
20. $\alpha = 01110111011$; **21.** $\alpha = 110101110010$; **22.** $\alpha = 0011010111$;
23. $\alpha = 11011001$; **24.** $\alpha = 1110001$; **25.** $\alpha = 110110100$.

VIII. Автоматы

Для автомата, заданного таблицей, постройте диаграмму Мура. Задайте этот автомат системой булевых функций.

1.

$x \backslash q$	0	1	2	3
0	1;1	3;0	2;0	2;0
1	2;1	2;0	3;0	3;0

2.

$x \backslash q$	0	1	2	3
0	1;0	2;0	2;1	3;0
1	3;0	3;1	2;1	1;0

3.

$x \backslash q$	0	1	2	3
0	0;1	1;1	2;1	3;1
1	0;0	0;1	3;1	2;1

4.

$x \backslash q$	1	2	3
0	2;0	2;1	3;1
1	1;1	3;0	3;1
2	1;1	2;1	1;0

5.

$x \backslash q$	0	1
0	0;0	0;1
1	0;1	1;0
2	0;1	1;0
3	1;0	1;1

6.

$x \backslash q$	0	1
0	0;0	1;1
1	1;1	1;1
2	1;1	1;1
3	0;0	1;1

7.

$x \backslash q$	1	2	3
0	1;0	2;1	3;2
1	2;1	2;1	3;2
2	3;2	2;1	3;2
3	1;0	2;1	3;2

8.

$x \backslash q$	0	1	2	3
0	1;0	3;1	2;0	1;0
1	3;0	1;1	0;1	3;1

9.

$x \backslash q$	0	1	2	3
0	0;0	1;1	3;1	2;0
1	2;0	0;1	3;1	1;0

10.

$x \backslash q$	0	1	2	3
0	2;0	0;0	3;1	1;0
1	1;0	0;0	0;0	3;0

11.

$x \backslash q$	0	1	2	3
0	3;0	2;0	1;1	0;1
1	0;1	1;1	2;0	3;0

12.

$x \backslash q$	0	1	2	3
0	2;1	2;1	2;1	2;1
1	1;1	3;1	0;0	1;0

Для автомата, заданного системой булевых функций, постройте диаграмму Мура и соответствующую таблицу.

13. $\begin{cases} z(t+1) = \overline{x_1}(t) \wedge \overline{z}(t) \vee \overline{x_2}(t) \wedge z(t), \\ y(t) = \overline{x_2}(t) \wedge z(t) \vee x_1(t) \wedge x_2(t); \end{cases}$

14. $\begin{cases} z(t+1) = x_1(t) \wedge x_2(t) \vee x_1(t) \wedge z(t) \vee x_2(t) \wedge z(t), \\ y(t) = x_1(t) \oplus x_2(t) \oplus z(t); \end{cases}$

15. $\begin{cases} z(t+1) = \overline{x_1}(t) \wedge x_2(t) \vee x_2(t) \wedge \overline{x_2}(t) \vee z(t), \\ y(t) = \overline{x_1}(t) \wedge x_2(t) \vee x_1(t) \wedge \overline{x_2}(t) \vee z(t); \end{cases}$

16. $\begin{cases} z(t+1) = x(t), \\ y(t) = x(t) \leftrightarrow z(t); \end{cases}$

17. $\begin{cases} z_1(t+1) = \overline{x}(t) \wedge (z_1(t) \leftrightarrow z_2(t)) \vee x(t) \wedge z_2(t), \\ z_2(t+1) = \overline{x}(t) \wedge z_2(t) \vee x(t) \wedge z_1(t), \\ y_1(t) = \overline{x}(t) \wedge (z_1(t) \vee z_2(t)) \vee x(t) \wedge z_1(t), \\ y_2(t) = \overline{x}(t) \wedge z_1(t) \wedge z_2(t) \vee x(t) \wedge z_1(t) \wedge \overline{z_2}(t); \end{cases}$

18. $\begin{cases} z_1(t+1) = x(t) \wedge z_2(t) \vee z_1(t) \wedge \overline{z_2}(t), \\ z_2(t+1) = \overline{z_1}(t) \wedge (\overline{z_2}(t) \wedge x(t) \vee \overline{x}(t) \vee \overline{x}(t) \wedge z_2(t)) \vee z_1(t) \wedge x(t), \\ y(t) = z_1 \wedge x(t) \vee z_1(t) \wedge z_2(t) \vee \overline{x}(t); \end{cases}$

19. $\begin{cases} z_1(t+1) = z_1(t) \downarrow x(t), \\ z_2(t+1) = z_1(t) \to z_2(t), \\ y(t) = x(t) \oplus z_1(t); \end{cases}$

20. $\begin{cases} z_1(t+1) = z_2(t) \wedge x(t), \\ z_2(t+1) = \overline{z_1}(t) \wedge \overline{x}(t) \vee z_1(t) \wedge x(t), \\ y(t) = \overline{z_1}(t) \wedge \overline{z_2}(t) \vee z_1(t) \wedge x(t); \end{cases}$

21. $\begin{cases} z_1(t+1) = z_1(t) \oplus z_2(t) \oplus x(t), \\ z_2(t+1) = z_2(t) \to x(t), \\ y(t) = z_2(t) \leftrightarrow x(t). \end{cases}$

Для автомата, заданного диаграммой Мура, выпишите соответственную таблицу и систему булевых функций.

22. См. рис. 9.6.

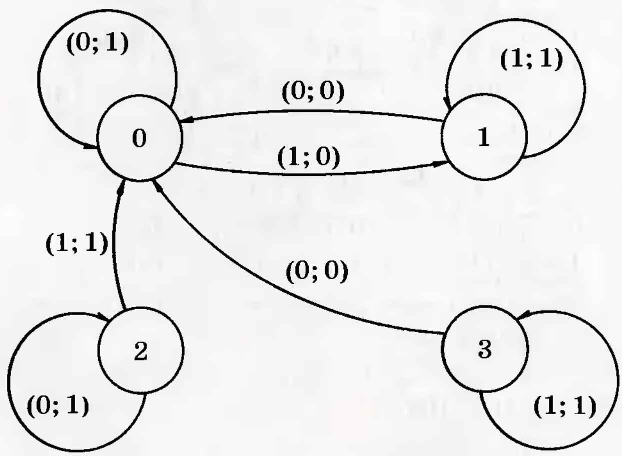


Рис. 9.6

23. См. рис. 9.7.

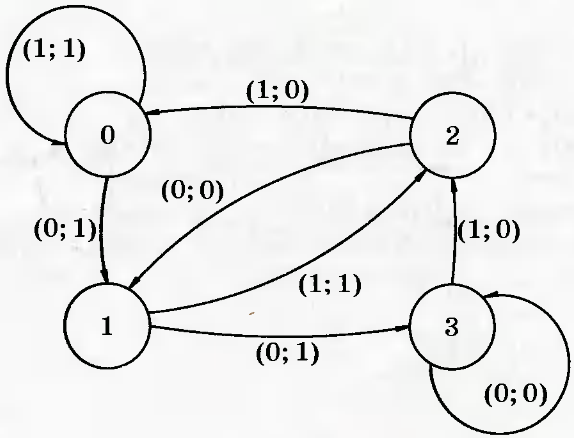


Рис. 9.7

24. См. рис. 9.8.

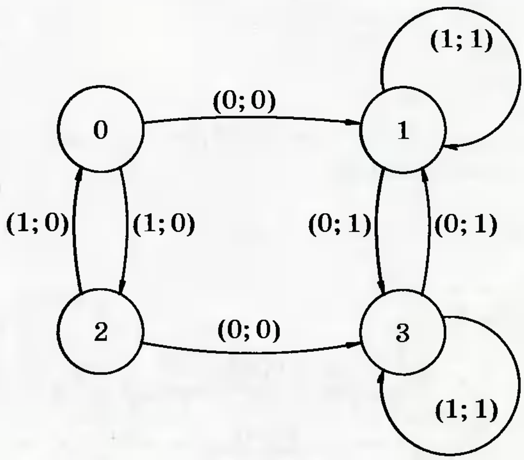


Рис. 9.8

25. См. рис. 9.9.

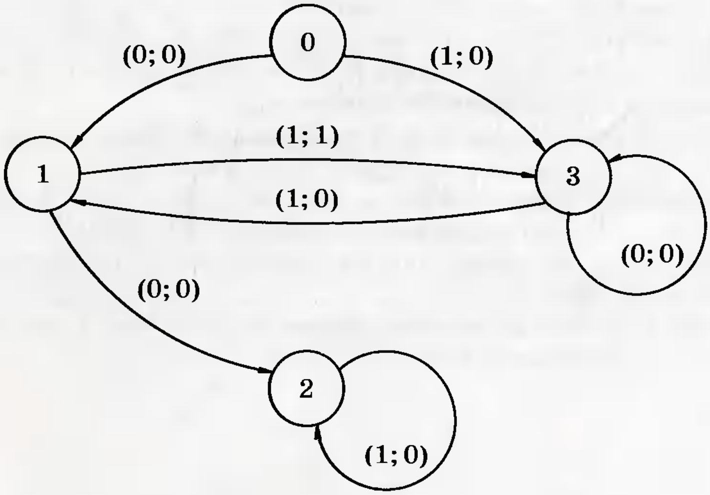


Рис. 9.9

Рекомендуемая литература

1. *Капитонов Ю.В. и др.* Лекции по дискретной математике. СПб. 2004.

2. *Ерусалимский Я.М.* Дискретная математика. М.: Вузовская книга, 2005.

3. *Судоплатов С.В., Овчинникова Е.В.* Элементы дискретной математики. М.–Новосибирск: ИНФРД-М НГТУ, 2002.

4. *Нефёдов В.Н., Осипова В.А.* Курс дискретной математики М.: Изд-во МАИ, 1992.

5. *Кузнецов О.П.* Дискретная математика для инженеров. М., 2005.

6. *Романовский И.В.* Дискретный анализ. СПб.–М., 2000.

7. *Горбатов В.А.* Фундаментальные основы дискретной математики. М.: Наука; Физматлит, 2000.

8. *Шапорев С.Д.* Математическая логика. Курс лекций и практических занятий. СПб.: БХВ-Петербург, 2005.

Учебное издание

Галушкина Юлия Ивановна
Марьямов Александр Наумович

КОНСПЕКТ ЛЕКЦИЙ ПО ДИСКРЕТНОЙ МАТЕМАТИКЕ

Ведущий редактор *В. В. Черноруцкий*
Художественный редактор *А. М. Драговой*
Технический редактор *Т. В. Исаева*
Компьютерная верстка *Е. Г. Иванов*
Корректор *З. А. Тихонова*

Подписано в печать 27.02.07. Формат 70×100/16.
Печать офсетная. Печ. л. 11. Усл.-печ. л. 14,3.
Тираж 5000 экз. Заказ № 6234.

ООО «Издательство «АЙРИС-пресс»
113184, Москва, ул. Б. Полянка, д. 50, стр. 3.

Отпечатано в ОАО «Можайский полиграфический комбинат»
143200, г. Можайск, ул. Мира, 93

Логика, множества, алгоритмы, автомат, графы, кодирование — все эти слова знакомы многим школьникам и студентам. Но то, что все эти термины объединяет одна наука — дискретная математика, станет ясно после ознакомления с данным конспектом лекций.

Курс лекций рассчитан на студентов нематематических факультетов вузов, впервые изучающих дискретную математику. Для успешного освоения основ аппарата дискретной математики не нужно никаких предварительных знаний, кроме правил арифметики. Простая, доступная и в то же время строгая форма изложения теоретического материала, большое количество задач и упражнений с подробными решениями и ответами позволят студенту освоить новый курс и успешно сдать экзамены.

Пособие заложит основы для дальнейшего совершенствования в отдельных областях дискретной математики, даст ключ к входу в безграничный мир компьютеров, программирования, кибернетики и криптологии.

Книга может быть также полезна преподавателям, предполагающим читать новый курс.

АЙРИС ПРЕСС

9 785811 225996