В.П. Битюцкий, С.С. Соколов

Основы дискретной математики

Часть 1

Федеральное агенство по образованию
ГОУ ВПО «Уральский государственный технический университет-УПИ»

В.П. Битюцкий, С.С. Соколов

Основы дискретной математики

Часть 1

Учебное пособие

Научный редактор доц., канд.техн.наук Н.В.Закурдаев

Екатеринбург
2005

УДК 512.817
ББК 22.176
Б66

Рецензенты:
кафедра информатики Уральской горной академии (зав.кафедрой д-р тех.наук, проф. М.Б. Носырев);доц., канд.тех.наук Г.Б.Захарова (Институт машиноведения УрО РАН)

Авторы: В. П. Битюцкий, С.С. Соколов.

Б66 **Основы дискретной математики. Часть 1.:** Учебное пособие по дисциплине «Дискретная математика» /В.П. Битюцкий, С.С. Соколов. Екатеринбург: ГОУ ВПО УГТУ-УПИ, 2005. 96 с.
ISBN 5-321-00373-4

Приводятся основные понятия и утверждения из теории множеств, теории отношений, важнейшие операции над графами, используемые в различных технических приложениях, основные понятия алгебры логики, теории групп и полугрупп. Материал сопровождается поясняющими примерами, содержит задачи, решение которых позволит глубже усвоить учебный материал.

Пособие предназначено для студентов специальностей 220100 - Вычислительные машины, комплексы, системы и сети, 071900 - Информационные системы в технике и технологиях.

Библиогр.: 11 назв. Табл. 30. Рис. 15.

УДК 512.817
ББК 22.176

ISBN 5-321-00373-4

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Дискретная математика имеет дело с количественными характеристиками реальных объектов. Эти объекты имеют дискретную структуру, и значения, которые принимают их характеристики, также являются дискретными. С такими объектами встречаются в задачах, решаемых на современных ЭВМ, поэтому разделы дискретной математики прежде всего интересны с точки зрения практических задач. Это заставляет в рассматриваемой работе, где только возможно, давать под названием «семантика» смысловую трактовку понятиям, задачам, методам. Разделы дискретной математики, представленные в работе, определяются стандартом специальности «Вычислительные машины, комплексы, системы и сети» и являются основой, опираясь на которую можно изучать следующие, более сложные темы.

Работу начинает теория множеств и бинарных отношений. Особое внимание уделяется отношению эквивалентности, широко используемому во многих дисциплинах при минимизации (например, в теории автоматов, алгоритмах и пр.)

Естественным продолжением материала первого раздела является теория графов. Из множества задач, сформулированных и решаемых в рамках этой теории, выбраны имеющие простые прикладные трактовки (сетевые графики, транспортные задачи, задачи назначения и др.). Где только возможно, описания методов и алгоритмов сопровождаются семантическими пояснениями.

В разделе «Группы и полугруппы» приводятся основные начальные понятия алгебры, необходимые в дальнейшем для успешного освоения таких дисциплин, как структуры данных, теория кодирования, теория автоматов. В этом разделе рассматривается известная задача В.М. Глушкова, для понимания которой приводится материал о свойствах множеств, порождающих симметрические полугруппы.

Продолжением алгебраической темы является последний раздел, в котором рассматриваются функции алгебры логики. Знание этого материала необходимо в математической логике и теории алгоритмов, теории автоматов, базах данных и базах знаний, экспертных системах и многих других дисциплинах. Важное место здесь занимают вопросы минимизации функций и функциональной полноты.

Все разделы имеют сходную структуру. Вначале даются необходимые определения, способы описания рассматриваемого объекта, определяется структура объекта, затем вводятся операции, позволяющие получать одни объекты из других. Большое внимание уделяется вопросам минимизации.

Материал излагается следующим образом. Вначале, как правило, формулируются теоремы, затем на содержательном уровне поясняется, по сути дела, необходимость условий их справедливости, а достаточность не доказывается, чтобы не перегружать читателя, поскольку тут важен результат, а не способ его получения. Однако там, где доказательство важно, оно приводится полностью.

Изложение материала сопровождается примерами. В конце каждого раздела приводятся задачи, решение которых рекомендуется для лучшего понимания материала. Каждая из них имеет несколько однотипных вариантов, поэтому достаточно найти решение одного или двух вариантов задачи.

Решать задачи рекомендуется ещё и потому, что при этом могут выявиться такие тонкости, на которые не обращают внимания при теоретическом изучении.

Для понимания изложенного в пособии материала необходимы знания в объёме средней школы.

1. МНОЖЕСТВА

1.1. Основные определения

Множество – неупорядоченная именованная совокупность элементов, удовлетворяющая следующим условиям:

- каждый элемент совокупности уникален, т. е. отличим от других;
- для любого объекта существует возможность установить, принадлежит ли он множеству или нет.

Принадлежность элемента a множеству A обозначается $a \in A$ ($\in$ происходит от греческой буквы ε).

Элементы множества в математике принято заключать в фигурные скобки. Таким образом, совокупность {*1*, *2*, *3*, *4*, *5*, *6*} является множеством и оно неотличимо от множества {*1*, *3*, *5*, *2*, *4*, *6*}, поскольку порядок элементов не играет роли. Совокупность {*1*, *2, 3, 1, 3, 5*} множеством не будет, т.к. первое условие в ней не выполняется. Здесь неразличимы элементы, стоящие в записи на третьем и пятом месте (элемент *3*), так же, как и элементы на первом и четвёртом месте (элемент *1*).

Элементами множества могут быть объекты разной природы и структуры. В частности, множества могут сами быть элементами множеств. Примеры: множество студентов одной группы; множество команд языка программирования; множество групп студентов 2-го курса и т.д. В последнем случае элементы (группы студентов) сами являются множествами.

Число элементов множества A обозначается как $|A|$ и называется *мощностью* (*размером, нормой, длиной* и др.) множества A. Вводится множество, не содержащее элементов, обозначаемое символом $\varnothing$ и называемое *пустым* множеством. Пустое множество может встретиться в реальных задачах и не является «изобретением» математиков. Так, например, может оказаться, что множество студентов, получивших две неудовлетворительные оценки, пусто (таких студентов просто нет).

Множество может быть представлено в виде:

- перечисления его элементов, например $A= \{a, b, c, d, e, f\}$;
- свойства, общего для всех его элементов, например $B=\{b_i \mid b_i$ – студенты старше 25 лет};
- процедуры формирования элементов, например $C=\{c_i \mid c_i = n \cdot 2,\ n \in N\}$.

Для сокращения записи используется символ **|** вместо слов «таких, что». В дальнейшем будем <u>применять</u> также символы & для обозначения связки И, | для обозначения связки ИЛИ, квантор общности $\forall a$ (для всех a) и квантор существования $\exists a$ (существует a).

Множества A и B равны, что обозначается как $A=B$, если

$$(\forall a \in A\ \exists b \in B,\ a=b)\ \&\ (\forall b \in B\ \exists a \in A,\ a=b).$$

Это условие лежит в основе методов проверки равенства двух множеств. Необходимость в проверке равенства множеств может возникнуть тогда, когда множества заданы по-разному и нужно убедиться, что множества совпадают. В общем случае проверка равенства множеств - достаточно сложная задача, требующая больших вычислительных затрат.

Если заведомо выполняется только условие, записанное в первой скобке определения равенства, то множество *A* является частью множества *B* или его *подмножеством*, что обозначается как $A \subseteq B$.

Если при этом второе условие не выполняется, то говорят о точном (или строгом) вхождении множества *A* в множество *B*, что обозначается как $A \subset B$.

Для множества *A* множество *B* называется *дополнением A,* если в *B* включены те и только те элементы, которые не принадлежат *A* (обозначается как $B = \sim A$, $B = \bar{A}$ или $B = \rceil A$). Эту операцию ещё называют НЕ, т.е. говорят *B* равно НЕ *A*.

Предполагается, что дополнение происходит до некоторого *универсального* множества (*универсума*), определяемого *предметной областью задачи*. Универсальное множество обозначается символом *U*. Любое множество является подмножеством универсального множества. Например, универсальным множеством может быть множество студентов факультета, и для него можно рассматривать множества студентов конкретных групп, студентов, получающих именные стипендии и т.п.

Упорядоченная совокупность элементов называется *кортежем*, *вектором* или *упорядоченным множеством*. Элементы кортежа заключаются в угловые < > или круглые () скобки. В кортеже элементы «приписаны» к месту. Различные элементы могут принимать одно и то же значение, в этом случае они отличаются друг от друга тем, что занимают разные места.

Примеры кортежей: <*1*, *2*, *4*, *6*, *2*>; <*2*, *4*, *a*, *d*>; <*a*, *d*, *2*, *4*>. **Удалено:** ,

Удалено: ,

Два кортежа *A* и *B* равны, если их компоненты попарно равны, т.е. $\forall i \quad a_i = b_i$. Таким образом, <*2*, *4*, *a*, *d*> и <*a*, *d*, *2*, *4*> - разные кортежи, хотя и состоят из одних и тех же элементов. **Удалено:** ,

1.2. Операции над множествами

Операция $\Re(A_1, A_2, \ldots, A_k)=B$ сопоставляет нескольким множествам $A_1, A_2, \ldots, A_k$ множество *B* - результат операции. Число *k* называется арностью операции $\Re$. **Удалено:** операций

Одну операцию мы уже ввели – операцию *дополнения*. Её арность равна 1 (унарная операция). **Удалено:** -

Пусть даны два множества *A* и *B* и множество *C* является результатом операции над ними (бинарная операция). Перечислим элементарные бинарные операции:

- *пересечение* множеств $C = A \cap B$, если $C=\{c \mid c \in A \;\&\; c \in B\}$. Эту операцию называют еще умножением множеств или операцией *И*;
- *объединение* множеств $C = A \cup B$, если $C=\{c \mid c \in A \mid c \in B\}$. Другое название операции - сложение множеств или операция *ИЛИ* ;

- *разность* множеств $C = A \backslash B$, если $C=\{c \mid c \in A \ \& \ c \notin B\}$. Иначе операцию называют A без B;
- *симметрическая разность* $C = A \Delta B$, если $C=\{c \mid c \in A \backslash B \cup c \in B \backslash A\}$.

Результат любой описанной операции снова является множеством той же предметной области, его можно использовать в качестве аргумента операций над множествами. Таким образом, можно строить сложные *формулы,* описывающие множество через другие множества. Например, $(A \cap \overline{B}) \cup (\overline{A} \cap B \cap C)$.

Две формулы *эквивалентны*, если им соответствуют равные множества. Обозначим это как $F1=F2$.

Операции на множествах можно графически представить в виде кругов Эйлера, когда множествам сопоставляются замкнутые фигуры на плоскости, взаимное расположение которых определяет результат операции (рис. 1.1). Так, пересечение двух фигур, сопоставленных

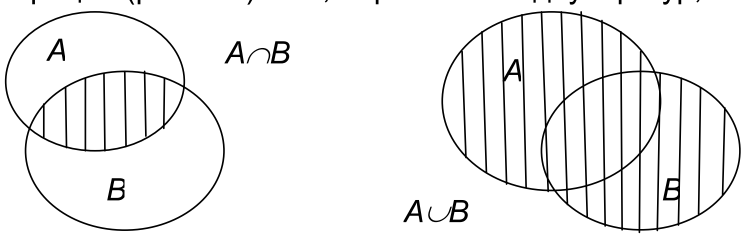


Рис. 1.1

множествам A и B, образует новую замкнутую фигуру, соответствующую общей части фигур A и B – результату операции пересечения, и т.п.

Разбиением или *покрытием* множества A называют множество его подмножеств $\{A_1, A_2, ..., A_k\}$ такое, что имеет место $(A_1 \cup A_2 \cup ... \cup A_k)=A$, и для любой пары подмножеств $A_i \cap A_j=\varnothing$, если $i \neq j$. При этом говорят, что множество A *разбито* на подмножества $A_1, A_2, ..., A_k$ или *покрыто* подмножествами $A_1, A_2, ..., A_k$, а подмножества $A_1, A_2, ...$ называются *классами разбиения* или *классами покрытия*. Выбор того или другого термина определяется смыслом предметной задачи.

1.3. Свойства операций

1. $A \cap A=A$ — *идемпотентность* операции пересечения.
2. $A \cap B = B \cap A$ —*коммутативность* операции пересечения.
3. $A \cap (B \cap C) = (A \cap B) \cap C$ — *ассоциативность* операции пересечения.
4. Свойства 1-3 имеет также операция $\cup$.
5. $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$ — *дистрибуция* операции пересечения по отношению к операции объединения. Справедливо также свойство дистрибуции для операции объединения относительно операции пересечения.
6. Дополнение дополнения множества равно множеству, т.е. $\sim\sim A=A$.

Удалено: ¶ <sp>¶ ¶ ¶ ¶ ¶

Удалено: ∩.
Удалено: ∩
Удалено: ∩.
Удалено: ∪.
Удалено: ∩
Удалено: ∪
Удалено: Аналогично
Удалено: выполняется
Удалено: я
Удалено: ∪
Удалено: ∩
Удалено:

7. Равенства де Моргана $\sim(A \cap B) = \sim A \cup \sim B$, $\sim(A \cup B) = \sim A \cap \sim B$.
8. Для любого множества $A \cap U = A$, $A \cap \varnothing = \varnothing$, $A \cup \varnothing = A$, $A \cup U = U$, $A \cap \sim A = \varnothing$; $A \cup \sim A = U$; $A \backslash A = \varnothing$.

Представленные в свойствах равенства используют для преобразования формул. Если в формуле заменить множество равным ему множеством, получится формула, равносильная исходной. Таким образом, можно в формулах удалять одинаковые члены (свойство 1), выносить за скобки или раскрывать скобки (свойство 4) и т.п.

Пример. $(A \cap B \cap C) \cup (A \cap B \cap \sim C) = (A \cap B) \cap (\sim C \cup C) = A \cap B$.

1.4. Уравнения на множествах

Пусть задано равенство двух множеств, определённых формулами, которое устанавливает отношения между входящими в формулы множествами. Необходимо выяснить эти отношения в терминах взаимного включения множеств. Часто второе множество задано как пустое.

Решение задачи основано на следующих простых соотношениях: если $A \cup B \cup C = \varnothing$, то $A = \varnothing$, $B = \varnothing$, $C = \varnothing$;
если $A \cap B = \varnothing$, то $A \subseteq \sim B$ или $B \subseteq \sim A$.

Если уравнение записано в виде $F1 = F2$, где $F2$ - непустое множество, то это уравнение можно привести к виду, когда справа будет стоять пустое множество. Для этого используем соотношение, вытекающее из определения равенства множеств: если $F1 = F2$, то из этого следует, что $(F1 \cap \sim F2) \cup (\sim F1 \cap F2) = \varnothing$.

Пример. Уравнение имеет вид: $(\sim A \cap B) \cup (A \cap \sim B \cap C) = \varnothing$. Значит, $(\sim A \cap B) = \varnothing$ и $(A \cap \sim B \cap C) = \varnothing$. Откуда следует, что $B \subseteq A$ и $(A \cap C) \subseteq B$.

1.5. Декартово произведение множеств

Декартовым произведением множеств A и B называют множество M, содержащее все возможные пары, в которых на первом месте стоит элемент из A, на втором – элемент из B. Формально $AxB = M$, $M = \{(a_i, b_j) \mid a_i \in A, b_j \in B\}$. Элементы декартова произведения являются кортежами.

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

Если $|A| = n_a$ и $|B| = n_b$, то $|AxB| = n_a \cdot n_b$

Аналогично вводится произведение трех, четырех и т.д. множеств как множество троек, четверок и так далее, в которых на первом месте записан элемент первого множества, на втором - элемент второго и т.д. Если все сомножители в произведении одинаковы, оно называется декартовой степенью: $A = A^1$; $AxA = A^2$; $AxAxA = A^3$; $AxAx\ A...A = A^k$, $k > 0$.

Результат декартова произведения не является подмножеством универсального множества.

Для декартова произведения не выполняются условия коммутативности и ассоциативности. Действительно, множество AxB

Удалено: определяет
Удалено: ∅.
Удалено: Если
Удалено: же
Удалено: Пример
Удалено: Произведение
Удалено: (декартово произведение)
Удалено: -
Удалено: ¶
Удалено: Пример
Удалено: a
Удалено: b
Удалено: a
Удалено: b
Удалено: .
Удалено: д.
Удалено: т.д

содержит в качестве элементов пары (a_i ,b_j), а множества *BxA* — пары (b_i ,a_j), т.е. результирующие множества содержат различные элементы. Элементы произведения (*AxB*)x*C* - пары ((a_i ,b_j),c_k), а *Ax(BxC)* – пары (a_i ,(b_j ,c_k)). Элементами множества *AxBxC* являются тройки (a_i ,b_j ,c_k). Значит, (*AxB*)x*C* ≠ *A* x (*B*x*C*) ≠ *AxB*x*C*.

1.6. Контрольные вопросы и задания

1.6.1. Задачи на множествах

1. Чему равна мощность множества ∅? А множеств $\{\{\varnothing\}\}$, $\{\{\{\varnothing\}\}\}$?
2. Даны два множества, такие, что $A \cap B = \varnothing$. Что представляют собой множества *A\B* и *B\A* ?
3. Даны два множества $C \cap \sim D = \varnothing$. Что можно сказать о множествах *C\D* и *D\C* ?
4. Даны множества *A, B, C*. Определить множество, включающее в себя элементы, входящие только в два из этих множеств.
5. Решить предыдущую задачу при условии, что множества *A, B* и *C* взаимно не пересекаются.
6. Доказать, что для любых двух множеств *A* и *B* справедливо равенство $\sim(A \cap B) = (A \cap \sim B) \cup (\sim A \cap B) \cup (\sim A \cap \sim B)$.
7. Доказать, что для любых *A, B, C* имеет место равенство $(A \cap B) \cup (B \cap C) \cup (A \cap C) = (A \cup B) \cap (B \cup C) \cap (A \cup C)$.
8. Пусть даны множества *A, B* и *C*, $C \subseteq B$. Какие из нижеследующих утверждений верны: а) $(A \cap C) \subseteq (A \cap B)$; б) $(A \cup C) \subseteq (A \cup B)$; в) $A \setminus B \subseteq A \setminus C$; г) $C \setminus A \subseteq B \setminus A$; д) $\sim B \setminus A \subseteq \sim C \setminus A$?
9. Пусть *A1, A2,..., Aк* множества, и $C_i = A1 \cup A2 \cup ... \cup Ai$, $i=1,...,k$. Доказать, что $S=\{A1, A2 \setminus C1, A3 \setminus C2, ..., Ak \setminus Ck\text{-}1\}$ -разбиение множества *Cк*.
10. Какие утверждения верны для любых множеств *A, B, C*:
    а) если $A \not\subset B$ и $B \not\subset C$, то $A \not\subset C$;
    б) если $A \subseteq B$ и не верно, что $B \subseteq C$, то $A \not\subset C$.
11. Доказать, что для любых конечных подмножеств M, N, K всегда имеет место:
    - $|M \cup N|=|M|+|N|-|M \cap N|$;
    - $|M \cup N \cup K|=|M|+|N|+|K|-|M \cap N|-|M \cap K|-|K \cap N|+|M \cap N \cap K.|$

    Сформулируйте и докажите аналогичную формулу для произвольного числа конечных подмножеств.

1.6.2. Уравнения на множествах

1. В каком отношении по включению находятся множества A. B, C, если выполняются равенства (a - j)? Постройте решение аналитически и изобразите его в виде кругов Эйлера.

a) $A \cap ((\sim B \cap \sim C) \cup (B \cap C)) \cup \sim A \cap ((B \cap \sim C) \cup (\sim B \cap C)) = \varnothing$.

b) $(A \cap B) \cup C = A \cap (B \cup C)$.

Удалено: о есть
Удалено: -
Удалено:
Удалено: ¶
Удалено:
Удалено:
Удалено:
Удалено:
Удалено: а
Удалено:)
Удалено:
Удалено:
Удалено:
Удалено:
Удалено:
Удалено:
Удалено: . Пусть
Удалено: 10.
Формат: Список
Удалено:
Удалено: 11.
Формат: Список
Удалено: ¶
Формат: Список

c) $A \cap \sim B = C \cup B$.
d) $A \cap \sim B = C \cap B$.
e) $(A \cup B) \cap C = A \cap \sim B$.
f) $(A \cup B) \cap \sim C = A \cap \sim B$.
g) $(A \cup \sim B) \cap C = \sim A \cap B$.
h) $(A \cup B) \cap C = A \cap B \cap C$.
i) $(\sim A \cup B) \cap C = (A \cap B \cap C$.
j) $((C \cup B) \cap A) \cup (\sim A \cap (\sim B \cup \sim C)) = \varnothing$.

2. Множество X определяется через множества A, B и C условием $X \cup C = A \cap B$. При каких условиях на A,B,C возможно решение? Найти X.

Удалено: ¶

1.6.3. Доказательство тождеств

Пояснения. Используя определения операций на множествах и исходя из отношения принадлежности, можно доказывать справедливость тождеств. Например: $A \cap (\sim A \cup B) = A \cap B$.

Доказательство. Данное тождество означает, что каждый элемент множества $A \cap (\sim A \cup B)$ принадлежит и множеству $A \cap B$ и наоборот.

Удалено: ¶
Удалено: ,
Удалено: а
Удалено: но

Пусть из $x \in A \cap (\sim A \cup B)$ следует, по определению $\cap$, что $x \in A$ и из $x \in (\sim A \cup B)$ следует ($x \in A$ и $x \in \sim A$) или ($x \in A$ и $x \in B$). Так как x не может одновременно принадлежать множествам A и $\sim A$, то получаем $x \in A$ и $x \in B \Rightarrow x \in A \cap B$.

Обратно: пусть $x \in A \cap B \Rightarrow x \in A$ и $x \in B$. Из того, что $x \in B$, следует, что x может принадлежать объединению B с любым множеством, тогда пусть $x \in (\sim A \cup B)$, и в итоге $x \in A \cap (\sim A \cup B)$.

Удалено: ,
Удалено: ,

Докажите следующие тождества:

Удалено: ¶

1. $A \cup (B \setminus A) = A \cup B$.

2. $A \cap (B \setminus C) = (A \cap B) \setminus C$.

3. $A \setminus (B \cap C) = (A \setminus B) \cup (A \setminus C)$.

4. $A \cap (B \setminus A) = \varnothing$.

5. $A \setminus (A \cap B) = (A \setminus B)$.

6. $A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C)$.

7. $(A \cup B) \setminus C = (A \setminus C) \cup (B \setminus C)$.

Удалено: C

2. ОТОБРАЖЕНИЯ И ОТНОШЕНИЯ

Пусть задано два множества A и B. Выделим некоторое подмножество $\mathfrak{R}$ декартова произведения AxB и будем трактовать его элементы (a_i,b_j) как выражение того факта, что a_i и b_j находятся в некотором соответствии. Нас не интересует характер этого соответствия, а только сам факт его наличия. Множество $\mathfrak{R}$ назовем *отображением* множества A на множество B.

Если $(a_i, b_j) \in \mathfrak{R}$, то a_i называется *прообразом* b_j, а b_j - *образом* a_i при отображении $\mathfrak{R}$. Множество $A^{\sim}$ всех прообразов в A есть область определения отображения $\mathfrak{R}$, а множество $B^{\sim}$ всех образов в B - область значений $\mathfrak{R}$. Если $B^{\sim}$ равно B, то говорят об отображении *на* B, если $B^{\sim}$ - только часть B, то об отображении *в* B.

Отображение будем обозначать как $A\mathfrak{R}B$ или $\mathfrak{R}(a) = b$. Если для каждого a_i образ b_j единственен, то отображение называют *функциональным*. Если в $\mathfrak{R}$ все пары (a_i, b_j) переписать «наоборот», как (b_j, a_i), получим отображение B в A, которое является *обратным* к $\mathfrak{R}$ и обозначается $\mathfrak{R}^{-1}$.

Пусть множества A и B совпадают, $\mathfrak{R} \subseteq A$xA. В этом случае $\mathfrak{R}$ называют *бинарным отношением*, а множество A – *базовым множеством отношения* R.

Если $(a_i, a_j) \in \mathfrak{R}$, то говорят, что элемент a_i находится в отношении с элементом a_j. В общем случае можно определить, что в отношении находится не пара, а k элементов, считать, что $\mathfrak{R} \subseteq A^k$. Величина k определяет арность отношения $\mathfrak{R}$. Говорят, что $\mathfrak{R} \subseteq A^k$ - *k-арное отношение*.

Термин «отношение» используют также, если арность >2 и множества в декартовом произведении различны.

Будем рассматривать в дальнейшем бинарные отношения на множестве A.

2.1. Способы описания бинарного отношения

Бинарное отношение $\mathfrak{R}$ как любое подмножество может быть представлено в виде перечисления, через указания свойства или через порождающую процедуру. Но наиболее часто используется представление *матрицей*, в котором учитывается специфика множества. Столбцам и строкам матрицы сопоставлены элементы базового множества A, значение элемента матрицы (a_i, a_j) равно 1, если $(a_i, a_j) \in \mathfrak{R}$, в противном случае значение соответствующего элемента равно 0.

2.2. Виды бинарных отношений

Бинарное отношение называется *рефлексивным*, если $\forall(a_i)\in A$ $(a_i\,,a_i\,)\in\ \Re$. Если отношение рефлексивное, то в каждой клетке главной диагонали стоят единицы.

Удалено: ¶
Удалено: является
Удалено: ым

Бинарное отношение *антирефлексивно*, если $\forall(a_i\,)\in\ A\quad(a_i\,,a_i\,)\notin\Re$.
В антирефлексивном отношении главная диагональ не содержит ни одной единицы.

Удалено: ¶

Бинарное отношение называется *симметричным*, если из того, что $(a_i,a_j)\in\ \Re$, следует $(a_j\,,a_i\,)\in\ \Re$. Для симметричного отношения таблица симметрична относительно главной диагонали.

Удалено: ¶

Бинарное отношение *антисимметрично*, если из того, что $(a_i,a_j)\in\Re$, следует, что$(a_j\,,a_i\,)\notin\ \Re$.

Бинарное отношение называется *транзитивным*, если из того, что $(a_i\,,a_j\,)\in\Re$ и $(a_j\,,a_k\,)\in\Re$, следует $(a_i\,,a_k\,)\in\Re$.

2.3. Эквивалентность

Бинарное отношение называется *отношением эквивалентности*, если оно одновременно рефлексивно, симметрично и транзитивно.

Два элемента связаны отношением эквивалентности, если они имеют одинаковое свойство из множества альтернативных свойств. Примерами таких отношений являются принадлежность студентов к одной учебной студенческой группе, отношение родства или отношение «иметь одинаковый цвет волос». Альтернативность предполагает, что случаи, когда один студент принадлежит к нескольким группам или один человек имеет разноцветные волосы, из рассмотрения исключаются (иначе не выполнялась бы транзитивность). Тогда множество разбивается на непересекающиеся подмножества элементов, удовлетворяющие свойству, которые при объединении покрывают все множество. Последнее обеспечивается свойством рефлексивности, когда для каждого элемента находится элемент, с которым он состоит в отношении (по крайней мере, с самим собой). Эти подмножества называются *классами эквивалентности*.

Удалено: находится
Удалено: -

Справедливо утверждение: *любому отношению эквивалентности однозначно сопоставляется разбиение множества и, обратно, любому разбиению множества однозначно сопоставляется отношение эквивалентности*.

2.4. Отношение порядка

Говорят, что отношение $\Re$ отвечает свойству *дихотомии*, если из того, что $(a_i\,,a_j\,)\notin\Re$, следует, что $(a_j\,,a_i\,)\in\Re$. Значит, если выполняется свойство дихотомии, то в множестве любые два элемента находятся в данном отношении. Примером дихотомии может служить отношение «быть по возрасту не старше» между людьми. Здесь, если Иван старше Петра, то, значит, Петр не старше Ивана.

Отношение $\mathfrak{R}$ называют отношением *порядка,* если оно антисимметрично и транзитивно. Если при этом отношение рефлексивно, то оно называется *отношением нестрогого порядка*, если антирефлексивно – *отношением строгого порядка*.

Удалено: то

Отношение нестрогого порядка между элементами a_i и a_j обозначается $a_i \le a_j$, отношение строгого порядка – $a_i < a_j$.

Удалено: ,

Приведенное выше отношение «быть не старше» является отношением нестрогого порядка, так как оно рефлексивное, антисимметричное и транзитивное. Примером отношения строгого порядка будет определенное на множестве людей отношение «быть старше», так как любой человек не может быть старше себя.

Удалено: является и

Удалено: ым

Удалено: и

Удалено: ым

Удалено: ,

Удалено: транзитивным

Удалено: ,

И наконец, если ввести в отношение необходимость дихотомии, то получим соответственно отношение *полного нестрогого* или *полного строгого порядка.*

Если в отношении условия дихотомии не выполняются, то отношение называется отношением *частичного порядка*.

Если на множестве людей определить отношение «быть начальником по службе» и считать, что начальник моего начальника является моим начальником (выполняется транзитивность), то это есть частичный порядок, ибо люди, работающие в разных организациях, в этом отношении не находятся. Строгий это порядок или нет, определится тем, считаем ли мы себя начальниками по службе для самих себя или нет.

Удалено: если

Элемент a_i *непосредственно следует* за a_j, если $a_i < a_j$ и не найдется такого a_k, что $a_i < a_k < a_j$. Значит, для любых элементов a_i и a_j, таких, что $a_i < a_j$, найдется цепочка элементов между a_i и a_j, в которой любой элемент непосредственно следует за предыдущим. Все такие цепочки определяют *схему транзитивного отношения*.

Для полного множества, когда любая пара элементов находится в отношении, цепочка будет единственной. Если множество A конечно, то при полном порядке всегда найдется единственный элемент $a_{\mathbf{max}}$ такой, что $\forall a \in A \ \ a_{\mathbf{max}} > a$. Этот элемент называют *максимальным*.

Аналогично для конечного множества в этом случае найдется *минимальный* элемент, меньший любого элемента из *A.*

Для частичного порядка в конечном множестве минимального и максимального элементов может и не быть. Назовём *наибольшим* элементом такой элемент, для которого не найдется в *A* элемента, большего его. Точно так_же определим как *наименьший* такой элемент, для которого в *A* нет меньших его. Наибольших элементов (как и наименьших) может быть несколько, они образуют *верхнюю* (*нижнюю*) *грань множества по данному отношению*.

Рассмотрим схему отношения на примере.

Пусть на множестве A={1,2,3,4,5,6} задано отношение {(1,3),(1,4),(1,5),(1,6),(2,3),(2,4),(2,5)(2,6),(3,4), (3,5),(3,6), (4,5), (4,6)}. Тогда цепочками схемы будут (1,3,4,5), (1,3,4,6), (2,3,4,5), (2,3,4,6), что определяет схему, которую удобно представить в виде рис.2.1.

В этом примере элементы 1 и 2 – наибольшие, элементы 5 и 6 – наименьшие. Они образуют соответственно верхнюю и нижнюю грани множества по отношению.

Отношения, в которых есть антисимметрия, но нет транзитивности, называют *предпорядком* или *отношением доминирования.* Примером такого отношения может служить заданное на множестве футбольных команд отношение «победа в игре». Действительно, из того, что *УРАЛМАШ* победил *РОТОР*, а *РОТОР* победил *ДИНАМО*, не следует, что *УРАЛМАШ* победит *ДИНАМО,* т.е. свойство транзитивности здесь не выполняется.

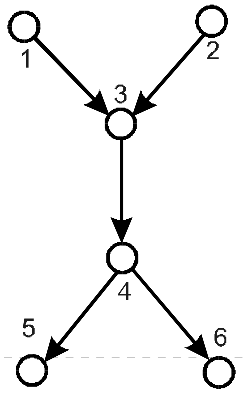


Рис.2.1

2.5. Замыкание отношений

Пусть на множестве *A* задано два отношения R и S. Определим отношение *Q* как результат операции *транзитивного продолжения* отношений R на S, если$(a_i, a_k) \in Q$ тогда и только тогда, когда $(a_i, a_j) \in R$ и $(a_j, a_k) \in S$. Обозначим это $Q = R \circ S$. Если $R = S$, то обозначим $S \circ S = S^2$, и по аналогии введем k-ю степень транзитивного продолжения как последовательное выполнение (k+1) раз операции $\circ$. Обозначим *S* как S^1.

Пример. Пусть на множестве всех людей задано бинарное отношение R «быть отцом». Тогда R^2 будет соответствовать бинарное отношение «быть отцом отца» или, что то же самое, «быть дедом по отцу».

Транзитивным *замыканием* (или просто замыканием) отношения $\Re$ называется бесконечное объединение R^i. Обозначим замыкание как R^*, т.е. $R^* = R \cup R^2 \cup \ldots \cup R^k$.

Пример. Пусть на множестве целых чисел *N* задано отношение $R = \{(x,y) | y = x+1\}$. Тогда замыканием R^* будет отношение $\{(x,y) \mid x < y\}$.

2.6. Основные понятия комбинаторики

Пусть задано декартово произведение $R = A_1 x A_2 x \ldots x A_n$. Элементы этого множества назовём *n-выборкой*. Чаще всего все сомножители произведения равны, т.е. $R = A^n$. Элемент этого множества называют ещё *упорядоченной n-выборкой* или *размещением из m элементов по n*. Здесь m – число элементов в множестве *A*.

Число элементов в A^n (число размещений из m элементов по n) равно m^n. Эта величина обозначается P^n_m.

Удалено: 2
Удалено: то есть
Удалено: 2
Удалено: R
Удалено: S
Удалено: R
Удалено: S
Удалено: $)\in Q$
Удалено: $)\in R$
Удалено: $)\in S.$
Удалено: $=R\circ S.$
Удалено: R
Удалено: S,
Удалено: $S\circ S=S$
Удалено: ,
Удалено: ,
Удалено: ,
Удалено: S
Удалено: Пример
Удалено: $\Re$
Удалено: $\Re$
Удалено: $\cup\Re$
Удалено:
Удалено: $=\Re\cup$
Удалено: $\Re$
Удалено: $\cup\ldots\cup\Re$
Удалено: $=\Re^*$
Удалено: Пример
Удалено: $\Re=\{($
Удалено: $\Re^*$
Удалено: x
Удалено: x
Удалено: ¶
Удалено: я
Удалено: то есть
Удалено: -

Два размещения равны, если попарно равны все их компоненты.

Если n=m, то это размещение называют *перестановкой m элементов*. Число различных перестановок равно m^m.

Пример. Пусть множество *A* = {1,2,3,4} и n=2. Тогда A^n= {(1,1), (1,2), (1,3), (1,4), (2,1), (2,2), (2,3), (2,4), (3,1), (3,2), (3,3), (3,4), (4,1), (4,2), (4,3), (4,4)}. Число элементов в A^n =16.

Число упорядоченных n-выборок без повторения элементов в выборках (размещений из m элементов по n без повторения) равно m!/(m-n)! .Эту величину принято обозначать A^n_m или (m/n).

Пример. Построим все размещения по 2из 4 элементов множества *A* рассмотренного примера, получим {(1,2), (1,3), (1,4), (2,1), (2,3), (2,4), (3,1), (3,2), (3,4), (4,1), (4,2), (4,3)}. Это множество содержит 12 элементов (4!/2!=(4·3·2)/2).

При n = m A^m_m = m! – число перестановок из m элементов.

Если в n-выборке не учитывать порядок компонент, то выборка называется *неупорядоченной выборкой* или *сочетанием из m элементов по n*.

Две неупорядоченные n-выборки равны, если каждая из них состоит из одних и тех же компонент из *A*, взятых одно и то же число раз.

Число неупорядоченных n выборок из m элементов без повторения (*сочетаний без повторения*), обозначаемое как N(m,n)= A^n_m/n! = C^n_m, где C^n_m – коэффициенты из биномиальной теоремы, получаемые по треугольнику Паскаля. C^n_m= m! / (n!·(m-n)!)

Пример. Множество всех возможных сочетаний из 4 элементов по 2 будет {(1,2), (1,3), (1,4), (2,3), (2,4), (3,4)}. Число элементов в этом множестве (4·3·2/2·2) =6.

Число неупорядоченных n - выборок из m с повторением будет C^n_{m+n-1} *Пример*. Множество неупорядоченных 2 - выборок из 4 будет {(1,1), (1,2), (1,3), (1,4), (2,2), (2,3), (2,4), (3,3), (3,4), (4,4)}. Их число равно (5·4·3·2/3·2·2).

Задачи комбинаторики.

Выделяют следующие задачи комбинаторики:

1. *Задача пересчёта*. Сколько элементов из заданного множества имеют некоторое свойство?

2. *Задача перечисления*. Перечислить элементы, обладающие заданным свойством.

3. *Задача классификации*. Если пересчёт приводит к большим числам, то элементы классифицируются с помощью какого-то соотношения.

4. *Задача минимизации*. На множестве элементов задана некоторая функция, необходимо найти элементы с экстремальными значениями этой функции.

Рассмотрим эти задачи на примере. Пусть задан квадрат, разделённый на kxk клеток (например, 5x5, см.табл.2.1). Необходимо разместить в нём k единичек так, чтобы в каждой строке и каждом столбце было по одной единичке.

Таблица 2.1

	1	2	3	4	5
1		1			
2	1				
3				1	
4					1
5			1		

Задача пересчёта. Сколько вариантов решения существует? Для заданного примера это число будет k!: 5!=120.

Задача перечисления. Перечислить все 120 вариантов.

Задача классификации. Будем рассматривать квадрат как матрицу смежности ориентированного графа. Тогда приведенному в табл. 2.1 квадрату будет соответствовать граф рис 2.2. Будем классифицировать варианты по одинаковости этого графа с точностью до имен его вершин. Сколько типов графов здесь возможно и каковы они?

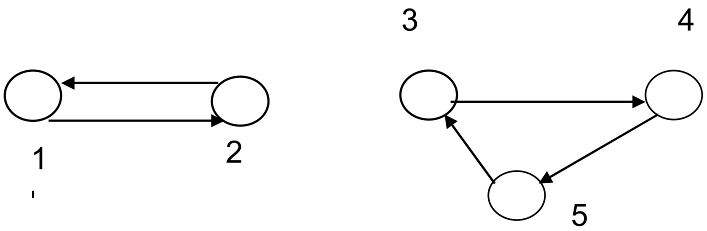


Рис.2.2

Задача минимизации. Сопоставим каждой клетке графа целое число (табл. 2.2). Задача заключается в выборе варианта, для которого сумма чисел в отмеченных единичками клетках будет максимальна. Эта задача может быть представлена как задача построения максимального совершенного паросочетания в двудольном графе, описанном данной матрицей. Эта задача будет рассмотрена в следующей главе.

Таблица 2.2

	1	2	3	4	5
1	3	5	4	5	2
2	3	2	6	2	4
3	2	5	2	2	8
4	4	3	5	7	6
5	6	5	4	5	3

Пояснение. Биномиальные коэффициенты C^k_n определяют коэффициенты при k-й степени *a* при разложении выражения $(1+a)^n$. Здесь 1 соответствует a^0 . Тогда C^0_1=1 и C^1_1=1, C^0_2=1, C^1_2=2 , C^2_2=1.

Удалено: 1.
Удалено: , или
Удалено: 2.
Удалено: 3.
Удалено:
Удалено: ¶ <sp>
Удалено: 4.
Удалено: 3
Удалено: С
Удалено: о
Удалено: С
Удалено: С
Удалено: С
Удалено: С
Удалено: С

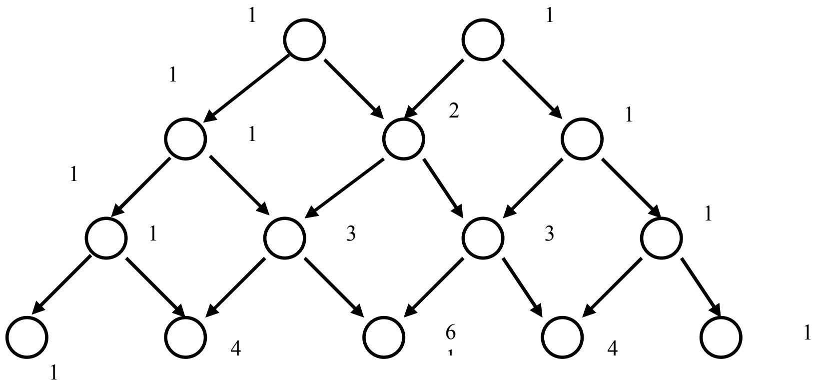


Рис.2.3.

Паскаль предложил считать коэффициенты C^k_n рекурсивно по следующему треугольнику (треугольник Паскаля, рис.2.3.). Здесь вершины расположены по горизонтальным рядам – ярусам. Ярусы нумеруются сверху вниз последовательно 1, 2, 3 и т.д.. Номер яруса определяет число n. Числа у вершин нижнего яруса получаются как сумма чисел вершин верхнего яруса, от которых к ней идут стрелки. Номер вершины в ярусе (число k) равен месту вершины в ярусе, начиная слева.

Удалено: С

2.7. Контрольные вопросы и задания

2.7.1. Свойства бинарных отношений.

Докажите или опровергните следующие утверждения

1: Если отношения *A* и *B* рефлексивны, то рефлексивны и отношения $A \cup B$, $A \cap B$.

2: Если отношения A и B симметричны, то симметричны и отношения $A \cup B$, $A \cap B$.

3: Если отношение *A* асимметрично, то пересечение $A \cap B$ асимметрично при любом *B*.

4: Если отношения *A* и *B* антисимметричны, то антисимметрично и отношение $A \cap B$.

5: Если отношения *A* и *B* транзитивны, то транзитивно и отношение $A \cap B$.

2.7.2. Отношение эквивалентности

1. Из определения отношения эквивалентности следует, что из того, что $(a,b)\in R$ следует $(b,a)\in R$, а из (a,b) и $(b,c)\in R$ следует $(a,c)\in R$. Из первого и второго условия, положив $a=c$, получим условие $(a,a)\in R$. Значит, из симметрии и транзитивности следует рефлексивность. Так ли это? Если нет, то в чем ошибка?

2. Заданное бинарное отношение R доопределите минимальным образом до отношения эквивалентности R`и выпишите классы эквивалентности для вариантов:

a) R={ (1,3),(2,2),(2,7), (1,5),(5.5),(7,10),(4,6),(8,8),(2,9)}; Удалено: 1:
b) R={(1,1),(1,6),(2,7),(9,10),(4,5),(6,3),(7,9),(8,8)}; Удалено: 2:
c) R={(1,2),(3,5),(7,4),(2,6),(2,2),(5,9)(4,10),(10,10)}; Удалено: 3:
a) R={(4,1),(3,2),(1,5),(6,8),(9,10),(7,7),(10,6)}; Удалено: 4:
b) R={(7,9),(8,8),(4,5),(6,3),(1,1),(9,10),(2,7),(1,6)}; Удалено: 5: R
c) R={(3,1),(2,4),(5,8),(6,2),(10,7),(9,1),(8,11)}; Удалено: 6:
d) R={(7,3),(4,2),(8,9),(9,4),(1,5),(6,11),(10,7),(9,9)}; Удалено: 7:
e) R={(1,1),(8,8),(7,9),(4,5),(6,3),(10,9),(7,2),(1,6)}; Удалено: 8:
f) R={(1,3),(2,2),(2,7),(1,5),(5,5),(10,7),(4,6),(8,8),(9,2)}; Удалено: 9:
g) R={(2,2),(1,5),(1,3),(7,2),(6,4),(10,7),(5,5),(7,7),(9,2)}. Удалено: 10:

3. На декартовом произведении $R\times R$, где R – множество вещественных чисел (кроме 0), задано отношение Q. Является ли это отношение эквивалентностью, если является, то опишите классы эквивалентности для вариантов: Удалено: .

a) $(a,b)Q(c,d)$, если $ad=cd$;
b) $(a,b)Q(c,d)$, если $(a- c)=(b^2- d^2)$;
c) $(a,b)Q(c,d)$, если $(a^2- c^2)=(d^2-b^2)$;
d) $(a,b)Q(c,d)$, если $(a+d^3)=(b^3+c)$;
e) $(a,b)Q(c,d)$, если $a/d =c/b$;
f) $(a,b)Q(c,d)$, если $a^2/c^2=d/b$. Удалено: ;

4. Каждому отношению эквивалентности однозначно сопоставляется разбиение множества на классы и, наоборот, каждому разбиению однозначно сопоставляется отношение эквивалентности. Каково должно быть разбиение конечного множества на два класса, чтобы их декартово произведение имело наибольшее число элементов?

5. Пусть множество M={1,2,...,r} и на M^n определена величина разности между элементами a и b: $\mathrm{I}(a,b)\mathrm{I} =\Sigma|a_i - b_i|$, где под знаком суммы стоит модуль разности. Пусть a и b находятся в отношении R, если $\mathrm{I}(a,b)=1$. Постройте замыкание $\mathfrak{R}$ этого отношения при r = 3. Находятся ли элементы (2,3,1) и (2,1,3) в отношении $\mathfrak{R}$?

2.7.3. Отношение порядка

1. Отношения в задаче 2 раздела 2.7.2 доведите минимальным образом (т.е. исключив минимальное число пар) до отношения частичного порядка. Определите нижнюю и верхнюю грани.

2. Эти же отношения дополните минимальным образом до отношения нестрогого полного порядка. При этом для обеспечения антисимметрии некоторые пары придется исключить.

3. Покажите, что если отношение R – строгий порядок, то симметричное ему R^{-1} также является строгим порядком.

4. То же самое для нестрогого порядка.

5. То же самое для частичного порядка.

2.7.4. Задачи на отображения

Пояснение. Неподвижной точкой при отображении называют элемент множества, для которого образ элемента равен самому элементу.

1. Сколько существует отображений множества $A=\{a, b, c, d\}$ в себя, имеющих неподвижные точки?

2. Пусть N – множество всех вещественных функций, заданных на всей вещественной оси; γ - отображение N в себя, ставящее в соответствие каждой функции $f(x)$ из N функцию $f(x)=(x^2 -1)f(x)$. Будет ли γ взаимно однозначным? Является ли оно отображением N на себя?

Удалено: -

Удалено: f

Удалено: x

Удалено: f

Удалено: x

Удалено: x

Удалено: f

Удалено: x

3. Каждому треугольнику T, длины сторон которого равны a ,$в$ и c, сопоставим треугольник со сторонами $(a+в)/2$, $(a+c)/2$, $(в+c)/2$. Будет ли это отображение множества всех треугольников в себя взаимнооднозначным? Будет ли оно отображением на себя? Какие треугольники будут неподвижными точками этого отображения?

Удалено: ¶

Удалено: ¶

2.7.5. Транзитивное замыкание отображений

1. Пусть $R=\{(a,b) \mid a=b+1,\ a,b\in N\}$. Как выглядят R^2 ,R^3,R^* ?

Удалено: *K*

Удалено: *K*

Удалено: *K*

Удалено:

Удалено: *K*

2. Пусть α и β являются бинарными отношениями в множестве A. Обозначим как умножение $\alpha\cdot\beta$ транзитивное продолжение отношения α на β.

3. Всегда ли из рефлексивности обоих отношений следует рефлексивность $\alpha\cdot\beta$?

4. Всегда ли из транзитивности обоих отношений следует транзитивность $\alpha\cdot\beta$?

5. Всегда ли из симметричности обоих отношений следует симметричность $\alpha\cdot\beta$?

6. Всегда ли из антисимметричности обоих отношений следует антисимметричность $\alpha\cdot\beta$?

3. ГРАФЫ

Понятие «граф» связано с понятием «графический», «графика». Действительно, графовые модели имеют простую и понятную графическую интерпретацию, позволяющую с их помощью образно представить самые разные объекты, в то же время оставаясь в рамках строгих математических моделей.

3.1. Основные определения

Граф является одним из способов описания бинарного отношения, рассмотренного в предыдущем разделе. Легко видеть, что бинарное отношение на множестве - наиболее общее описание связей между элементами множества, указывающее факт наличия или отсутствия связи между парами элементов из множества независимо от характера связи.

Определение. *Графом G называют пару <A, U>, где*

$A=\{a_1,a_2,\dots,a_n\}$ – множество вершин графа;

$U \subseteq A\times A=\{(a_i,a_j)\}$ – множество его дуг.

Как и всякое бинарное отношение, граф представляется матрицей *nxn*, где $n = |A|$, которую называют матрицей смежности. Граф имеет следующую графическую интерпретацию: сопоставим каждому элементу множества *A* (вершинам) кружок на плоскости рисунка и соединим кружки, сопоставленные вершинам a_i и a_j, стрелкой, направленной из первого кружочка во второй, если пара $(a_i,a_j) \subseteq U$. Граф, определённый таким образом, называют ориентированным графом (*орграфом*) или графом Бержа.

Говорят, что дуга (a_i,a_j) *исходит* из вершины a_i и *заходит* в вершину a_j. Вершину a_i называют *началом*, a_j - *концом* дуги. Эти вершины называются *смежными*, или *инцидентными* дуге (a_i,a_j). Две дуги *смежны,* если они имеют общую граничную вершину.

Таблица 3.1

	1	2	3	4	5
1	1	1	0	0	0
2	1	0	1	0	0
3	1	0	0	0	1
4	0	0	1	1	0
5	0	1	0	1	0

Пример. Граф $G=(A,U)$, где $A=\{1,2,3,4,5\}$, и *U* задано матрицей смежности табл. 3.1., изображен на рис.3.1.

Число дуг, исходящих из вершины a_i, называют степенью ее исхода d_i^-, заходящих в a_i – степенью захода d_i^+, сумма степеней исхода и захода ($d_i=d_i^-+d_i^+$) – степенью вершины i. Так, вершина 3 имеет степень захода, равную 2, степень исхода – 2. Степень вершины равна 4.

Удалено: *}-*
Удалено: *множество, называемое*
Удалено: -
Удалено: *множество, называемое*
Удалено: емой
Удалено: ¶
Удалено: Пример
Удалено: Для г
Удалено: а
Удалено: на рис.3.1
Удалено: называется
Удалено: -
Удалено: -
Удалено: -

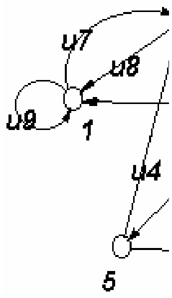


Рис.

Удалено: ¶

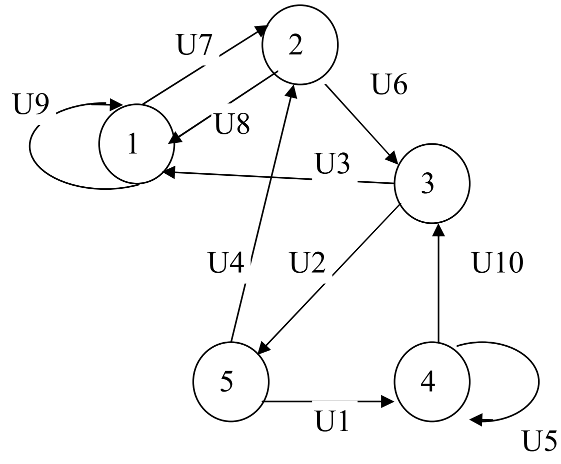


Рис.3.1

Теорема. *В графе число вершин с нечетной степенью четно*.

Удалено: ¶ ¶

Доказательство основано на том, что любая дуга связана с двумя вершинами, значит, сумма степеней всех вершин вдвое больше числа вершин, т.е. всегда четна.

Удалено: четно

Для подмножества вершин $'A \subseteq A$ множество дуг, исходящих из $'A$, обозначают $'A^-$,а множество дуг, заходящих в $'A$ - $'A^+$.

В рассматриваемом выше примере для $'A=\{4,5\}$ $'A^-=\{U_4,U_{10}\}$, $'A^+=\{U_2\}$.

Последовательность дуг, в которой конец каждой предыдущей дуги совпадает с началом последующей дуги, называют *путем* между вершиной – началом первой дуги (*началом* пути) и вершиной – концом последней дуги (*концом* пути). Число дуг в последовательности принимается за *длину* пути.

Удалено: -

Удалено: -

В приведенном примере путь между 1 и 4 вершинами состоит из дуг U_7, U_6, U_2, U_1. Между этими же вершинами существует путь U_7,U_8, U_7, U_6, U_2, U_4, U_6, U_2, U_1. Первый путь имеет длину 4, второй – 9.

Удалено: -

Удалено: длину

Путь назовем *простым*, если в нем никакая дуга не повторяется дважды, иначе путь будет *составным*. В примере первый путь - простой, второй - составной. Путь назовем *элементарным*, если в нем никакая вершина не встречается дважды. Любой составной путь не является элементарным, простой путь может быть как элементарным, так и неэлементарным.

Удалено: назовём

Удалено: *составным*

Удалено:

Путь, в котором начало и конец совпадают, называется *контуром*. Для контура используются те же понятия, что и для пути: контур может быть простым или составным, элементарным или неэлементарным. Начальная (она же и конечная) вершина контура считается входящей только один раз, хотя в записи она будет встречаться дважды. В примере путь U_7, U_6, U_3 является контуром, так как он начинается и кончается в вершине 1.

Удалено: и

Контур единичной длины называется *петлей*. В примере дуга U_9 образует петлю.

Граф называется сильносвязанным, если любая пара вершин в нем связана путем с началом в первой и концом во второй вершине.

Приведенный в примере граф является сильносвязанным. Если сменить ориентацию дуги (3,5) на противоположную, то полученный граф сильносвязанным не будет. В нем ни одна вершина не будет связана путем с вершиной 5.

Граф называют *полным*, если он содержит все возможные дуги. Для полного графа матрица смежности будет содержать единицы во всех клетках. Полный граф обозначается как K_n , где n – мощность множества A.

Удалено: и
Удалено: таблицы
Удалено: -

Введем еще одну трактовку графа. Будем считать, что U описывает отображение множества A в себя. Тогда множество вершин, связанных с подмножеством $'A \subseteq A$ по исходящим дугам (конечные вершины дуг из $'A^-$), будет образом множества $'A$ (обозначается $U('A)$), множество вершин, связанных по заходящим дугам (начальные вершины дуг из $'A^+$), – прообразом $'A$ (обозначается $U^-('A)$). В примере для подмножества $'A=\{3,\ 5\}$ $U('A)=\ \{1,\ 2,\ 4\}$, $U^-('A)=\{2,4\}$. В зависимости от задачи будем использовать обе эти трактовки: U как множество дуг и U как отображение.

Удалено: -

3.2. Части графа

Для графа $G=<A,U>$ граф $H=<A,U'>$ называется *частичным* графом графа G, если в нем $U' \subseteq U$. Отметим, что частичный граф задан *на всех вершинах* исходного графа.

Граф $P=<A',U''>$ называют *подграфом* графа G, если $A' \subseteq A$ и U'' – подмножество всех дуг из U, заданных на вершинах из A'.

Граф $Q=<A',U^*>$, где $U^* \subseteq U''$, называют *частичным подграфом* графа G.

Если в графе на рис.3.1 из множества его дуг убрать, например, дуги (3,5) и (2,1), то получим частичный граф исходного графа. Если убрать, например, вершину 5 и все связанные с ней дуги (дуги (5,2), (5,4) и (3,5)), то получим подграф исходного графа. И, наконец, если из полученного подграфа убрать, например, дуги (3,2) и (2,1), получим пример частичного подграфа.

Удалено: рассмотренном пр
Удалено: имере
Удалено: , а именно,
Удалено: ,

3.3. Неориентированные графы

Граф называют *неориентированным* или *неорграфом*, если в нем элементы множества U рассматривают как неупорядоченные, т.е. в нем пара (a_i,a_j) неотличима от пары (a_j,a_i). В этом случае элементы множества U называются *ребрами* и на рисунке они изображаются в виде отрезков кривых без стрелок.

Удалено: то есть
Удалено: ,

Аналогом пути в неорграфе является понятие *цепь*. Цепь может быть простой или составной, элементарной или нет. Замкнутая цепь называется *циклом* и вводится аналогично понятию *контур*.

Каждому орграфу однозначно сопоставляется неорграф, если заменить все дуги ребрами, т.е. убрать ориентацию. Если в орграфе есть пары дуг, соединяющие одни и те же вершины в противоположном направлении, то они заменяются одним ребром. Так, неорграф, сопоставленный приведенному в примере графу, будет иметь число ребер

Удалено: то есть

на одно меньше числа дуг, потому что паре дуг (1,2) и (2,1) будет сопоставлено одно ребро.

Термин *цепь* можно ввести и для ориентированного графа, понимая под ней последовательность дуг без учета ориентации. Так, в нашем примере можно говорить о цепи (1,2,5,4).

Удалено: Понятие
Удалено: *и*
Удалено: ней

Граф называется *связным*, если в нем любая пара вершин связана цепью.

Компонентой связности графа называется такой его связный подграф, для которого не существует другого связного подграфа, включающего данный в качестве своего подграфа.

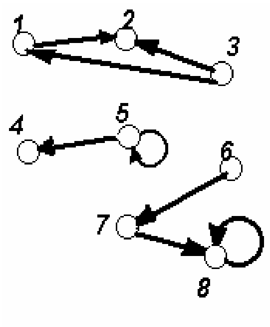


Рис. 3.2

Таким образом, компонента связности есть максимальный связный подграф в графе. На рис.3.2 приведен граф, в котором три компоненты связности: подграфы на вершинах {1, 2, 3}, {4,5}, {6,7,8}.

Теперь можно назвать связным такой граф, который содержит только одну компоненту связности.

Удалено: определить

Ребро графа, удаление которого увеличивает число компонент связности, называется *мостом* или *перешейком*. В нашем примере одним из мостов будет ребро (6,7).

3.4. Расширения модели

Модель графа при описании конкретных объектов, процессов или явлений может быть расширена. Для этого используют следующие способы:

- Взвешивание дуг (ребер). Каждому ребру (дуге) графа приписывается число – вес дуги (ребра). Вес может определять, например, расстояние между городами, если вершинам приписаны имена городов, а ребра – дороги между ними. Для описания взвешенного графа используется матрица смежности, в ячейках которой записаны веса.
- Взвешивание вершин. Аналогично дугам веса приписываются вершинам. Например, вершинами представлены магазины и склады, а вес вершины определяет количество некоторого товара на складе или в магазине.
- Взвешивание дуг и/или вершин векторами. Взвешивание производится не числом, а набором чисел. Например, для дороги могут быть указаны расстояние, стоимость проезда, время в пути и т.д.

Удалено: следующим образом.
Удалено: В
Удалено: -
Удалено: обозначать
Удалено: м
Удалено: -
Удалено: в ячейках
Удалено: Если особо не оговорено, то под взвешенным графом будем понимать такой граф.
Удалено: вершинам
Удалено: составлены
Удалено: тогда
Удалено: веса
Удалено: могут обозначать, например,

- В графе допускается между двумя вершинами несколько «параллельных» дуг (или рёбер). В этом случае говорят о кратных дугах (рёбрах), а такие графы называют *мультиграфами*. Для описания мультиграфов используются такие же таблицы, как и для простых графов, но в клетках записаны не 0 и 1, а кратность дуги.
- В графе используется не бинарное, а r-арное отношение, где $r > 2$. Такие графы называются *гиперграфами*. Для представления этих графов на плоскости вершины, которые относятся к одному ребру, объед иняются замкнутыми линиями, как на рис. 3.3. Здесь граф имеет три ребра: (1, 2, 3), (1, 2, 4), (4, 5, 6). К таким графам приходят при описании логических сетей, когда элементы находятся в отношении, если они имеют полюса, связанные общей цепью.

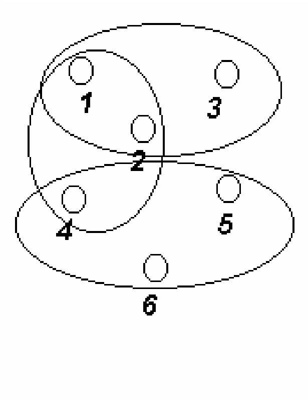


Рис. 3.3.

3.5. Оптимизационные задачи на графах

Большое количество практических задач формулируются как задачи поиска фрагментов графа или каких-то его характеристик, причем существует множество вариантов решения. Каждое решение оценивается числом, и среди множества решений нужно найти такое, для которого оценка имеет экстремальное значение - минимальное или максимальное. Чаще всего в качестве оценок используется сумма весов дуг или ребер, входящих в решение, - тогда оценка называется *аддитивной*, или произведение весов - тогда говорят о *мультипликативной* оценке. Наиболее часто ограничиваются случаем, когда веса дуг являются неотрицательными целыми числами.

3.5.1. Поиск путей в графе

Пусть в графе заданы две вершины $a_н$ и $a_к$, названные соответственно начальной и конечной. Выделим несколько задач, связанных с поиском путей в графе:

- найти путь между начальной и конечной вершиной. Эта задача называется задачей поиска пути в лабиринте;
- найти минимальный путь между заданными вершинами;
- найти максимальный путь;
- найти цикл Эйлера.

3.5.1.1. Кратчайшие пути

Рассмотрим задачу поиска минимального пути между двумя заданными вершинами $a_н$ и $a_к$ в графе при условии, что каждой дуге *(i,j)* сопоставлен вес $c_{i,j}$-неотрицательное число и оценка аддитивна. Если веса обозначают длину дуги, то задача формулируется как задача нахождения кратчайшего пути между заданными вершинами.

Рассмотрим классический алгоритм решения этой задачи - алгоритм Дейкстры, в основе которого лежит следующий

тезис Дейкстры: если кратчайший путь проходит через вершину a_i, то длина части пути от $a_н$ до a_i должна быть минимально возможной.

Алгоритм представим следующей последовательностью шагов.

1. *Начальные присваивания*. Каждой вершине, кроме начальной, сопоставим вес $l(a_i)$, равный бесконечности, назовем этот вес *временным*. Начальной вершине сопоставим вес, равный нулю: $l(a_н)=0$, назовем этот вес *постоянным*, вершину $a_н$ назовем *текущей* и обозначим как *p* .

Удалено: Пункт

2. *Обновление весов*. Всем вершинам, связанным с текущей по исходящим дугам и имеющим временные веса, изменим веса по правилу $l(a_i)=min(l(a_i), l(p)+c_{p,j})$

Удалено: Пункт 2

3. *Смена текущей вершины*. Среди вершин с временной оценкой найдем вершину с минимальным весом и назовем ее текущей, а ее вес - постоянным. Если это есть вершина $a_к$, то перейдем к пункту 4, иначе - к пункту 2.

Удалено: Пункт 3

4. *Выделение пути обратным ходом*. Определим конечную вершину как текущую; для каждой вершины, связанной с ней по заходящим дугам, определим разность между весом текущей вершины и весом дуги. Вершину, вес которой совпадает с этой разностью, назовем текущей, а дугу отнесем к пути. То, что такая вершина всегда найдется, гарантируется способом построения весов вершин. Возможно, что таких вершин будет несколько, что говорит о существовании нескольких решений задачи. Выберем в этом случае любую из них. Повторим эту процедуру до тех пор, пока текущей не станет начальная вершина. В результате множество отнесенных к пути дуг дадут искомое решение.

Удалено: Пункт 4

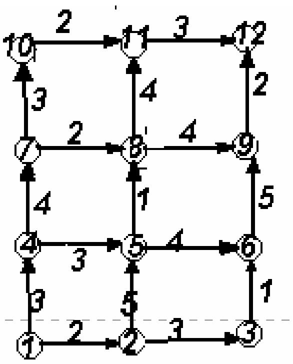


Рис. 3.4

Пример. Заданный граф приведен на рис. 3.4. Ход решения представим таблицей 3.2.

Удалено: Пример

Выполняя шаг 4 алгоритма, выделим путь 12-9-6-3-2-1. Но существует еще один путь той же длины-12-9-8-5-4-1, так как для вершины 9 имеем два одинаковых по длине пути к началу, проходящих через вершины 8 и 6.

Легко видеть, что решением задачи по алгоритму Дейкстры всегда будет путь минимальной длины.

3.5.1.2. Нахождение максимального пути

Таблица 3.2

p	a_i	Вес
1	2 4	2 3
2	3 5	5 7
4	5 7	6 7
3	6	6
5	8	7
6	9	11
7	10	10
8	11	11
9	12	13
11		
12		

Удалено: ¶ ¶ ¶ ¶

Путь максимальной длины есть смысл искать только в графах, где нет контуров. Действительно, в графе с контурами решение однозначно и равно бесконечности, ибо существует путь, бесконечное число раз проходящий по этому контуру. Одной из задач, где используется поиск максимального пути, является задача анализа сетевого графика. Дадим необходимые определения.

Семантика задачи. Сопоставим каждой дуге работу. Вес дуги - время выполнения работы. Вершине сопоставим событие, состоящее в том, что все работы, приписанные заходящим в нее дугам, выполнены, и можно начинать все работы, приписанные исходящим дугам. Граф с такой трактовкой называется *диаграммой ПЕРТ* (от *Program Evaluation and Review Technique*) или *сетевым графиком*

Удалено: а
Удалено: а
Удалено: а
Удалено: а

В сетевом графике максимальному пути, называемому *критическим*, будет соответствовать минимальное время, необходимое для выполнения всех работ. Для сокращения общего времени можно рассматривать, например, вопрос об автоматизации работ на критическом пути с целью сокращения времени их выполнения.

Для решения задачи поиска максимального пути можно воспользоваться алгоритмом Дейкстры, изменив его соответствующим образом. Однако задано добавочное условие на граф задачи: в нем нет контуров. Добавочные условия могут потребовать проверки их выполнения и усложнить алгоритм, но можно использовать эти условия и для сокращения алгоритма, что мы и сделаем.

Удалено: ¶
Удалено: ¶

Справедлива следующая теорема: *если в графе нет контуров, то его вершины можно перенумеровать таким образом, что всякая дуга идет из вершины с меньшим номером в вершину с большим номером и при этом возможна последовательная нумерация вершин от 1 до n=|A|.*

Будем использовать широко распространенный в программировании прием для сокращения решения комбинаторных задач - вначале тратятся добавочные усилия на предварительное упорядочение, затем в упорядоченном множестве более просто находят решение.

1. *Упорядочим множество вершин* так, чтобы любая дуга исходила из вершины с меньшим номером и заходила в вершину с большим номером. Это можно сделать так: присвоим начальной вершине номер 1. Всем вершинам *A*', связанным с начальной по исходящим дугам,

Удалено: Пункт 1

сопоставим номера от 2 до $|A'|$. После этого проверим номера этих вершин на выполнение условия. Для любой пары вершин, где условие не выполняется, переставим номера вершин. Затем рассматриваются вершины, связанные по исходящим дугам с множеством A' и т. д.

2. Присвоим всем вершинам вес $l(i)=0$.

3. Последовательно для вершин 1, 2, 3, ... проведем пересчет весов по формуле $l(i) = \max (l(i), l(j)+C_{ji})$, где максимум берется по всем вершинам *j,* из которых есть дуги в вершину *i*. Это можно сделать, так как ко времени пересчета вершины *i* веса всех вершин *j* вычислены раньше, ибо $i>j$. Сокращение вычислительных затрат по сравнению с алгоритмом Дейкстры связано с тем, что отпадает необходимость на каждом шаге определять очередную вершину с минимальным весом для продолжения расчетов.

4. Как и в алгоритме Дейкстры, выделим обратным ходом искомый путь.

Пример. Пусть исходный граф имеет вид рис. 3.5. На рисунке показана ситуация после выполнения первого шага, когда произведена перенумерация вершин. В табл. 3.3 указаны веса вершин после выполнения второго и третьего шагов алгоритма.

Решение находится в соответствии с шагом 4, критический путь проходит через вершины 11, 10, 7, 6, 3, 2, 1.

В сетевых графиках существует целый ряд задач, решение которых сводится к поиску максимального пути между заданными вершинами. Так, определяются наиболее ранний ($t_{рд}$) и наиболее поздний ($t_{пд}$) допустимые сроки свершения некоторого события, резерв времени у заданной работы, лежащей не на критическом пути. Этот резерв определяет, на какой срок можно сдвинуть выполнение работы, чтобы это не повлияло на время выполнения всей работы ($R_о$–общий резерв) или на условия выполнения следующих работ ($R_{св}$–свободный резерв). Для работ на критическом пути резервы времени равны нулю.

Для рассматриваемого примера можно определить, что выполнение работы, сопоставленной дуге (5,8), можно задержать на 8 единиц или увеличить на эту величину, и общее время выполнения работы останется прежним. Здесь максимальный путь от вершины 8 до конечной равен 18, от начальной вершины до вершины 5 равен 8, время выполнения работы (5,8) равно 3. Значит, максимальный путь через эту дугу равен 18+8+3=29, что на 8 меньше критического.

Более сложной задачей является задача сокращения общего времени выполнения работы, если можно часть работы передать с критического пути на некритический. Если вес дуги на критическом пути может быть уменьшен за счет увеличения веса дуги, не принадлежащей критическому пути (работа «передана» другим исполнителям), то как за счет этого уменьшить общий срок выполнения работы? Эту и ряд других специальных задач рассматривать не будем.

Удалено: Пункт 2
Удалено: Пункт 3
Удалено: а
Удалено: С
Удалено: Пункт 4
Удалено: Пример
Удалено: ¶

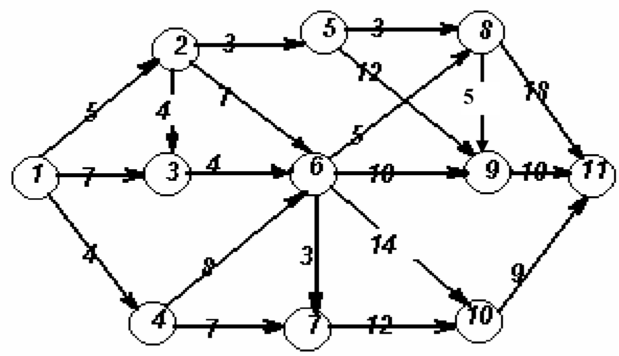


Рис. 3.5

Таблица 3.3

Вершина	Вес
2	5
3	9
4	4
5	8
6	13
7	16
8	18
9	23
10	28
11	37

Удалено: может быть уменьшен за счет увеличения веса дуги, не принадлежащей критическому пути (работа «передана» другим исполнителям), то как за счет этого уменьшить общий срок выполнения работы? Эту и ряд других возможных задач рассматривать не будем.¶

3.5.1.3. Циклы Эйлера

Под циклом Эйлера понимается простой цикл из всех ребер графа.

Цикл Эйлера может в графе не существовать, но если он существует, то весь граф может быть нарисован на бумаге без отрыва карандаша от бумаги таким образом, что по каждому ребру карандаш пройдет ровно один раз и начальная вершина совпадёт с конечной. Условие существования цикла Эйлера определяется следующей теоремой.

Теорема. Цикл Эйлера существует тогда и только тогда, когда в нем нет вершин нечётной степени.

Доказательство проводится построением цикла Эйлера. Начиная с любой вершины, будем обходить граф и удалять пройденное ребро для того, чтобы исключить его из рассмотрения. При этом будем пользоваться одним правилом: не будем ходить по мосту, если можно идти по другому ребру. Тогда мы не попадем в ситуацию, когда после удаления моста граф распадется на несколько компонент связности и не будет возможности перейти из одной компоненты в другую. В этой ситуации все вершины обойдены не будут.

При обходе, войдя в любую вершину, всегда есть возможность выйти из неё по другому ребру, так как степень всякой вершины - чётное число. Если условие не выполняется и находится вершина с нечетной степенью, то, однажды войдя при обходе в эту вершину, из нее нельзя выйти, что и определяет необходимость сформулированного утверждения.

3.5.2. Деревья

Определение.

Неориентированным деревом называют граф, удовлетворяющий одному из условий:

- *связный граф без циклов;*
- *связный граф из n вершин и (n-1) ребра;*
- *граф, в котором любые две вершины связаны ровно одной цепью.*

Задача. Докажите, что все три условия описывают один и тот же объект и что из того, что граф обладает одним из этих свойств, следуют два других.

Ориентированным деревом называют связный граф, в котором в каждую вершину, кроме одной, называемой корнем дерева, заходит ровно одна дуга. В корень дерева ни одна дуга не заходит.

Вершины, из которых не выходит ни одна дуга, называются *листьями*.

Задача. Покажите, что если ориентированному дереву сопоставить неорграф, получим неориентированное дерево.

На рис.3.6 приведен граф – ориентированное дерево. Корнем дерева является вершина 1, вершины 12, 13, 14, 7, …,19, 20 - листья. Вершины дерева разбиваются по уровням. Корень относят к нулевому уровню, связанные непосредственно с ним вершины относят к 1-му уровню и т.д. *К j+1*-му уровню относят вершины, непосредственно связанные с *j-м* уровнем.

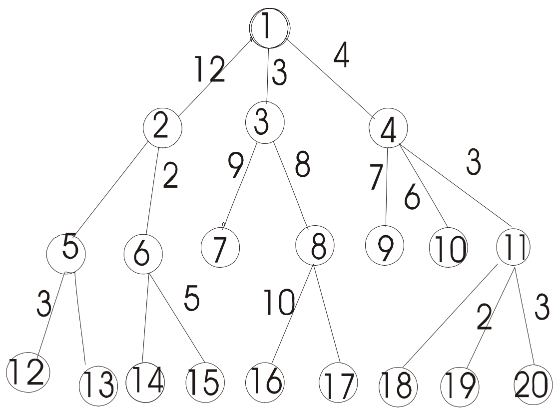


Рис. 3.6

Деревья используются для описания структур организаций, предприятий и др. Такие структуры называются иерархическими. Примером может быть структура управления, где корень дерева - управляющий, с ним связаны непосредственно подчиненные ему руководители - вершины 1-го уровня, которым, в свою очередь непосредственно подчинены другие - вершины 2-го уровня и так вплоть до исполнителей нижнего уровня – листьев. Дерево образует структура предприятия, где корень - само предприятие, под ним - входящие в него цехи и службы, ниже - входящие в цехи участки и т.д. Принято дерево изображать именно так, как приведено на рис.3.6, корнем вверх.

3.5.2.1. Дерево решений

В жизни мы постоянно сталкиваемся с проблемой принятия решения из множества возможных решений - с проблемой выбора. В результате выбора мы попадаем в новую ситуацию, когда снова нужно делать выбор.

Возможности выбора при решении проблемы можно представить в виде дерева, где в корне – проблема, дуге соответствует один из вариантов выбора, вершине - новая ситуация, возникающая в результате реализации приписанного дуге варианта. Такой трактовке соответствует граф типа дерева, получивший название «дерево решений». Предположим, что можно оценить эффективность принятого выбора. Тогда возникает задача поиска среди возможных путей от корня (когда проблема поставлена) к одному из листьев (когда проблема решена) пути, имеющего оптимальную оценку.

Удалено: <sp><sp>

Стратегии поиска по дереву решений.

Предположим, что дерево решений очень велико, поэтому поиск путём полного перебора всех возможных решений невозможен. Рассмотрим стратегии поиска, связанные с сокращённым перебором решений: метод ветвлений и метод ветвей и границ.

Метод ветвлений. Поиск начинается от корня дерева. Выбирают минимальную по стоимости исходящую дугу и по ней переходят к следующей вершине. Если эта вершина не является листом, снова переходят по минимальной исходящей дуге, пока не будет достигнут лист. Алгоритм очень просто реализуется, но очевидно, что при этом решение может быть далёким от минимального.

Пример. Определим по методу ветвления путь в дереве на рис.3.6. Путь пройдет по вершинам 1,3,8,16 и равен 21.

Удалено: Пример

Метод ветвей и границ. На каждом шаге выделяется множество вершин в качестве границы, веса вершин которой оцениваются длиной пути до неё от корня, на границе находится вершина с минимальной оценкой. Если эта вершина является листом, то решение найдено – путь до этого листа и будет минимальным решением, если нет, то строится новая граница заменой найденной вершины на вершины, связанные с ней по исходящим дугам, и поиск решения продолжается. Метод гарантирует получения минимального решения, однако часто связан с большим объёмом вычислений.

Рассмотрим шаги построения по этому методу на примере дерева, приведённого на рис.3.6

На первом шаге границей является корень дерева, который заменяется вершинами, связанными с ним по исходящим дугам – вершинами 2(12), 3(3), 4(4). Здесь в скобках приведены веса, приписываемые вершинам. Среди вершин границы выбирается вершина 3, имеющая минимальный вес. Эта вершина не является листом, она удаляется из границы, а вместо неё в границу вводятся вершины 7, с весом, равным 12 (сумма весов вершины 3, равной 3, и дуги (3,7), равной 9) и 8 с весом 11.

Новая граница: 2(12), 7(12), 8(11), 4(4). На ней выбирают вершину 4, которую заменяют вершинами 9,10 и 11, что приводит к новой границе: 2(12), 7(12), 8(11), 9(11), 10(10), 11(7). Для продолжения берётся ветвь, связанная с вершиной 11. Она удаляется из границы, и вместо неё вводятся связанные с ней по исходящим дугам вершины 18, 19, 20. Результатом будет новая граница из вершин 2(12), 7(12), 8(11), 9(11), 10(10), 18(11), 19(9), 20(10). Выбирается новая ветвь с минимальным весом – вершина 19, а она есть лист. Решение найдено. Им является путь в вершине 19, вес решения равен 9.

3.5.2.2. Поиск минимального остова

Остовом графа G называется его частичный граф типа дерево. Таким образом, остов выделяется из графа *G* удалением из него ребер до тех пор, пока он не станет графом типа дерево, включающим все вершины.

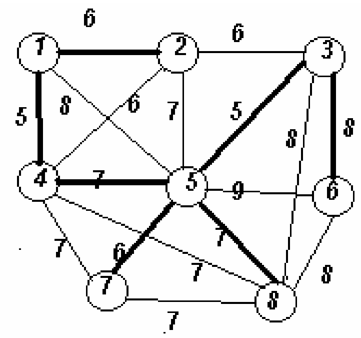


Рис. 3.7

Так, на рис. 3.7 приведен граф, в котором толстыми линиями выделен один из его остовов. В графе может быть несколько различных остовов.

Если ребра графа взвешены, то возникает задача выделения остова с минимальной или максимальной оценкой.

Семантика задачи. Предположим, что вершинам графа сопоставлены полюса схемы, на которые необходимо подать питание, а ребрам - разрешенные связи для цепи питания. Тогда любой из остовов будет определять вариант цепи питания.

Действительно, цепи сопоставляется граф без петель, включающий все вершины. Если веса определяют расстояние между полюсами, то остову с минимальной суммой весов соответствует разводка питания с минимальной суммарной длиной проводников. Такая задача решается при трассировке печатных плат.

Может быть предложена такая трактовка задачи. Вершины - абоненты, связанные линиями связи, веса на ребрах - оценка потери конфиденциальной информации при ее передаче по этой линии связи. Тогда задаче выделения остова с минимальным произведением весов, входящих в остов ребер, соответствует задача определения наиболее надежной сети для передачи информации.

Рассмотрим два алгоритма решения задачи: жадный алгоритм и алгоритм Прима.

Жадный алгоритм.

Выбирается ребро, имеющее минимальный вес среди всех рёбер, и включается в остов. Из оставшихся ребер выбирается снова то, которое имеет минимальный вес, и включается в остов, если при этом не образуются циклы. Процесс продолжается до тех пор, пока все вершины не будут включены в остов.

Алгоритм прост для понимания и обеспечивает получение минимального решения. Однако сложность его состоит в том, что в нем неявно присутствует процедура проверки на появления циклов, которая связана с перебором по всему графу, так же как и поиск очередного ребра.

Поиск по этому алгоритму в графе на рис.3.7 приведет к решению, которое представим в виде последовательности просмотренных ребер: (1,4), (3,5), (1,2), [(2,4)], (2,3), (5,7), [(2,5)], [(4,5)], [(4,7)], (4,8), [(5,8)], [(7,8)], (3,6). В квадратные скобки заключены те ребра, которые не включены в остов из-за появления циклов.

Алгоритм Прима.

1. Включим любую вершину в остов.

Удалено: Пункт 1

2. Рассмотрим все ребра, исходящие из вершин, включенных в остов к оставшимся вершинам, и из них выберем ребро с минимальным весом. Его концевую вершину включим в остов. Повторяем этот пункт, пока не все вершины включены в остов.

Удалено: Пункт 2

В этом алгоритме не нужно следить за образованием циклов. Алгоритм начинает работу с произвольно выбранной вершины. Кроме того, на каждом шаге просматривается только одна строка матрицы смежности.

Задача. Докажите, что независимо от выбора начальной вершины получается минимальное решение.

Если для графа на рис 3.7 алгоритм начать с вершины 7 , то последовательность действий будет такой:

Пункт 1. Вершину 7 отнесем к остову.

Пункт 2. Оценим ребра от вершины 7 к 4, 5 и 8. Выберем ребро с минимальным весом (7,5) и вершину 5 отнесем к остову.

Пункт 2. Для вершин 5, 7 рассмотрим ребра в вершины 1, 2, 3, 6, 8, 4 и по минимальной стоимости ребра отнесем вершину 3 к остову.

Пункт 2. Из вершин 5, 7, 3 ребра идут к вершинам 1, 2, 4, 6, 8. По наименьшему из этих ребер в остов отнесем вершину 2.

Пункт 2. Для вершин 5, 7, 3,2 рассматриваем ребра в вершины 1, 4, 6, 8.
По минимальному ребру отнесем к остову вершину 1.

Пункт 2. Для вершин 5, 7, 3, 2, 1 по минимальному исходящему из них ребру выберем вершину 4.

Пункт 2. Для вершин остова 1, 2, 3, 4, 5, 7 по минимальному исходящему из них ребру выделим вершину 6.

Пункт 2. Для вершины 8 определим связывающее ее ребро с вершиной или 4, или 7, или 5.

Решение построено.

3.5.2.3. Деревья Штейнера

Пусть на двумерной плоскости расположены вершины графа, связанные ребрами. Ребро взвесим расстоянием между вершинами, связанными этим ребром. Допустим возможность вводить произвольным образом добавочные вершины, называемые точками Штейнера, и заменять ребра цепью из ребер, проходящей через точки Штейнера. Задачу выделения остова в таком расширенном графе называют задачей Штейнера, а выделенный остов – деревом Штейнера (по фамилии математика, занимающегося этой задачей).

На рис.3.8,а приведен граф из четырех вершин, для которого нужно построить остов, объединяющий все вершины. Минимальный остов выделен жирными линиями. Как видно из рис. 3.8,б, если ввести в этот граф точку Штейнера – вершину 5, то выделенный остов в этом графе будет короче.

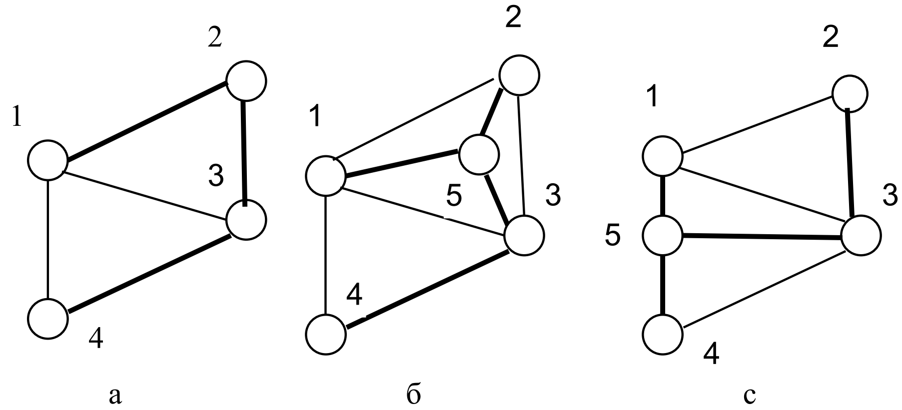


Рис. 3.8

Задача Штейнера имеет очевидную инженерную интерпретацию: вершинам сопоставляются эквипотенциальные полюса сети, например полюса, на которые должно быть подано питание. Ребрам соответствуют допустимые способы связи между полюсами. Тогда решению будет соответствовать электрическая цепь минимальной длины, объединяющая все эти вершины. Точки Штейнера - точки «разветвления» в цепи, известны в инженерной практике как «Т-цепи».

В зависимости от метрики пространства, в котором рассматривается задача, возможны следующие задачи Штейнера.

Пусть ребром связаны вершина 1 с координатами (x_1, y_1) и вершина 2 с координатами (x_2, y_2). Если расстояние между вершинами 1 и 2 (длина дуги (1,2)) определяется, как $L(1,2)= \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$, то задача называется *евклидовой*.

Если расстояние определяется как

$L(1,2)=|x_1-x_2| + |y_1 - y_2|$,

то задача называется *линейной*.

С линейной метрикой имеют дело при работе с печатными платами, где связь возможна только в решетке по взаимно перпендикулярным направлениям.

Для евклидовой задачи установлено: вершины Штейнера имеют степень три и располагаются в таком месте, что углы между инцидентными ребрами равны 120°.

Если пространство линейное, то в качестве мест для расположения точек Штейнера достаточно рассмотреть только точки пересечения вертикальных и горизонтальных линий, проходящих через вершины графа. Это резко сокращает число возможных вариантов. В примере для линейного случая рис.3.8,в точка Штейнера – вершина 5.

3.5.3. Раскраска графа

В неорграфе $G(A,U)$ взвесим вершины целыми положительными числами и потребуем при этом, чтобы смежным вершинам были сопоставлены разные числа. Если число трактовать как номер краски (цвета), то такое взвешивание вершин называют раскраской графа.

Раскраска графа, в которой используется минимальное число красок, называется *минимальной*. Число красок минимальной раскраски является одной из характеристик графа и называется *хроматическим* числом.

К задаче определения минимальной раскраски графа сводятся многие задачи. В качестве примера назовем задачу нахождения минимального числа внутренних переменных в программе. При построении программ необходимо вводить добавочные временные переменные, например, переменную - счетчик цикла в операторе `for (int i=1;i<10;i++)`.

Если вершинам сопоставить вхождение переменной в программу, а ребром обозначить зависимость одной переменной от другой, то идентификатору переменной будет соответствовать краска. Действительно, зависимые переменные должны иметь разные имена. Тогда нахождение минимального числа внутренних переменных сведётся к минимальной раскраске вершин графа. В этой трактовке задача рассматривалась А.П. Ершовым, одним из крупнейших теоретиков и практиков программирования, предложившим интересный эвристический алгоритм решения этой задачи.

Удалено: ¶

Алгоритм раскраски А.П. Ершова

Удалено: ¶

Введем понятие расстояния между вершинами как длину минимального пути между ними. Назовем множество вершин на расстоянии единицы от *a* *1*-окрестностью вершины *a*, *p*-окрестностью вершины *a*- множество вершин на расстоянии *p* от *a*.

Формат: Список

1. Выберем вершину и присвоим ей первую краску.
2. Выберем из *2*-окрестности этой вершины любую вершину и присвоим ей ту же краску. Объединим эти две вершины в одну так, что все ребра, связывающее не раскрашенные вершины с этими вершинами, заменяются ребрами к этой объединенной вершине.
3. Для полученного графа находим новую нераскрашенную вершину из *2*-окрестности, если таковая есть, красим ее в ту же краску и объединяем с объединенной вершиной.
4. Пункты 2 и 3 выполняются до тех пор, пока существуют нераскрашенные вершины в *2* - окрестности объединенной вершины.
5. Затем из числа нераскрашенных вершин выбирается новая вершина и для нее процедура раскраски повторяется, и так до тех пор, пока все вершины не будут раскрашены.

Алгоритм достаточно просто реализуется, если граф представлен в виде матрицы смежности, однако приведены примеры, когда решение имеет не минимальное количество красок. Алгоритм А.П. Ершова относится к так называемым эвристическим алгоритмам, построенным на основе некоторых разумных с точки зрения здравого смысла методах, для которых гарантий оптимальности нет. В алгоритме Ершова исходят из того, что нужно в одну и ту же краску покрасить как можно больше вершин. Для этого предлагается выбирать следующую вершину для раскраски на минимально возможном расстоянии от уже покрашенных в ту же краску вершин, чтобы обеспечить больше простора следующим шагам выбора.

Если хроматическое число графа равно двум, граф называется *бихроматическим*,. В таких графах все вершины делятся на два непересекающихся подмножества $A1$ и $A2$, в каждом из которых элементы не связаны между собой. Такие графы называют ещё *двудольными* и обозначаются как $G=<A1,A2,U>$, где $A1\cap A2=\varnothing$, $U \subseteq A1 x A2$.

К задачам о двудольных графах сводится большое число задач, называемых задачами о назначениях. К ним относятся задачи:

- распределения множества задач по процессорам в многопроцессорной системе;
- распределения деталей по обрабатывающим центрам в гибких автоматизированных системах;
- назначения сотрудников на должности;
- распределения товаров со складов по торговым точкам.

Во всех этих задачах исходное множество состоит из двух различных по смыслу подмножеств, а связи могут быть только между элементом одного и элементом второго из этих подмножеств.

Свойство бихроматического графа определяет следующая теорема.

Теорема. *Граф является бихроматическим тогда и только тогда, когда он не содержит циклов нечётной длины*.

Необходимость доказывается просто: если в графе находится цикл нечётной длины, раскраска двумя красками его вершин невозможна. Достаточность доказывается сложнее и здесь не приводится.

3.5.4. Паросочетания

Паросочетанием в графе называется такое подмножество его рёбер, в котором никакая пара рёбер не смежна.

Интерес представляют две задачи, связанные с нахождением паросочетаний. Если паросочетание оценивать числом входящих в него ребер, то *наибольшим* называют паросочетание, имеющее наибольшее число ребер.

Если ребрам приписаны веса, то *максимальным* называют паросочетание, для которого сумма весов входящих в него ребер максимальна.

Алгоритм нахождения наибольшего паросочетания использует следующие понятия.

Назовём вершину в графе *связанной*, если она принадлежит паросочетанию, иначе вершину назовём *свободной*.

Цепь назовём последовательной, если в ней последовательно повторяются ребра, принадлежащие и не принадлежащие паросочетанию.

Если все рёбра паросочетания можно включить в последовательную цепь, в которой начальная и конечная вершины свободные, то найдётся паросочетание, в котором будет число ребер на единицу больше. Оно получается, если в выделенной цепи все рёбра, не входящие в паросочетание, включить в него, а все входящие – исключить.

Алгоритм основан на этом утверждении. Строится последовательная цепь и описанной процедурой увеличивается число рёбер в паросочетании, пока это возможно.

Пример. Рассмотрим граф, приведённый на рис 3.9. Выделим в нём последовательную цепь, например, 1,2,3,5,10,11,7,12,8,4, в которой рёбра (1,2), (3,5),(10,11),(7,12),(8,4) включены в паросочетание.

Удалено: Пример

Для этой цепи ищем цепь, в которую пробуем включить свободные вершины 6 и 9 так, чтобы они были начальной и конечной вершинами цепи. Такая цепь находится: (9, 10, 11, 7, 12, 8, 4, 3, 5, 1, 2, 6). Значит, получаем новое паросочетание, в котором число рёбер на единицу больше первого. Так как это паросочетание содержит все вершины, оно является наибольшим. Решением будет множество {(9, 10), (11, 7), (12, 8), (4, 3), (5, 1), (2, 6)}.

Можно убедиться, что это решение не единственное.

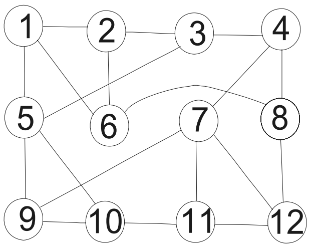


Рис. 3.9.

Паросочетание называют *совершенным*, если в нём участвуют все вершины. Очевидно, что совершенное паросочетание, если оно в графе существует, будет и наибольшим.

Так как с каждым ребром связана пара вершин и рёбра в паросочетании не смежные, то с любым паросочетанием связано чётное число вершин. Значит, имеет место теорема:

Необходимым условием существования в графе совершенного паросочетания является чётное число его ребер.

Удалено: Теорема

3.5.5. Паросочетания в двудольном графе

Задача нахождения максимального или наибольшего паросочетания в двудольном графе $G = <A \cup B, U>$, $U \supseteq AxB$ называется *задачей о назначениях.*

Не теряя общности, можно предположить, что $|A|=|B|$. Если это не так, то всегда можно меньшее множество дополнить элементами, не связанными с другими элементами.

Будем решать следующую задачу: найти в двудольном графе максимальное совершенное паросочетание.

Рассмотрим задачу в следующей семантике. Множеству A сопоставляется множество работников, множеству B – множество работ, ребра взвешены числами (целыми и положительными), определяющими эффективность назначения работника на работу. Число работников равно числу работ. Нужно найти такое назначение, когда *все* работники назначены на работы и *все* работы заняты и при этом достигается максимальная эффективность использования работников.

Условие существования совершенного паросочетания в двудольном графе определяется *теоремой Кёнига-Холла*:

Совершенное паросочетание существует тогда и только тогда, когда для любого $A' \subseteq A$ *имеет место* $|A'| \le |U(A')|$

Метод выделения максимального паросочетания заключается в следующем (этот метод в литературе называют *венгерским*).

Взвесим вершины графа весами x_i для вершин множества A и y_i для вершин множества B таким образом, чтобы условие $x_i + y_i \ge c_{ij}$ удовлетворялось для любого ребра. Если в решении оставлять только те ребра, для которых выполняется равенство

$$x_i + y_i = c_{ij} \qquad (*),$$

и обеспечить подбором x_i и y_i выполнение условия теоремы Кёнига-Холла, то при этом сумма $\mathbf{c_{ij}}$ в паросочетании (которое существует по теореме Кёнига-Холла) будет максимально возможной, сумма всех x_i и y_i –

минимально возможной, так как первая сумма ограничивает вторую сверху, вторая сумма ограничивает первую снизу. В силу условия (*) при этом $\Sigma c_{ij}=\Sigma x_i + \Sigma y_i$

Начинается расчет с того, что значению $x_\mathbf{i}$ приписывается max $\boldsymbol{c_{ij}}$ по всем j, а $y_\mathbf{i}$ –нулевые значения.

Удалено: а

Построим граф претендентов, содержащий только те ребра исходного графа, для которых выполняется равенство (*).Для графа претендентов проверяем условие теоремы Кёнига-Холла. Если находится подмножество *A*', для которого условие теоремы не выполняется, то для каждого его элемента вес $\mathbf{x_i}$ уменьшим на единицу, а вес каждого элемента из *U*(*A*') увеличим на единицу. Если появляются новые ребра, для которых справедливо условие (*), их вносят в претенденты и снова проверяют условия теоремы. Это продолжается, пока не будут выполнены условия теоремы Кёнига-Холла.

В графе претендентов решение находится путём перебора. При этом вначале для всех вершин, степень которых равна 1, т.е. для которых назначения однозначны, производятся назначения, и соответствующие пары вершин из графа удаляются, что сокращает объём перебора.

Удалено: то есть

3.5.6. Поток в транспортной сети

Транспортной сетью называют связный граф без контуров, в котором:

- существует единственная вершина x_0, из которой дуги только выходят, называемая входом сети;
- существует единственная вершина z, в которую дуги только заходят, называемая выходом сети;
- каждая дуга (i,j) взвешена неотрицательным целым числом c_{ij}, называемым пропускной способностью дуги.

Потоком в сети φ называется целочисленная функция, заданная на дугах графа, такая, что:

- для любой дуги $0 \le \varphi_{ij} \le c_{ij}$;
- для любой вершины i, кроме входа и выхода сети, имеет место равенство

$$\sum_{j\in P}\varphi_{ij} = \sum_{j\in Q}\varphi_{ji},$$

где P - множество вершин, связанных с вершиной i по исходящим, Q -по заходящим дугам.

Последнее равенство означает, что суммарный поток, входящий в вершину, равен суммарному потоку, выходящему из неё. Из этого равенства и из того, что в сети нет контуров, следует, что суммарный поток, исходящий из вершины x_0, равен суммарному потоку, заходящему в вершину z, т.е. $\Sigma\varphi_{0j}=\Sigma\varphi_{zi}=\varphi$. Эта величина называется *величиной потока*.

Удалено: то есть

Задача состоит в том, чтобы для заданной сети определить поток, имеющий *максимальную* величину.

Назовем дугу *насыщенной*, если для нее величина потока равна пропускной способности дуги: $\varphi_{ij}=c_{ij}$. Путь между вершинами x_0 и z назовём *полным*, если он содержит хотя бы одну насыщенную дугу. Поток назовем *полным*, если в нём любой путь полный.

Рассмотрим алгоритм решения этой задачи, предложенный Фордом и Фалкерсоном.

Алгоритм Форда и Фалкерсона.

1. Зададим любой поток и доведем его до полного. Для этого перебираем все пути между входом и выходом сети и если некоторый путь неполон, то, увеличивая поток по нему, доводим его до полного.

Удалено: Пункт

2. Проведем последовательную разметку графа:

Удалено: Пункт

- вершину x_0 пометим символом 0;
- если вершина $\mathbf{x_i}$ отмечена, то непомеченные вершины, связанные с ней по исходящим дугам, пометим как +*i,* если дуга не насыщена, и связанные по заходящим дугам как -*i*, если поток в дуге не равен нулю. Если в результате разметки будет помечен выход сети, то переходим к пункту 3, иначе - конец алгоритма.

Формат: Список

Удалено: ;¶
е

3. Разметка выделяет цепь между входом и выходом сети. Будем двигаться по ней от входа к выходу и изменять поток в проходимых дугах по правилу: если направление движения совпадает с направлением дуги, то значение потока увеличим на единицу, если противоположен, то уменьшим на единицу. Так как для дуги цепи, исходящей из входа, и дуги, заходящей в выход сети, направление дуги и направление движения всегда совпадают, то в результате величина потока увеличится.

Удалено: Пункт

После этого переходим к пункту 2.

Можно показать, что в результате такого изменения потока условия, сформулированные в определении потока, нарушены не будут. Действительно, для отмеченной вершины *x* возможные сочетания инцидентных дуг к другим отмеченным вершинам показаны на рис.3.10.

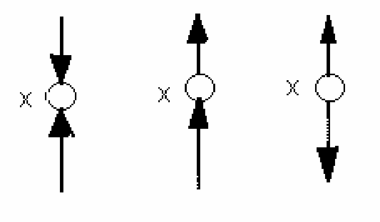


Рис.3.10

Предположим, что при движении по цепи вершина проходится в направлении снизу вверх. Тогда изменение потока по описанной процедуре в первом случае увеличит входящий в дугу поток на единицу по нижней дуге и уменьшит по верхней, величина суммарного потока по входящим дугам не изменится. Во втором случае величина входящего и выходящего потока изменится на единицу, значит, их равенство останется. В третьем варианте величина исходящего потока остается постоянной и равенство потоков не изменится.

Пример. Рассмотрим транспортную сеть, приведенную на рис 3.11.

Удалено: Пример

Здесь цифрами в скобках отмечена величина потока по дугам, без скобок - пропускные способности. Нетрудно убедиться, что поток является полным, т.е. по каждому пути его увеличить нельзя. Жирными линиями выделена цепь, полученная после разметки графа в соответствии с п. 3 алгоритма. По этому пути поток по дугам (*x*1,*x*4), (*x*4,*x*6),(*x*2,*x*7) и (*x*7,*x*8) увеличится на единицу, поток по дуге (*x*2,*x*6) уменьшится на единицу, общий поток увеличится на единицу.

Удалено: то есть

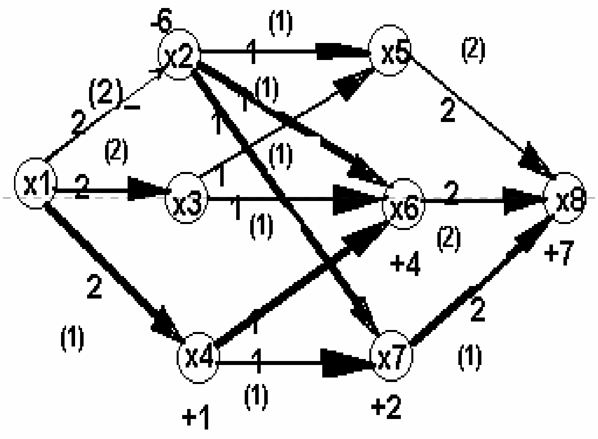


Рис.3.11

Повторная разметка графа невозможна, значит, полученный поток максимален.

Разобьем множество вершин *A* сети на два подмножества, *A*1 и *A*2, при котором в первое подмножество включен вход сети, во второе - её выход. Множество дуг, начинающихся в *A*1 и заканчивающихся в *A*2, называется *разрезом* сети, а сумма пропускных способностей этих дуг называется *величиной разреза*.

Разрез, величина которого минимальна, называется *минимальным* разрезом.

Для проверки того, является ли поток максимальным, служит теорема Форда и Фалкерсона: *максимальный поток в сети равен величине ее минимального разреза.*

Теорема позволяет проверить, является ли поток максимальным. Если находится разрез, в котором все дуги насыщены, то величина потока будет максимальной.

В приведенном примере величина потока после первого шага разметки и изменения потока становится максимальной, так как разрез, когда в одно подмножество включена вершина *x*1, а в другое - все остальные вершины, содержит только насыщенные дуги.

3.5.7. Транспортная задача

Наряду с задачей отыскания максимального потока большое практическое значение имеет задача экономичного распределения потока по дугам транспортной сети, получившая название *транспортной задачи*. Поскольку транспортная сеть во многих случаях представляет собой схему организации перевозок каких-либо грузов, то решение транспортной задачи позволяет определить наиболее рациональный план перевозок, т.е. такое распределение маршрутов, которое обеспечивает, например, минимальную стоимость перевозок или доставку грузов к потребителю в кратчайшее время.

Первая задача получила название *транспортной задачи по критерию стоимости*, а вторая — *по критерию времени*.

Для каждой дуги (x_i, x_j) введём ещё один вес - стоимость прохождения единицы потока по ней - и обозначим его d_{ij}.

Транспортную задачу по критерию стоимости в терминах теории графов можно сформулировать следующим образом. Даны транспортная сеть с наибольшим потоком φ^m и поток φ_z, который должен быть пропущен по этой транспортной сети. Требуется найти такое распределение потока φ_z по дугам, которое обеспечивает минимальную стоимость прохождения потока. При этом для каждой дуги должно выполняться соотношение $\varphi(x_i, x_j) \le c_{ij}$, а стоимость прохождения потока $\varphi(x_i, x_j)$ по дуге (x_i, x_j) равна $d_{ij}\varphi(x_i, x_j)$.

Для решения этой задачи будем рассматривать величины d_{ij} как длины соответствующих дуг. В этом случае стоимость прохождения потока φ по какому-либо пути μ от x_0 до z будет равна произведению длины этого пути на величину потока φ и задача минимизации стоимости прохождения потока сведется к решению рассмотренной ранее задачи нахождения кратчайшего пути в графе от x_0 до z. При отсутствии ограничений на пропускную способность дуг, кратчайший путь является путем, который обеспечивает минимальную стоимость прохождения потока.

Удалено: x

Удалено: x

При наличии ограничений на пропускную способность дуг задача решается в несколько этапов, путем нахождения частичных потоков на каждом этапе. Общий путь решения задачи состоит в следующем.

В графе $G_1 = (X, R)$, изображающем транспортную сеть с длинами дуг $d_{ij} = |(x_i, x_j)|$, ищется кратчайший путь μ_1 от x_0 до z. Пусть c_1 — пропускная способность этого пути. По нему пропускается поток

Удалено: x

$$\varphi_1 = \begin{cases} \varphi_z, \text{ если } \varphi_z \le c_1; \\ c_1, \text{ если } \varphi_z > c_1. \end{cases}$$

Если $\varphi_z \le c_1$, то задача решена и путь μ_1 является наиболее экономичным для потока φ_z.

Если $\varphi_z > c_1$, то поток φ_1 рассматриваем как частичный и переходим к графу G_2, который получается из графа G_1 путем замены пропускных способностей дуг c_{ij} на c'_{ij} из соотношения

$$c'_{ij}=\begin{cases} c_{ij}\text{-}c_1 \ \text{для } u\in \mu_1; \\ c_1 \ \text{для } u \notin \mu_1. \end{cases}$$

При этом дуги, у которых $c'_{ij} = 0$, исключаются из рассмотрения. Поток, распределение которого ищется в графе G_2, принимаем равным $\varphi`_1=\varphi_z\text{-}\varphi_1$.

Теперь возникает первоначальная задача отыскания наиболее экономичного распределения потока φ'_z, но уже по отношению к графу G_2. Ее решение дает путь μ_2 с пропускной способностью c_2, через который пропускается частичный поток

$$\varphi_2=\begin{cases} \varphi`_z, \text{ если } \varphi`_z\le c_2; \\ c_2, \ \text{ если } \varphi`_z> c_2. \end{cases}$$

Если $\varphi'_z \le c_2$, то задача решена и наиболее экономичным распределением потоков в графе G_1 будет прохождение потока φ_1 по пути μ_1 и потока φ_2 по пути μ_2.

Если $\varphi'_z > c_2$, то следует перейти к новому графу G_3 и найти новый частичный поток φ_3 . Этот процесс повторяется до тех пор, пока сумма частичных потоков не достигнет значения φ_z. Эти частичные потоки, пропущенные по графу G_1, и дают оптимальное распределение потока φ_z.

Для иллюстрации описанного метода рассмотрим наиболее часто встречающийся вариант транспортной задачи по критерию стоимости.

Однородный груз имеется на станциях $x_1,...,x_m$ в количествах $a_1,...,a_m$. Его требуется доставить на станции $y_1, ..., y_r$ в количествах $b_1, ..., b_r$. Предполагается, что общее количество требуемого груза равно имеющимся запасам: $\Sigma\, a_i=\Sigma\, b_j$.

Стоимость перевозки груза со станции x_i на станцию y_i равна d_{ij}. Требуется найти наиболее экономичные маршруты перевозки грузов. Исходные данные удобно записывать в виде табл. 3.4.

Таблица 3.4

Удалено: .

	b_1	…	b_r
A_1	d_{11}	…	d_{1r}
…	…	…	…
a_m	d_{m1}	…	d_{mr}

Транспортная сеть для этой задачи строится следующим образом. Вход x_0 соединяется с каждой из вершин x_i дугой с пропускной способностью $c(x_0, x_i) = a_i$. Каждая из вершин y_i соединяется с выходом z дугой с пропускной способностью $c(y_i, z) = b_j$. Стоимость прохождения потока по дугам (x_0, x_i) и (y_i, z) считается равной нулю. Наконец, каждая вершина x_i соединяется с каждой вершиной y_i дугой с бесконечной пропускной способностью, стоимость прохождения единицы потока по которой равна d_{ij}. Далее к этой транспортной сети применяется рассмотренный метод.

Пример. Найти наиболее экономичные маршруты для транспортной задачи, заданной табл. 3.5

Таблица 3.5

	5	10	20	15
10	8	3	5	2
15	4	1	6	7
25	1	9	4	3

Применяя описанный выше метод к транспортной сети, построенной по табл. 3.5 (рис. 3.12), находим частичные потоки и маршруты, перечисленные в табл. 3.6 в порядке их получения.

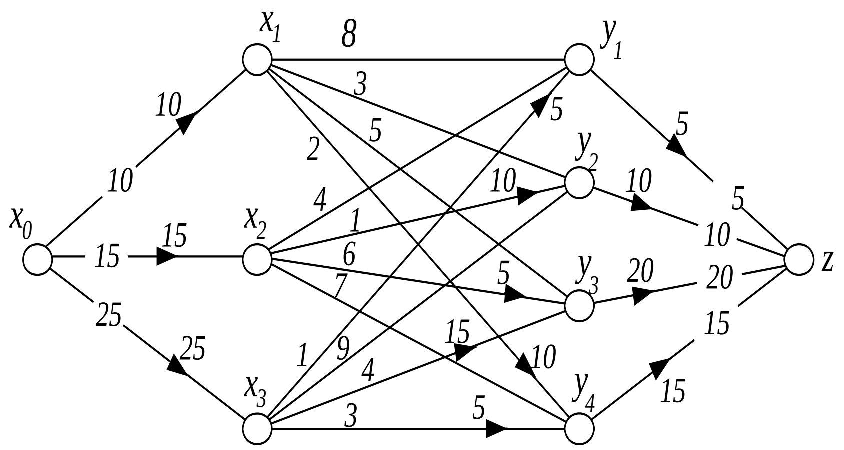


Рис. 3.12

Таблица 3.6

Номер маршрута	Маршрут (x_i, y_j)	Частичный поток φ_k, ед.	d_{ij}	Стоимость перевозки $d_{ij}\,\varphi_k$, ед.
1	(x_3, y_1)	5	1	5
2	(x_2, y_2)	10	1	10
3	(x_1, y_4)	10	2	20
4	(x_3, y_4)	5	3	15
5	(x_3, y_3)	15	4	60
6	(x_2, y_3)	5	6	30
Всего	–	50 45	–	140

Следовательно, разобранный метод решения транспортной задачи дает, по существу, способ нахождения величин частичных потоков $\varphi(x_i, y_j)$, минимизирующих указанную сумму.

Решение транспортной задачи по критерию времени рассмотрим на примере транспортной сети, заданной табл. 3.6, в которой величины d_{ij} будем теперь трактовать как время, необходимое на перевозку груза из пункта x_i в пункт y_j и обозначить далее через t_j. С подобными задачами можно столкнуться три транспортировке скоропортящихся продуктов, при доставке средств помощи в районы стихийных бедствий, при вывозе зерна нового урожая в заготовительные пункты и т. п. Во всех этих задачах стоит требование доставки всех грузов в пункты назначения за возможно более короткий промежуток времени.

Рассмотрим общий путь решения этой задачи. Предположим, что каким-либо методом найдено некоторое распределение потока φ_z в графе *G,* изображающем рассматриваемую транспортную сеть. Выделим из графа *G* частичный граф *G′,* в который включим только дуги, участвующие в передаче потока φ_z. Пусть μ — некоторый путь, ведущий из x_0 в *z,* а t_μ — время прохождения потока по этому пути. Очевидно, что время, необходимое на перевозку всех грузов из x_0 в *z,* будет определяться путем, имеющим наибольшую продолжительность прохождения потока, так как перевозка грузов по остальным путям закончится раньше. Следовательно, время *T,* требуемое на перевозку всех грузов, будет равно $T = \max t_\mu$.

Удалено: а

Удалено: а

Решение транспортной задачи по критерию времени сводится, таким образом, к тому, чтобы выделить из графа *G* такой частичный граф *G',* который был бы способен пропустить весь поток φ_z и в котором длительность наиболее продолжительного пути была бы минимальной по сравнению со всеми другими подобными графами.

Задача решается последовательным улучшением графа *G′* путем удаления из него наиболее продолжительных путей и введения более коротких, но не примененных ранее, и соответствующего перераспределения потока φ_z.

Обратимся к рассматриваемому примеру. В качестве первого приближения к наилучшему решению примем решение, полученное на основе критерия стоимости. Распределение потоков для этого случая показано на рис. 3.12. Из этого рисунка видим, что время прохождения потока по наиболее продолжительному маршруту (x_2, y_3) равно 6. Однако оказались неиспользованными менее продолжительные маршруты (x_1, y_2), (x_2, y_1), (x_1, y_3). Возможно, что использование какого-либо из этих маршрутов позволит исключить маршрут (x_2, y_3).

Построим частичный граф *G′,* в который включим только дуги графа *G,* имеющие $t_{ij} < 6$ и в котором распределение потока останется тем же, что и в графе *G.* Граф *G′* показан на рис. 3.13, где для удобства вершины y_j

обозначены через $x_{m+j} = x_{3+j}$. Поток φ'_z через этот граф равен 45 ед., т. е. меньше, чем первоначальный поток φ_z = 50 ед. Этот поток является полным, так как в графе G' все пути из x_0 в z содержат насыщенные дуги. Однако возможно, что он не является наибольшим.

Удалено: z

Если наибольший поток в графе G' равен φ_z, то найдется распределение этого потока, при котором наиболее продолжительный путь будет иметь время t_μ < 6 ед. Следовательно, дальнейшее решение задачи сводится к определению наибольшего потока в графе G'.

На рис. 3.13 произведена разметка вершин графа G' в соответствии с алгоритмом Форда и Фалкерсона. Разметка вершин указывает на существование пути $(x_0, x_2, x_4, x_3, x_6, z)$, на котором поток в направлении от x_0 к z может быть увеличен на 5 ед.

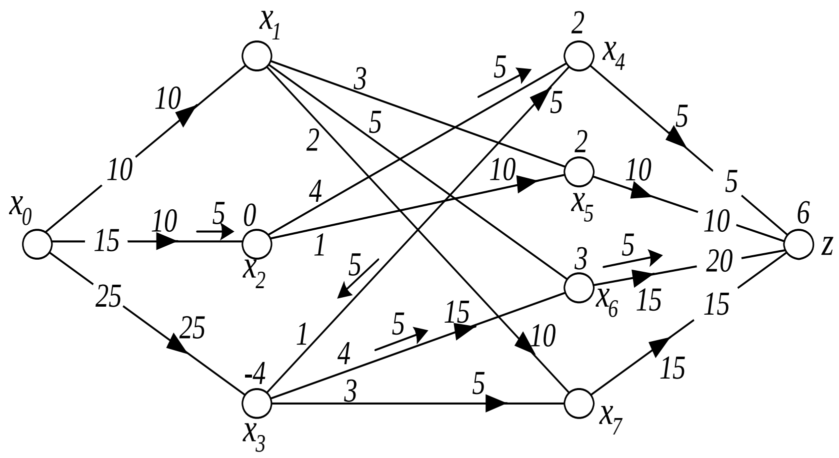


Рис. 3.13.

Этот добавочный поток показан стрелками. Как видим, наибольший поток в графе G' равен φ_z = 50 ед. и наиболее продолжительный маршрут имеет время 4 ед. Дальнейшего уменьшения времени прохождения потока добиться невозможно, так как к вершине $y_3(x_6)$ не идут маршруты со временем, меньшим 4 ед.

Окончательное распределение частичных потоков по маршрутам, дающее решение транспортной задачи по критерию времени, приведено в табл. 3.7.

Таблица 3.7

Маршрут, ед.	Частичный поток	Время прохождения потока
(x_2, y_2)	10	1
(x_1, y_4)	10	2
(x_3, y_4)	5	3
(x_2, y_1)	5	4
(x_3, y_3)	20	4
–	φ_z = 50	T_{max} = 4

3.5.8. Цикломатическое число графа

Пусть в графе *m* - число ребер, *n*- число вершин, *p* -число компонент связности. Величина $\rho = n - p$ называется коцикломатическим числом. *Цикломатическим* числом графа называют число $\nu = m-n+p$.

Формально понятие *независимых циклов* вводится следующим образом. Введём в графе произвольную ориентацию ребер. Рассмотрим некоторый произвольный цикл и сопоставим ему вектор, который содержит m компонент, сопоставленных рёбрам по правилу: значение компоненты $r_{ij} = r^{+}_{ij} - r^{-}_{ij}$, *где* r^{+}_{ij} - число проходов по циклу по направлению ребра, r^{-}_{ij} –число проходов против направления ребра.

Введем две операции на множестве всех векторов: сложение векторов как покомпонентное сложение их компонент и умножение вектора на число как умножение каждой компоненты на это число.

Множество векторов называется *линейно независимым*, если равенство $\alpha_1 r_1 + \alpha_2 r_2 ... + \alpha_к r_к = 0$ выполняется только при $\alpha_1 = \alpha_2 = ... = \alpha_к = 0$. Множество векторов с введенными операциями образуют *линейное пространство*. Множество, содержащее максимальное число его линейно независимых векторов, называется *базисом пространства*, число элементов базиса называется *размерностью* пространства. В этом пространстве любой вектор может быть выражен как сумма произведений базисных векторов на константы.

Теорема. В графе число ν определяет число независимых циклов в нем.

Доказательство проводится по индукции. Предположим, что теорема справедлива для числа рёбер m, и покажем, что тогда она справедлива и для $m` = m+1$. Добавим в граф *G* ещё одно ребро *(a,b),* тогда возможны два случая:

1. В *G* вершины *a* и *b* связаны цепью, тогда в расширенном графе *G'* добавится еще один цикл, т.е. будет $m' = m+1$, $p' = p$, $\nu' = \nu+1$.
2. В *G* вершины *a* и *b* цепью не связаны, тогда в расширенном графе *G'* уменьшится на единицу число компонент связности, т.е. $m' = m+1$, $p' = p - 1$, $\nu' = \nu$.

Для любого графа проделаем такую процедуру: удалим из него все ребра и затем введём их по одному. Для графа без ребер утверждение теоремы верно (число циклов и число ребер равно 0, число вершин равно числу компонент связности). Каждое новое ребро будет приводить к описанным выше двум случаям, т.е. будет выполняться утверждение теоремы.

Если графу сопоставить электрическую сеть, то ρ – наибольшее число независимых разностей потенциалов в сети между ее узлами, ν – число независимых круговых токов, которые могут протекать в этой сети.

3.5.9. Планарные графы

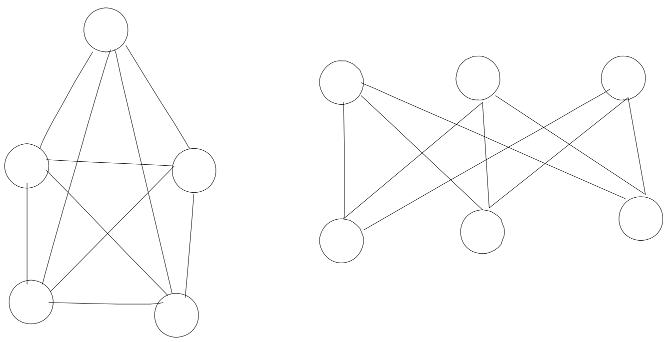
Плоскими графами называют графы, которые можно так расположить на плоскости, чтобы ребра пересекались только в вершинах графа. Графы, которые расположены на плоскости без пересечения, называют планарными. Планарные графы, получающиеся один из другого непрерывной деформацией плоскости, не считаются различными.

Часть плоскости, ограниченная ребрами и не содержащая внутри себя ни вершин, ни ребер, называется *гранью*. Одна из граней является *внешней*. Грани, разделенные ребрами, называются *смежными*.

Для планарного графа число граней f связано с числом его вершин n и числом его ребер m формулой Эйлера n-m+f=2.

В справедливости формулы можно убедиться следующим образом. Граням сопоставим циклы из ребер, их ограничивающих. Число граней больше на единицу числа независимых циклов (за счет внешней грани), т.е. с учетом формулы для ν f-1=m-n+1, откуда непосредственно следует формула Эйлера.

Можно убедиться, что минимальным неплоским графом по числу элементов будет граф K_5 (рис 3.14,а), по числу ребер–максимальный двудольный граф с числом вершин 6 (такой граф обозначается как $K_{3,3}$, рис.3.14,б). Если такой граф входит в G, то G становится неплоским.



а б

Рис.3.14

Граф G' называют *гомеоморфным* графу G, если он может быть получен из G заменой некоторых цепей на ребра.

Условие планарности графа определяет теорема Понтрягина-Куратовского:

Граф G будет плоским тогда и только тогда, когда он не содержит графов, гомеоморфных графам K_5 и $K_{3,3}$.

Теорема Понтрягина-Куратовского определяет условия того, что граф является плоским, но не даёт конструктивного способа организации проверки этого свойства (кроме полного перебора).

К практическим задачам, связанным с расположением графа на плоскости, можно отнести такие задачи, как

- расположение графа на плоскости с минимальным числом пересечений его рёбер;
- выделение минимального числа рёбер, удаление которых делает граф плоским;
- разбиение графа на минимальное число плоских подграфов.

3.6. Операции над графами.

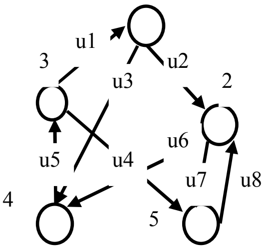


Рис.3.15

Так как граф описывает бинарное отношение, то все операции над бинарными отношениями можно трактовать как операции над графами, в том числе и теоретико-множественные операции: дополнение, пересечение, объединение. Дополнение проводится до полного графа и включает в результат все дуги, которых нет в исходном графе.

Рассматриваются специальные «графовые» операции.

Рёберный граф графа *G*. В нём каждому ребру *G* сопоставляется вершина, вершины связываются ребром, если в *G* сопоставленные им ребра инцидентны.

Граф достижимости графа *G*. В нём множество вершин то же, что в *G*, и вершины связаны дугой, если в графе *G* между ними существует путь. В табл. 3.8 и 3.9 приведены матрицы смежности графа дополнения и реберного графа для графа, приведённого на рис.3.15.

Группа операций связана с операциями над матрицами смежности, описывающими граф. Обозначим операцию, сопоставленную матричному умножению для графов *G* и *H,* как $G{\cdot}H$.

Рассмотрим частный случай, когда $G=H$, т.е. операцию $G{\cdot}G$. Обозначим её как G^2.

Таблица 3.8

	1	2	3	4	5
1	1		1		1
2	1	1	1		
3		1	1	1	
4	1	1		1	1
5	1		1	1	1

В этом случае получается матрица, (*I,j*)- я компонента которой вычисляется по формуле, где $n = |A|$.

$$(i,j) = \sum_{k=1}^{n}(i,k)\cdot(k,j)$$

Удалено: ¶
¶

Таблица 3.9

	1,2	1,4	2,5	2,4	3,1	3,5	4,3	5,3
1,2	1	1	1	1				
1,4	1	1		1			1	
2,5	1		1	1		1		1
2,4	1	1	1	1			1	
3,1	1	1			1	1	1	1
3,5			1		1	1	1	1
4,3				1	1	1	1	1
5,3			1		1	1	1	1

Результирующая матрица может содержать значения большие, чем 1, т.е. результатом будет мультиграф. В этом графе дуга (i,j) имеет кратность, равную числу путей длины 2 между вершинами a_i и a_j в графе G. На рис.3.16 приведён граф G, для которого G^2 представлен на рис.3.17. В нём дуга (2,5) имеет кратность 2, так как между вершинами 2 и 5 два пути длины 2– через вершины 1 и 3.

Удалено: a

Можно показать, что для графа G^3 результатом будет мультиграф, в котором между a_i и a_j будет столько дуг, сколько путей длины 3 между этими вершинами в исходном графе.

Удалено: a

Удалено: a

Значит, если объединить все степени графа G (считая, что G^1=G) от 1 до бесконечности, то результатом будет граф достижимости, описанный выше.

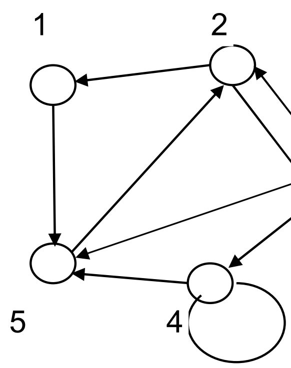


Рис.3.16

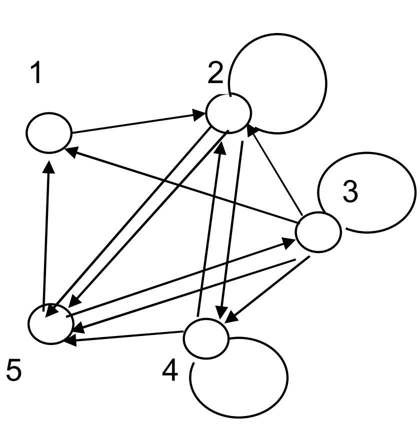


Рис.3.17

Введём ещё одну бинарную операцию над графами, результирующий граф для которой задан на декартовом произведении множеств вершин графов-аргументов.

Пусть G=<A,R> и H=<B,S>. P=GxH, где P=<C,T>, C=AxB, T={((a,b),(R(a),S(b)))}.

Такую операцию называют произведением графов. Рассмотрим эту операцию подробнее на примере. Пусть G и H имеют вид рис. 3.18 и 3.19.

Произведение этих графов приведено на рис.3.20, его матрица записана в табл.3.10.

Матрица произведения может быть получена по такому правилу. Запишем матрицу одного. из сомножителей, затем в ней каждую клетку заменим на матрицу, размер которой совпадает с размером матрицы второго графа, а содержание равно пустой матрице, если соответствующий элемент матрицы первого графа равен 0, или совпадает с содержимым матрицы второго графа, если элемент равен 1. Такие матрицы получили название *правильные клеточные матрицы.*

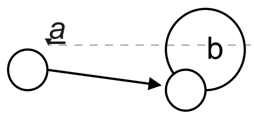


Рис.3.18

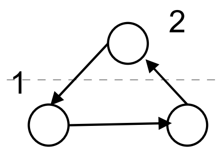


Рис3.19

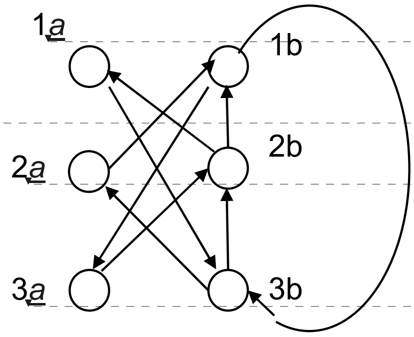


Рис3.20

Удалено: a

Удалено: a

Удалено: a

Удалено: a

Семантика операции. Пусть имеется два блока, составляющие систему, функционирующую в общем дискретном времени. Каждый блок описан графом, вершины которого сопоставлены с состояниями блока, дуга (*i,j*) определяет возможный переход блока из состояния *i* в следующий момент времени. Блоки функционируют независимо и параллельно.

Таблица 3.10

	1a	1b	2a	2b	3a	3b
1a						1
1b					1	1
2a		1				
2b	1	1				
3a				1		
3b			1	1		

В этом случае возможные состояния системы определяются состояниями блоков (декартовым произведением), а произведению соответствует граф перехода системы во времени. Так, если в примере первый блок находится в состоянии 1, второй в это время находится в состоянии b, то система находится в состоянии (1,b) и в следующий момент возможны переходы в состояние (3,a) или (3,b).

Удалено: b

Удалено: b

Удалено: a

Удалено: b

Задание. Система состоит из двух блоков, функционирующих в общем времени, и в каждый момент времени меняет состояние только один блок (последовательная работа блоков). Определите операцию над графами, описывающими переходы блоков системы, результатом которой является граф переходов системы.

3.7. Декомпозиция графов

Удалено: .

Задача декомпозиции формулируется так: задан граф, для которого необходимо определить, является ли он произведением двух графов. В случае положительного ответа нужно найти графы - сомножители.

Семантика задачи. Описана система, функционирующая во времени. Необходимо определить, нельзя ли её представить в виде системы из двух независимых блоков, как описано выше. В этом случае систему можно описать гораздо проще. Так, если система имеет 100 состояний, то при положительном решении она будет описываться как два блока, каждый из которых имеет только десять состояний.

В предыдущем разделе было показано, что если граф представим произведением, то его матрица будет иметь вид правильной клеточной матрицы.

Рассмотрим ещё раз пример, приведённый в предыдущем разделе в табл. 3.10. Предположим, что в системе состояния пронумерованы следующим образом: 1=(1*a*), 2=(1*b*), 3=(3*b*), 4=(2*b*), 5=(3*a*), 6=(2*a*), т.е. 3-ю и 6-ю вершины поменяем местами. Получим табл. 3.11.

Удалено: a
Удалено: b
Удалено: b
Удалено: b
Удалено: a
Удалено: a
Удалено: то есть
Удалено: Тогда таблица примет вид

Таблица 3.11

	1	2	3	4	5	6
1			1			
2			1		1	
3				1		1
4	1	1				
5				1		
6		1				

Эта таблица уже не будет правильной клеточной матрицей, хотя она описывает граф, совпадающий с исходным с точностью до имён вершин.

Значит, если матрица графа не является правильной клеточной матрицей, но может быть получена из неё путём переименования вершин, то такой граф будет представлять собой произведение графов.

... [1]

Есть ещё один простой критерий того, что граф не является произведением. Если Q=<C,T>, G=<A,R>, H=<B,S>, и Q=GxH, то имеет место условие: $|C|=|A|\cdot|B|$. Значит, если число вершин n графа есть простое число, то он не может быть представлен произведением графов.

Удалено: ¶

Если n не простое и может быть выражено как $n=kr$, где k и r больше 1, то можно ставить задачу проверки этого графа на возможность представить его в виде произведения. Если граф содержит n вершин, то для проверки его необходимо в общем случае провести n! переименований. Уже для n=6 это число составит 720, для n=8 уже более

40000. Поэтому возникает необходимость сокращения объёма вычислений.

Возможность сокращения основана на следующем утверждении. Если граф представлен в виде правильной клеточной матрицы, то при переименовании вершин, связанных с переименованием вершин в одном графе-сомножителе, матрица останется правильной клеточной матрицей. Точно так же, если матрица не является правильной клеточной, то перестановка имен, связанная с изменением имён в одном графе-сомножителе, её правильной не сделает. Это значит, что если $n=k\cdot r$, то необходимый перебор вариантов можно сократить в $k!\cdot r!$ раз. Например, перебор для n=6 сократится до 60 (6=2·3, сокращение в 2!·3!=12 раз).

Алгоритм декомпозиции

Рассмотрим еще раз операцию произведения. Если вершина *a* в первом графе-сомножителе имеет степень захода d_a, а вершина *b* во втором графе имеет степень захода d_b, в результирующем графе вершина (*a,b*) будет иметь степень захода ($d_a d_b$). То же самое можно сказать и о степени исхода результирующего графа.

Удалено: a
Удалено: b
Удалено: a
Удалено: b

Алгоритм основан на том, что для каждой *i* –той вершины графа определяются все возможные разложения её степеней захода $s_i=r_i\cdot t_i$, т.е. предполагается, что если задача решается и этой вершине сопоставлена пара (*x,y*), то эти вершины имеют степени захода r_i и t_i соответственно. То же самое относится к степени исхода вершины *i*: $t_i=p_i\cdot q_i$.

Проведём разложения для всех степеней вершин. Построим два разбиения вершин: разбиение на *к* классов по *l* элементов $\pi=(\pi1,\pi2,\ldots,\pi к)$ и разбиение на *l* классов по *к* элементов $\rho=(\rho1,\ \rho2,\ \ldots,\ \rho l)$, $n=k\cdot l$, чтобы произведением этих разбиений было разбиение на одноэлементные множества.

В один класс π попадут вершины *a* и *b*, если для них в разложении $r_a=r_b$ и $p_a=p_b$.

В один класс ρ попадут вершины *a* и *b*, если в разложениях $t_a=t_b$ и $q_a=q_b$.

Удалено: a
Удалено: b
Удалено: a
Удалено: b
Удалено: a

Если такие разбиения построить можно, то ему сопоставляется перестановка элементов по следующему правилу. Зафиксируем порядок блоков в разбиениях π и ρ. Сопоставим этому порядку перестановку: вначале старшинство определяется по разбиению π, затем внутри этих блоков - по разбиению ρ.

Рассмотрим метод на примере. Пусть граф описывается матрицей табл. 3.12, разложения *s* и *t* приведены в этой таблице. Значение степени, равное 0, представляется произведением 0 на произвольное число, обозначаемое как *x*. Первой задачей является задача выбора среди

Удалено: /
Удалено: разложения

разложений тех, которые приведут к разбиениям вершин по степеням исхода и захода.
При выборе разложений будем учитывать следующее.

Для вершины 4 выбираем разбиение 2·2, так как вершиныу с сомножителем 4 в пару не найти. В разложения π вершины 4 и 5 должны быть в разных блоках, так как у них обе компоненты по *s* разные. То же самое по отношению *t* для вершин 3 и 4. Значит, 3 и 5 будут в одном блоке. Вершины 3 и 6 должны быть в одном блоке по *t* и в разных блоках по *s,* 1 и 5 – в разных блоках по *t*, чтобы по первой компоненте их можно было объединить в разложении ρ. Курсивом выделены разбиения. В итоге

Таблица 3.12

	1	2	3	4	5	6	S
1		1		1			2=*1·2*=2·1
2				1		1	2=1·2=*2·1*
3							0=x·0=0·x
4		1	1	1		1	4=*2·2*=1·4=4·1
5				1			1=*1·1*
6							0=x·0=0·x
t	0=0·x =x·0	2=1·2 =*2·1*	1=*1·1*	4=*2·2* 1·4=4·1	0=x·0 =0·x	2=*1·2* =2·1	

получаем разложение π={{1,4,6}, {2,3,5}}.

После этого однозначно получаем ρ={{1,5},{3,6},{2,4}}.

Зафиксируем порядок в этих разбиениях. Тем самым мы предполагаем, что если решение существует, то граф можно представить произведением двух графов, один из которых содержит 2 вершины (пусть, например, вершины *a* и *b*), второй содержит три вершины (пусть, например, 1, 2 и 3). Тогда вершинам первого блока разбиения π сопоставлены элементы декартова произведения с вершиной *a* в качестве первого слагаемого, для вершин второго блока первым слагаемым будет *b*.

Аналогично в соответствующем декартовом произведении вторым сомножителем для вершин первого блока разбиения ρ будет вершина 1, второго блока – вершина 2, третьего блока – вершина 3. Составим разбиениям π и ρ упорядочение <1,4,6,5,2,3>, которое «испортило» матрицу, исправим её, используя обратное упорядочение

1 2 3 4 5 6
1 5 6 2 4 3.

Применим эту перестановку к матрице, получим табл.3.13. Как видно из таблицы, она является правильной клеточной матрицей, что приводит к решению: граф представим произведением двух графов, матрицы смежности которых легко получаются из табл. 3.13 Матрицы графов-сомножителей представлены в табл. 3.14 и 3.15.

Таблица 3.13

	1	2	3	4	5	6
1				1		
2			1	1		
3				1		1
4			1	1	1	1
5						
6						

Таблица 3.14

	a	b
a		1
b	1	1

Таблица 3.15

	1	2	3
1		1	
2		1	1
3			

Удалено: --Разрыв страницы-

3.8. Контрольные вопросы и задания

3.8.1. Сетевые графики

Удалено: ¶ Задачи.¶

1. В сетевом графике определить ранние и поздние сроки начала и окончания работы (**a**,**b**). Для неё определить свободный и полный резервы времени

.

Удалено: ¶

Вариант 1.

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
a		*3*		*3*				
b						*5*	*7*	
c						*5*	*6*	
d		*4*		*8*				
e			*3*			*7*		
f							*9*	
g								
h	*4*			*8*	*9*			

Вариант 2.

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
a		*3*			*3*			
b			*5*				*7*	
c							*9*	
d			*7*			*3*		
e		*4*		*7*				
f			*5*				*6*	
g								
h	*4*			*9*	*8*			

Вариант 3.

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
a		*3*		*3*				
b						*5*	*7*	
c						*5*	*6*	
d		*4*			*6*			
e			*3*			*7*		
f							*9*	
g								
h	*4*			*8*	*9*			

Удалено: ¶

Вариант 4.

Удалено: ¶

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
a		*3*					*3*	
b					*5*			*7*
c					*5*			*6*
d								*8*
e				*3*	*7*			
f		*4*				*7*		
g								
h	*4*					*9*	*8*	

Вариант 5.

	a	b	c	d	e	f	g	h
a		3		3				
b						5	7	
c						5	6	
d		4			7			
e			3			7		
f							9	
g								
h	4			8	9			

Вариант 6

	a	b	c	d	e	f	g	h
a					4			
b							8	
c	2				3	5		
d		6	5			7		
e							5	
f		3			2		6	
g								
h	5		7	4				

Вариант 7.

Удалено: --Разрыв страницы--

	a	b	c	d	e	f	g	h
a		5				6		
b							1	
c					8	5		
d	4		2			9		
e						2	6	
f		2					2	
g								
h	6		7	4				

Вариант 8.

	a	b	c	d	e	f	g	h
a		6				7		
b							2	
c					9	6		
d	5		3			9		
e						2	7	
f		3					4	
g								
h	7		8	5				

3.8.2. Выделение минимального остова

Задачи.

В заданном графе выделить остов

Вариант 1

	1	2	3	4	5	6	7	8	9
1		4	5	5			3		
2				5	4				
3					3	4		6	
4						3	4		
5							5		7
6									6
7								7	
8									5
9									

Вариант 2

	1	2	3	4	5	6	7	8	9
1		3	6	5	7				
2				5	4				
3					3	5			
4						6	4		
5							8		7
6								5	6
7								7	
8									3
9									

Вариант 3

	1	2	3	4	5	6	7	8	9
1		5	7	5			3		
2				6	4				
3					3	4		6	
4						2	4		
5							8		7
6									6
7								7	
8									5
9									

Вариант 4

	1	2	3	4	5	6	7	8	9
1		3	2	5	7				
2				5	6				
3					3	5			
4						6	4		
5							8		7
6								5	6
7								7	
8									3
9									

Вариант 5

	1	2	3	4	5	6	7	8	9
1		4	7	5			3		
2				5	4				
3					3	4		6	
4						3	4		
5							4		7
6									6
7								7	
8									5
9									

Вариант 6

	1	2	3	4	5	6	7	8	9
1		4	5	2			7		
2				5	5				
3					5	4		2	
4						3	4		
5							2		7
6									6
7								7	
8									5
9									

3.8.3. Задачи назначения

Удалено: ¶

Определить совершенное максимальное паросочетание в двудольном графе.

Удалено: .¶

Вариант 1

	1	2	3	4	5	6
1	3	5	6	5	2	1
2	2	4	7	6	4	3
3	4	5	4	3	3	4
4	1	2	1	2	3	3
5	2	4	3	4	5	3
6	3	2	3	4	2	2

Вариант 2

Удалено: ¶

	1	2	3	4	5	6
1	5	6	2	4	3	1
2	2	3	2	4	3	3
3	4	5	4	4	3	4
4	1	2	5	4	3	3
5	2	2	3	3	5	3
6	2	2	3	1	4	2

Вариант 3

Удалено: ¶

	1	2	3	4	5	6
1	2	5	6	5	2	1
2	1	4	7	6	4	3
3	3	3	6	4	3	4
4	1	2	3	2	3	3
5	3	2	3	4	2	3
	2	4	1	4	5	2

Вариант 4

Удалено: ¶

	1	2	3	4	5	6
1	2	3	1	5	2	1
2	2	4	4	6	4	3
3	1	5	4	2	3	4
4	1	2	1	6	3	3
5	2	4	3	1	5	3
6	1	4	3	4	1	2

Вариант 5

	1	2	3	4	5	6
1	3	5	2	3	2	1
2	2	4	7	6	4	3
3	4	5	4	4	2	4
4	1	2	1	2	3	3
5	2	4	3	4	5	3
6	3	2	3	4	2	2

Вариант 6

	1	2	3	4	5	6
1	3	5	6	5	2	1
2	8	4	7	6	4	3
3	6	5	4	4	3	4
4	1	2	1	2	3	3
5	2	4	3	4	5	3
6	3	6	3	4	2	2

Вариант 7

Удалено: ¶

	1	2	3	4	5	6
1	3	5	3	3	2	1
2	2	4	3	6	4	3
3	4	5	2	4	3	4
4	3	2	1	2	2	1
5	3	4	3	4	5	3
6	4	2	2	4	2	2

Вариант 8

	1	2	3	4	5	6
1	3	5	3	5	2	1
2	2	4	2	6	4	3
3	4	5	2	4	3	4
4	1	2	3	2	1	3
5	1	2	1	4	5	2
6	3	2	3	4	2	2

3.8.4. Потоки в сетях

Определить величину максимального потока, который можно пропустить через заданную сеть.

Вариант 1.

	a	b	c	d	e	f	g	h
a		3		3				
b						5	7	
c						5	6	
d		4			8			
e			3			7		
f							9	
g								
h	4			8	9			

Вариант 2.

	a	b	c	d	e	f	g	h
a		3			3			
b			5				7	
c							9	
d			7			3		
e		4		7				
f			5				6	
g								
h	4			9	8			

Вариант 3.

	a	b	c	d	e	f	g	h
a		3		3				
b						5	7	
c						5	6	
d		4			6			
e			3			7		
f							9	
g								
h	4			8	9			

Вариант 4.

	a	b	c	d	e	f	g	h
a		3				3		
b				5			7	
c				5			6	
d							8	
e			3	7				
f		4			7			
g								
h	4				9	8		

Вариант 5.

	a	b	c	d	e	f	g	h
a		3		3				
b						5	7	
c						5	6	
d		4			7			
e			3			7		
f							9	
g								
h	4			8	9			

Вариант 6.

	a	b	c	d	e	f	g	h
a					4			
b							8	
c	2				3	5		
d		6	5			7		
e							5	
f		3			2		6	
g								
h	5		7	4				

Вариант 7.

	a	b	c	d	e	f	g	h
a		5				6		
b							1	
c					8	5		
d	4		2			9		
e						2	6	
f		2					2	
g								
h	6		7	4				

Вариант 8.

	a	b	c	d	e	f	g	h
a							5	
b							8	
c	3			2		5		
d	4							
e		6	5			7		
f	2	3					6	
g								
h			7	5	4			

Удалено: ¶
Удалено: ¶
Удалено: ¶ ¶
Удалено: ¶ ¶ Разрыв страницы
Удалено: ¶ ¶
Удалено: ¶ ¶ ¶ ¶ ¶
Удалено: ¶ ¶

3.8.5. Декомпозиция графа

Для заданного графа определить, разложим ли он в произведение двух графов. Если разложим, то найти графы – сомножители.

.

Вариант 1.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	1	1	1	0	1	0
2	0	0	0	0	1	0	0	1	0
3	0	0	0	0	1	0	0	1	0
4	0	0	0	1	1	1	0	1	0
5	0	1	0	0	0	0	0	0	1
6	0	0	0	1	1	0	0	0	0
7	0	1	0	0	0	0	0	0	1
8	0	1	0	0	0	0	0	0	0
9	0	0	0	0	1	0	0	0	0

Вариант 2.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	1	0	1	0	1	1
2	0	0	0	0	0	0	0	1	1
3	0	0	0	0	0	0	0	1	1
4	0	0	0	1	0	1	0	1	1
5	0	0	0	0	0	0	0	0	1
6	0	0	0	1	0	0	0	0	1
7	0	1	0	0	1	0	0	0	0
8	0	1	0	0	0	0	0	0	0
9	0	1	0	0	1	0	0	0	0

Вариант 3.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	0	1	0	0	1	0
2	0	0	0	0	1	0	0	1	0
3	0	0	0	1	1	1	0	1	0
4	0	0	0	1	1	1	0	1	0
5	0	1	0	0	0	0	0	0	1
6	0	0	0	1	1	0	0	0	0
7	0	1	0	0	0	0	0	0	1
8	0	1	0	0	0	0	0	0	0
9	0	0	0	0	1	0	0	0	0

Вариант 4.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	1	1	0	1	1	0
2	0	0	0	0	1	0	0	1	0
3	0	0	0	0	1	0	0	1	0
4	0	0	0	1	1	0	1	1	0
5	0	1	0	0	0	0	0	0	1
6	0	1	0	0	0	0	0	0	1
7	0	0	0	1	1	0	0	0	0
8	0	1	0	0	0	0	0	0	0
9	0	0	0	0	1	0	0	0	0

Вариант 5.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	1	1	1	0	0	1
2	0	0	0	0	1	0	0	0	1
3	0	0	0	0	1	0	0	0	1
4	0	0	0	1	1	1	0	0	1
5	0	1	0	0	0	0	0	1	0
6	0	0	0	1	1	0	0	0	0
7	0	1	0	0	0	0	0	1	0
8	0	0	0	0	1	0	0	0	0
9	0	1	0	0	0	0	0	0	0

Вариант 6.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	1	1	1	1	0	0
2	0	0	0	0	1	0	1	0	0
3	0	0	0	0	1	0	1	0	0
4	0	0	0	1	1	1	1	0	0
5	0	1	0	0	0	0	0	0	1
6	0	0	0	1	1	0	0	0	0
7	0	1	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	1
9	0	0	0	0	1	0	0	0	0

Вариант 7.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	1	1	1	1	0	0
2	0	0	0	0	1	0	1	0	0
3	0	0	0	0	1	0	1	0	0
4	0	0	0	1	1	1	1	0	0
5	0	1	0	0	0	0	0	0	1
6	0	0	0	1	1	0	0	0	0
7	0	1	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	1
9	1	0	1	0	1	0	0	1	0

Вариант 9.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	0	1	1	0	0	0
2	0	0	0	0	1	0	1	0	0
3	0	0	0	0	1	0	1	0	0
4	0	0	0	1	1	1	1	1	0
5	0	1	0	0	0	0	0	0	1
6	0	0	0	1	1	0	0	0	0
7	0	1	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	1
9	0	0	0	0	1	0	0	0	0

Удалено: ¶

... [3]

Вариант 8.

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	1	1	1	1	0	0
2	0	0	0	0	1	0	1	0	0
3	0	0	0	0	1	0	1	0	0
4	0	0	0	1	1	1	1	1	0
5	0	1	0	0	0	0	0	0	1
6	0	0	0	1	1	0	0	0	0
7	0	1	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	1
9	0	0	0	0	1	0	0	0	0

4. ГРУППЫ И ПОЛУГРУППЫ

Удалено: Введение

4.1. Алгебры

Удалено: ¶

Пусть задано некоторое множество *A*. Для любого натурального числа *к* множество $A^k = AxAx...xA$ – декартова степень множества *A* – состоит из всевозможных упорядоченных наборов к элементов из *A*. Правило, позволяющее однозначно сопоставить любому элементу из $A^к$ элемент из *A*, называется *операцией на A*. Величина **к** называется *арностью операции*. Запись вида

$$O(a,b, ..., d)=g \qquad (*)$$

означает, что *g* есть результат операции O, примененной к элементам *a,b, ..., d* (аргументы операции).

Операция называется *унарной,* если k=1, *бинарной*, если k=2. Для бинарной операции, кроме записи вида *O(a,b)=c*, используется запись *aOb=c*.

Определение. *Алгеброй называется пара* $G=\langle A, \varphi\rangle$, *где A – множество,* $\varphi = \{\varphi_i \mid i = 1,k\}$ – *множество операций на A.*

Множество *A* называется *основным* множеством или *основой* алгебры, множество φ - *сигнатурой* алгебры.

Далее будем рассматривать алгебру, сигнатура которой содержит только бинарные операции.

Операция называется *ассоциативной*, если $(aOb)Oc=aO(bOc)$ для любых *a, b, c* из *A*. В этом случае результат не зависит от порядка выполнения операций. При этом скобки можно опустить и писать *aObOc*.

Операция называется *коммутативной*, если $aOb=bOa$ для любых *a, b* из *A*.

Определение. Множество с одной ассоциативной операцией на нём называется *полугруппой.*

Удалено: ¶

Чаще других используется мультипликативная запись операции, т.е. операцию называют умножением и обозначают значком ·(точка). Если не возникает разночтения, символ точки можно опускать, Например, вместо $a{\cdot}b$ можно писать *ab*. Используют также аддитивную запись, когда операцию называют сложением и обозначают знаком **+**. Будем пользоваться, если особо не оговорено противное, мультипликативной записью.

Если операция является коммутативной, то полугруппа называется *абелевой.*

Если найдется элемент $e\in A$ такой, что $\forall a\in A \; a{\cdot}e=e{\cdot}a=a$, то *e* называется *единицей* полугруппы и обозначается символом 1. При аддитивной записи элемент *e* называют *нулём* и обозначают символом 0.

Определение. Полугруппа с единицей называется *моноидом*.

Упражнение. Докажите, что в моноиде единица единственна.

4.2. Группы

Удалено: Её принято обозначать через 1.¶

Пусть *a* и *b* – элементы полугруппы. Элемент *a* называется *обратным* к элементу *b*, если $a\cdot b = b\cdot a = 1$. Обозначим обратный элемент к *a* через a^{-1}.

Удалено: *в*

Определение. Если для каждого элемента полугруппы с единицей найдется обратный к нему элемент, то полугруппа называется *группой.*

Пример 1. Множество всех целых чисел с операцией сложения является группой, в которой единицей будет число 0. Для каждого элемента *k* обратным к нему будет элемент -*k*.

Удалено: Пример

Пример 2. Множество целых чисел от 0 до *n-1*, где *n* – натуральное число, будет группой с операцией сложения по модулю *n*. Результатом операции сложения по модулю *n* является остаток от деления суммы чисел на *n*. Так, сумма по модулю 8 чисел 7 и 6 равна 5. Единицей группы будет число 0. Для числа *k* обратным будет число *n-k*.

Удалено: Пример

Подмножество *B* множества *A* называют *замкнутым* относительно операции ·(точка), если $\forall a,b\in B\ a\cdot b\in B$. Замкнутое относительно групповой операции подмножество группы, содержащее вместе с каждым своим элементов и обратный ему элемент, называется *подгруппой*.

В предыдущем примере 1 подгруппой будет множество чётных чисел. Действительно, сумма двух чётных чисел будет число чётное, значит, множество их будет замкнуто относительно сложения, и обратным к чётному числу будет число чётное.

Легко проверить, что пересечение двух подгрупп будет подгруппой. Все подгруппы имеют непустое пересечение, так как в каждую из них входит единичный элемент. Сама группа, так же как и одноэлементное подмножество, состоящее из единичного элемента, будет подгруппой этой группы. Подгруппа, не совпадающая с самой группой, называется собственной подгруппой.

Рассмотрим некоторый элемент *g* группы *G*. Элемент $g^k=g\cdot g\cdot \quad \cdot g$ (*k* сомножителей) при $k \geq 1$ называется *k*–й степенью элемента *g*. Для $k< 0$ принимают $g^k = (g^{-1})^{(-k)}$ и $g^0=1$.

Если для элемента g найдется число $k \geq 1$ такое, что $g^k=1$, то минимальное среди таких *k* называют *порядком* элемента *g.* Если для элемента такого *k* нет, то элемент называют элементом бесконечного порядка

Нетрудно показать, что если $g^k=1$, то *k* будет кратным порядку *g*.

Множество всех степеней элемента образует подгруппу в *G*. Эта группа (порожденная одним элементом) называется *циклической*. Циклическая группа может содержать в себе собственные подгруппы. Любая подгруппа циклической группы – циклическая. Циклическая группа является абелевой.

Пусть в группе, определенной на множестве $G = \{g_1,g_2, . . .,g_k\}$, *H* – ее собственная подгруппа. Возьмем элемент *a* группы и построим множество элементов, равных $a \cdot g_i$ для всех $g_i \in H$. Обозначим это множество через *aH* и назовем *правым смежным классом* по *H.*

Упражнение. Покажите, что в этом классе число различных элементов совпадает с числом элементов в подгруппе *H*. (Подсказка: нужно показать, что $g_r \cdot h_t \neq g_r \cdot h_q$ для любых двух различных элементов h_t, $h_q \in H$).

Выберем элемент *b* (если таковой есть), не принадлежащий ни подгруппе *H*, ни построенному классу *aH*, и построим новый правый смежный класс, включающий этот элемент. Повторяя эту процедуру, построим множество всех правых смежных классов по *H*. Это множество обозначим как $(G:H)_{пр}$.

Аналогично стоится множество всех левых смежных классов $(G:H)_{лев}$ умножением всех элементов *H* на элементы группы *G* слева. Для правых и левых смежных классов доказывается теорема.

Теорема: Мощность смежного класса по H равна мощности H. Смежные классы попарно не пересекаются.

Число правых смежных классов равно числу левых смежных классов

Упражнение. Покажите, что имеет место соотношение

$$aH = bH \Leftrightarrow a^{-1} \cdot b \in H.$$

Число правых смежных классов *G* по *H* обозначают через $|G:H|$ и называют *индексом подгруппы H в группе G*.

Теорема Лагранжа: Если H – подгруппа конечной группы **G***, то* $|G|=|H| \cdot |G:H|$.

Доказательство непосредственно вытекает из предыдущей теоремы.

Важнейшие следствия этой теоремы:

1) порядок подгруппы всегда делит порядок группы;

2) всякая группа простого порядка – циклическая.

Особую роль в теории групп играют подгруппы, относительно которых правые смежные классы совпадают с левыми. Такие подгруппы называются *нормальными* в **G.**

Пусть *G* – группа и *H* – ее нормальная подгруппа. Тогда $aH \cdot bH=(a \cdot b)H$, т.е. множество левых смежных классов замкнуто относительно поэлементного умножения классов. Легко проверить, что это множество является группой. Эта группа называется *фактор-группой* группы **G** по нормальной подгруппе *H* и обозначается через *G/H.*

4.3. Изоморфизмы и гомоморфизмы

Под *изоморфизмом* понимается взаимно однозначное отображение одной группы в другую, сохраняющее операцию, т.е. отображение, при котором различные элементы имеют различные образы и образ произведения элементов равен произведению образов этих элементов.

Формальное определение таково: пусть φ – отображение группы G в группу G^*, что обозначается как $\varphi: G \to G^*$. Тогда φ называется изоморфизмом G в G^*, если $\forall a,b \in G$:

- $\varphi(a) \neq \varphi(b)$ при $a \neq b$;
- $\varphi(a \cdot b) = \varphi(a) \cdot \varphi(b)$.

Удалено: b

Формат: Список

Удалено:

Удалено: b

Удалено: b

Если существует изоморфизм G на G^*, то существует изоморфизм G^* на G – обратное отображение φ^{-1}, определяемое так: $\varphi^{-1}(a)=b$, если $\varphi(b)=a$.

Определение. *Группы G и G^* изоморфны (запись: $G \cong G^*$), если существует изоморфизм одной из них на другую.*

Если отказаться от взаимной однозначности отображения, то приходим к понятию *гомоморфизма*.

Пусть φ – гомоморфизм группы G в G^*.

Множество всех элементов из G, отображающихся при φ в единичный элемент из G^*, называется *ядром* гомоморфизма и обозначается через $Ker\ \varphi$.

Теорема: Ядро любого гомоморфизма группы является в ней нормальной подгруппой.

Упражнение. Докажите утверждение теоремы.

Следствие. Ядро гомоморфизма φ тогда и только тогда тривиально ($Ker\ \varphi = 1$), когда φ – изоморфизм.

Гомоморфизм группы в себя называется её *эндоморфизмом*.

Свойства группы во многом определяются её подгрупповой структурой, т.е. тем, какие подгруппы она имеет и как вложены они друг в друга. Важным шагом в изучении строения конечных групп является знаменитая теорема Силова.

Теорема. Пусть G – конечная группа, p – простое число. Для каждой степени p^α, делящей порядок G, в G существует подгруппа порядка p^α. Если $p^{\alpha+1}$ делит порядок G, то каждая подгруппа порядка p^α из G вложена в некоторую подгруппу порядка $p^{\alpha+1}$ из G. В частности, максимальные p-подгруппы из G – это в точности подгруппы порядка p^r, где p^r – максимальная степень p, делящая порядок G. Все максимальные p-подгруппы сопряжены в G. Количество максимальных p-подгрупп из G сравнимо с 1 по модулю p и делит порядок G.

Удалено: *Силова*

Подгруппа, имеющая порядок p^r, определяемый теоремой, называется *силовской* подгруппой G относительно простого числа p, или просто силовской p-подгруппой группы G.

Упражнение. Если порядок абелевой группы равен произведению двух простых чисел p и q, то группа будет циклической. Докажите это.

4.4. Симметрические группы

Пусть B — множество мощности n, $\varphi = \{\varphi_i | \ i = 1,2, \ldots ,m\}$ – множество всех отображений B в себя. Нетрудно показать, что $m = n^n$. На этом множестве определяется операция композиции отображений, понимаемая как последовательное их исполнение. Пусть, например, n=5 и B={1,2,3,4,5}. Тогда, если $\varphi_i=\begin{pmatrix}12345\\24353\end{pmatrix}$, $\varphi_r=\begin{pmatrix}12345\\43512\end{pmatrix}$, то их композиция

Удалено:

$\varphi_i\varphi_r=\begin{pmatrix}12345\\31525\end{pmatrix}$. Множество всех отображений B в себя с операцией композиции является полугруппой с единицей. Единицей будет тождественное отображение $e=\begin{pmatrix}12345\\12345\end{pmatrix}$. Эта полугруппа называется *симметрической полугруппой* степени n и обозначается через Σ_n.

Отображению φ из симметрической полугруппы сопоставляют величину разности между мощностью множества B и мощностью его образа при этом отображении. Эту величину называют дефектом отображения φ и обозначают через $\Delta\varphi$.

Взаимно однозначное отображение B в себя (т.е. отображение на себя или, что то же самое, отображение φ с $\Delta\varphi = 0$) называют *подстановкой*. Можно показать, что множество всех подстановок с операцией композиции будет группой. Действительно, для любого элемента обратным к нему будет «перевернутый» элемент. Например, для элемента $\begin{pmatrix}12345\\23154\end{pmatrix}$ обратным будет элемент $\begin{pmatrix}12345\\31254\end{pmatrix}$. Произведение этих элементов равно единичному элементу. Эту группу называют *симметрической* группой степени n и обозначают через S_n.

Циклической подстановкой называют такую подстановку, в которой для некоторого подмножества $B^1 \subseteq =\{b_1, b_2, \ldots b_k\}$, $k \le n$, образом $b_i^1 \in B^1$ является b_{i+1}^1 для любого i от 1 до $k-1$, для b_k^1 образом является b_1^1, а для всех других элементов образ элемента совпадает с самим элементом. Запись циклической подстановки $(b_1^1, b_2^1, \ldots, b_k^1)$. Из определения следует, что ту же циклическую подстановку можно представить как $(b_i^1, b_{i+1}^1, \ldots, b_k^1, b_1^1, b_2^1, \ldots, b_{i-1}^1)$, т.е. сдвинуть в записи элементы по циклу. Например, циклическую подстановку (1,2,3,4,5) можно записать в виде (2,3,4,5,1), или (3,4,5,1,2), или (4,5,1,2,3), или (5,1,2,3,4).

При k=2 циклическая подстановка называется *транспозицией*.

Легко показать, что если циклы не имеют общих элементов, то они перестановочны. Например, подстановку (1,2,3)(4,5) можно записать в виде (4,5)(1,2,3) или как (5,4)(2,3,1).

Любую циклическую (а следовательно, и любую) подстановку можно представить произведением транспозиций. Можно убедиться, что

$(b_1,b_2, \ldots ,b_к)=(b_1,b_2)(b_1,b_3)\ldots(b_1,b_к)$. Число транспозиций в произведении равно k-1.

Подстановка называется *четной*, если она представима произведением четного числа транспозиций, иначе подстановка *нечетная*. Нетрудно показать, что произведение четных подстановок является четной подстановкой, четного числа нечетных подстановок – также четной. Произведение нечетного числа нечетных подстановок является подстановкой нечетной. Отсюда следует, что множество четных подстановок образует подгруппу в S_n, называемую *знакопеременной* группой степени n. Эта группа обозначается как A_n.

4.5. Полные множества и задача В. М. Глушкова

Пусть задано множество $\Im$ отображений множества $A = \{1,2, \ldots , n\}$ в себя, $|A| = n$, т.е. $\Im$ - подмножество в Σ_n. Множество отображений называют замкнутым по операции композиции, если оно содержит вместе с каждой парой отображений их композицию. Замыканием множества $\Im$ по операции композиции называют минимальное по числу элементов замкнутое множество отображений, содержащее $\Im$. Если замыкание $\Im$ совпадает с Σ_n, то $\Im$ назовем полным в Σ_n и будем говорить, что $\mathcal{J}$ порождает Σ_n. Каковы свойства полного множества? Этот вопрос в определенном смысле эквивалентен задаче Поста о свойствах функционально полного множества для переключательных функций. Рассмотрим эти свойства.

Так как множество отображений с дефектом 0 замкнуто и образует группу S_n, то полное множество $\mathcal{J}$ должно содержать хотя бы одно отображение с дефектом, большим 0. Из того, что $\Delta\gamma_i\gamma_j \geq \max(\Delta\gamma_i, \Delta\gamma_j)$, следует, что в $\mathcal{J}$ есть *хотя бы одно отображение с единичным дефектом.*

Удалено: а

Подмножество всех отображений из $\Im$ *с нулевым дефектом должно порождать симметрическую группу* S_n. Как показал Н. Н. Воробьев, *эти два условия необходимы и достаточны для порождения* Σ_n.

Необходимым условием порождения S_n *является наличие в множестве* $\mathcal{J}$ *хотя бы одного нечетного отображения*, так как четные отображения могут порождать только отображения из A_n.

Примером множества, порождающего S_n, может служить множество, содержащее две подстановки: цикл длины n и транспозицию элементов, соседних в этом цикле. Добавив в это множество любой элемент с единичным дефектом, получим полное множество.

Если $\mathcal{J}$ порождает Σ_n, то любое отображение можно представить произведением элементов из $\Im$. Интересна следующая задача: для любого отображения из Σ_n найти равное ему произведение отображений из $\mathcal{J}$ с минимальным числом сомножителей и разработать алгоритм решения этой задачи.

Эти задачи были поставлены известным кибернетиком Виктором Михайловичем Глушковым. К ним он свел ряд задач о минимизации структуры ЭВМ.

Удалено: Задачи

4.6. Контрольные вопросы и задания

1. Докажите, что если обратный элемент к некоторому элементу в полугруппе существует, то он единственный.

2. Покажите, что правые смежные классы по H попарно не пересекаются, т.е. множество $(G:H)_{пр}$ является разбиением множества G.

3. Определите, что является единицей фактор-группы G/H? Что будет обратным элементом для элемента aH ?

4. В группе порядка 15 определить число элементов, имеющих порядок 15.

5. В циклической группе порядка 20 определить число различных порождающих множеств, состоящих из двух элементов.

6. Пусть порядок элемента s группы есть число $p{\cdot}q$, где p и q взаимно просты. Доказать, что в группе найдутся такие элементы u и v, для которых выполняются равенства $s=uv=vu$, $u^p=v^q=1$.

7. Пусть для четырех элементов группы u_1, u_2, v_1, v_2 выполняются равенства: $u_1v_1=v_1u_1=u_2v_2=v_2u_2$, $u_1^p=u_2^p=v_1^q=v_2^q=1$, где p и q – взаимно простые натуральные числа. Доказать, что $u_1=u_2$, $v_1=v_2$.

8. Выяснить, каковы группы, у которых множество всех подгрупп:

а) состоит из одной подгруппы;

Удалено: а

б) состоит из двух подгрупп;

в) состоит из трех подгрупп.

9. Покажите, что любую подстановку из S_n можно представить произведением циклических подстановок без общих элементов, причем это представление единственно с точностью до порядка множителей.

10. Какова структура циклической группы, порождённой подстановкой (1,2,3,4,5,6,7,8,9)? Сколько в ней элементов и какие подгруппы она содержит?

5. ФУНКЦИИ АЛГЕБРЫ ЛОГИКИ

5.1. Основные определения

Булевой (логической) переменной называют переменную, принимающую значение из множества {0,1}. Название «логическая» следует из того, что её значения трактуются чаще всего как «*истина*» (для 1) и «*ложь*» (для 0).

Функцией алгебры логики (переключательной или булевой функцией) от n переменных называют однозначное отображение множества всевозможных наборов значений n булевых переменных в множество {0,1}.

Такую функцию можно представить в виде таблицы из n+1 столбцов и 2^n строк. Эта таблица называется *таблицей истинности*.

Наборы значений переменных располагают в *лексикографическом порядке* (в порядке возрастания), как в примере в табл. 5.1 для n = 3.

Число всевозможных наборов значений переменных составляет N=2$^{\boldsymbol{n}}$. Число различных функций, которые могут быть записаны в таблице, равно 2^N.

Второй способ описания функции состоит в том. что перечисляются наборы значений, на которых функция равна 1 (множество T1), или равна 0 (множество T0).

Для приведённого примера функцию можно представить как T1={001,010,100, 111}.

Третий способ описания – представление функций в виде вектора. Так как порядок перечисления наборов входных переменных установлен, то достаточно указать только столбец функции. Для приведенного примера это будет вектор <01101001> .

Таблица 5.1

x1	x2	x3	f
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Определение. Булева функция *существенно зависит от переменной x_i*, если найдутся два набора значений переменных, отличающиеся только i-й компонентой, на которых значения функции не совпадают. Переменная, от которой функция существенно не зависит, называется *несущественной* или *мнимой* для данной функции.

Пример. Пусть функция на наборах значений переменных <00110> и <01110> равна, соответственно, 1 и 0. Эта функция существенно зависит от второй переменной, потому что её значение на этих наборах определяется *только* значением этой переменной.

Удалено: Пример

Будем считать, что функция не изменится, если в нее добавить или из нее убрать любое количество несущественных переменных.

Таблица 5.2

x	0	x	$\bar{x}$	1
0	0	0	1	1
1	0	1	0	1

Среди четырех функций одной переменной, приведенных в табл.5.2, две функции, первая и последняя, являются константами (не зависящими ни от одной переменной). Их обозначают соответственно как *0* и *1*. Вторая функция повторяет переменную. Третья противоположна ей, называется *инверсией* и обозначается $\bar{x}$.

Пусть [*n*] – число функций, существенно зависимых от *n* переменных. Тогда [1] = 2, [0] = 2. Для любого *n* это число можно подсчитать по рекуррентной формуле

$[n] = 2^N - [0] - C_n^1[1] - C_n^2[2] - .... C_n^{n-1}[n-1].$

Здесь $N=2^n$, первая компонента – число функций от *n* переменных, из которого последовательно для *i*=0,1,..., *n*-1 вычитаются произведения числа функций, существенно зависимых от i переменных, на число способов, которыми можно выбрать i переменных из *n*.

Так, для *n* = 2 [2]=16-2-2·2=10. Для *n*=3 [3]=256-2-3·2-3·10=218.

5.2. Простейшие функции

Простейшими называют функции от двух переменных. В табл.5.3 приведены все функции, существенно зависящие от двух переменных. Для восьми из них введены названия и обозначения в табл. 5.4.

Таблица 5.3.

x y	1	2	3	4	5	6	7	8	9	10
0 0	0	0	0	0	0	1	1	1	1	1
0 1	0	0	1	1	1	0	0	0	1	1
1 0	0	1	0	1	1	0	0	1	0	1
1 1	1	0	0	0	1	0	1	1	1	0

Задание. Объясните, почему остальные 6 функций не вошли в таблицу.

Таблица 5.4

Номер	Обозначение.	Название
1	x&y	*конъюнкция*
4	x⊕y	*сложение по модулю 2*
5	x∨y	*дизъюнкция*
6	x↑y	*стрелка Пирса (функция Вебба)*
7	x≈y	*эквивалентность*
8	y→x	*импликация из y в x*
9	x→y	*импликация из x в y*
10	x|y	*штрих Шеффера*

Функция *конъюнкция* ещё называется функция И или *логическое умножение* и обозначается x&y=x∧y= x·y= xy

Функцию *дизъюнкции* ещё называют функцией ИЛИ (*логическим сложением).*

С помощью простейших можно строить более сложные функции, заменяя переменные функциями. В результате функции сопоставится формула, задающая последовательность выполнения операций. Для определения порядка вычисления функций можно использовать скобки. Например, функция, приведенная в табл. 5.1, может быть описана формулой

$f = (x1 \oplus (x2 \oplus x3))$.

Свойства простейших функций.

Между операциями над множествами, описанными в главе 1, и некоторыми простейшими функциями можно провести следующую аналогию. Для множества *A* сопоставим каждому элементу универсального множества функцию a=1, если $a \in A$, и $a=0$, если $a \notin A$. Точно также переменную можно связать с любым множеством. Тогда имеет место следующие условия. Для элементов из $\overline{A}$ функция $\overline{a}=1$, для элементов $A \cap B$ – $a \cdot b=1$, для элементов $A \cup B$ $a \vee b=1$. Значит, свойства операций, рассмотренных в главе 1, будут верны и для простейших функций.

Ниже перечислены свойства простейших функций, в истинности которых просто убедиться элементарной проверкой с помощью перебора.

- *Коммутативность* функций конъюнкции, дизъюнкции, сложения по модулю 2: $x \vee y = y \vee x$.
- *Ассоциативность* этих же функций: $(x \vee y) \vee z = x \vee (y \vee z) = x \vee y \vee z$.
- *Дистрибуция*: $x \vee (y \cdot z) = (x \vee y) \cdot (x \vee z)$, $x \cdot (y \vee z) = (xy) \vee (xz)$, $x(y \oplus z) = (xy) \oplus (xz)$.
- *Правило де Моргана*: $\overline{(x \vee y)} = \overline{x} \cdot \overline{y}$, $\overline{(x \cdot y)} = \overline{x} \vee \overline{y}$.
- *Свойства констант*: $x \cdot 0=0$, $x \cdot 1=x$, $x \vee 0=x$, $x \vee 1=1$, $x \oplus 1=\overline{x}$, $x \oplus x=0$, $x \vee \overline{x}=1$, $x \cdot \overline{x}=0$.

Используя эти свойства, можно преобразовывать формулы, удаляя лишние элементы, раскрывая скобки, вынося элементы за знаки скобок и т.п.

5.3. Дизъюнктивные нормальные формы и теорема о разложении

Удалено: ¶ ¶ ¶ ¶

Введём понятие степени булевой переменной. Будем считать, что $x^1=x$, $x^0=\overline{x}$.

Из определения следует, что $x^\alpha=1$, если $x=\alpha$, и $x^\alpha=0$, если $x \neq \alpha$. В справедливости этого можно убедиться простым перебором значений.

Конъюнкция $x1^{\alpha 1} x2^{\alpha 2} \ldots xn^{\alpha n}$ называется *элементарной*, если в ней каждая переменная встречается не более одного раза.

Удалено: 2

Удалено: n

Рангом элементарной конъюнкции называется число букв, образующих эту конъюнкцию.

Дизъюнктивной нормальной формой (ДНФ) называется дизъюнкция элементарных конъюнкций.

Длиной ДНФ называется число элементарных конъюнкций, образующих эту ДНФ. *Длину ДНФ будем обозначать буквой L.*

Дизъюнктивная нормальная форма, имеющая наименьшую длину по сравнению со всеми другими ДНФ, эквивалентными данной функции, называется *кратчайшей* ДНФ (КДНФ).

Дизъюнктивная нормальная форма, содержащая наименьшее число букв $x_i^{\alpha i}$ по сравнению со всеми другими ДНФ, эквивалентными данной функции, называется *минимальной* ДНФ (МДНФ).

Теорема о разложении. Для любой функции $\mathbf{f}(x_1,x_2,\dots,x_n,)$

$$f(x_1,x_2,\dots,x_i,\dots,x_n)=\bigvee_{\forall\alpha_1\alpha_2\dots\alpha_i} x1^{\alpha 1}\cdot x2^{\alpha 2}\dots xi^{\alpha i}\cdot f(\alpha_1,\alpha_2,\dots,\alpha_i,x_{i+1}\dots,x_n)$$

Доказательство. Возьмём произвольный набор значений переменных $(\beta_1,\beta_2,\dots,\beta_n)$ и подставим его в выражение справа и слева от равенства. Получим

$$f(\beta_1,\beta_2,\dots,\beta_n)=\bigvee_{\forall\alpha_1\alpha_2\dots\alpha_i}\beta_1^{\alpha 1}\cdot\beta_2^{\alpha 2}\dots\beta_i^{\alpha i}\cdot f(\alpha_1,\alpha_2,\dots,\alpha_i,\beta_{i+1},\dots,\beta_n)$$

Справа для любого набора значений $(\alpha_1, \alpha_2, \dots, \alpha_i)$**,** не равного $(\beta_1,\beta_2,\dots,\beta_i)$, соответствующая конъюнкция будет равна нулю, так как она в силу условия $x^{\alpha} = 0$**,** если $x \neq \alpha$**,** будет содержать нулевой сомножитель. Останется единственная конъюнкция, где $(\alpha_1, \alpha_2, \dots, \alpha_i)=(\beta_1, \beta_2, \dots, \beta_i)$, т.е. получим равенство $f(\beta_1,\beta_2,\dots,\beta_n)= f(\beta_1,\beta_2,\dots,\beta_n)$, что и доказывает теорему.

Удалено: то есть

Следствия из теоремы.

1. $f(x_1,x_2\ \dots)=x_1 f(x_1=1)\vee \bar{x}_1\cdot f(x_1=0)$. Это равенство известно как формула разложения К. Шеннона.

Формат: Список

2. $$f(x_1,x_2,\dots,x_n)=\bigvee_{\forall\alpha_1\alpha_2\dots\alpha_n} x_1^{\alpha 1}\cdot\dots x_n^{\alpha n}\cdot f(\alpha_1,\alpha_2,\dots,\alpha_n) =\bigvee_{\forall(\alpha_1\alpha_2\dots\alpha_n)\in T1} x_1^{\alpha 1}\cdot x_2^{\alpha 2}\dots x_n^{\alpha n}\cdot$$

Удалено:

Удалено:

Это равенство получается из формулы при $i=n$, здесь $T1$ –множество наборов значений аргументов, на которых функция равна 1. Оно известно как представление функции в виде *совершенной дизъюнктивной нормальной формы (СДНФ).*

Совершенная дизъюнктивная нормальная форма состоит из элементарных конъюнкций ранга n, т.е. конъюнкций наибольшего возможного для данной функции ранга, поэтому с этой точки зрения СДНФ является наиболее сложной.

Удалено: то есть

Любую функцию можно представить в виде СДНФ и это представление единственно, так как оно однозначно сопоставляется таблице истинности функции.

Пример. Найдём разложение функции $f=(a\oplus b)\to \bar{a}c$ по переменной a:
$F=a\cdot f(a=1)\vee \bar{a}\cdot f(a=0)=a\cdot((1\oplus b)\to 0)\vee \bar{a}\cdot((0\oplus b)\to c)=a\cdot(\bar{b}\to 0)\vee \bar{a}(b\to c)$.
Здесь мы учли, что $(1\oplus x)=\bar{\mathbf{x}}$, $(0\oplus x)=x$. Если ещё учесть, что $(x\to 0)=\bar{x}$, что следует из таблицы истинности импликации, то получим окончательную формулу для функции $f=a\cdot b\vee \bar{a}\cdot(b\to c)$.

Таблица 5.5

x	y	x→y
0	0	1
0	1	1
1	0	0
1	1	1

Пример. Представим функцию импликации в виде СДНФ, для чего запишем её табличное представление (табл.5.5). Здесь в множестве *T*1 три набора 000, 001 и 111, поэтому в СДНФ будет три конъюнкции $x\to y=\bar{x}\cdot\bar{y}\vee\bar{x}\cdot y\vee x\cdot y$

5.4 Минимизация функций в классе ДНФ

Проблема простейшего представления функций в заданном базисе связана с изучением свойств функций этого базиса. В настоящее время существенные результаты получены для базиса, состоящего из отрицания, конъюнкции и дизъюнкции. Именно этот базис и будет рассматриваться в дальнейшем.

Любая функция алгебры логики может быть записана в виде СДНФ. То, что запись в СДНФ часто является неэкономной, видно из примера.

Пример 1. Пусть задана СДНФ

$\bar{x}_2(x_1,x_2,x_3)=x_1x_2x_3\vee x_1x_2\bar{x}_3\vee x_1\bar{x}_2x_3\vee x_1\bar{x}_2\bar{x}_3\vee\bar{x}_1x_2x_3$.

Преобразуем эту СДНФ. Добавим еще один конъюнктивный член $x_1x_2x_3$. Это добавление не меняет данной функции, так как $x\vee x=x$,
$f(x_1,x_2,x_3)=x_1x_2x_3\vee x_1x_2\bar{x}_3\vee x_1\bar{x}_2x_3\vee x_1\bar{x}_2\bar{x}_3\vee x_1x_2x_3\vee\bar{x}_1x_2x_3$.

Преобразуем это выражение, используя равенства: $a\bar{b}\vee ab=a$ (операция склеивания) и $a\cdot b\vee a=a$, $a\cdot b\vee\bar{a}=b\vee\bar{a}$ (операция поглощения).

Получим:
$f(x_1,x_2,x_3)=x_1x_2(x_3\vee\bar{x}_3)\vee x_1\bar{x}_2(x_3\vee\bar{x}_3)\vee x_2x_3(x_1\vee\bar{x}_1)=x_1x_2\vee x_1\bar{x}_2\vee x_2x_3$.
Аналогично предыдущему делаем дальнейшие преобразования:

$f(x_1,x_2,x_3)=x_1(x_2\vee\bar{x}_2)\vee x_2x_3=x_1\vee x_2x_3$.

Если к исходной ДНФ применять лишь операции склеивания и поглощения, то наступит момент, когда эти преобразования окажутся уже невозможными. В этом случае получим тупиковую ДНФ (ТДНФ). Среди множества ТДНФ содержится и МДНФ. Получая всевозможные тупиковые ДНФ и сравнивая их по числу букв, можно найти все минимальные формы

Удалено: ¶ ¶
Удалено: Пример…Рассмотрим …ю… … …,…найдём её разложение … …..…) … … …¶ … . ... [4]
Удалено: Пример…¶ ... [5]
Удалено: Пример
Удалено: x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x ... [6]
Удалено: x…x…x ... [7]
Удалено: f…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x ... [8]
Удалено: a…b…a…b…a…a b…a…a…a…b…a…b… ... [9]
Удалено: f…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x ... [10]
Удалено: f…x…x…x…x…x…x…x…x…x…x ... [11]

для данной функции. Величина необходимого перебора определяется числом элементов класса тупиковых ДНФ для функции алгебры логики, зависящей от *n* аргументов, и может быть очень большой.

5.5. Минимизация функций

5.5.1. Метод минимизации по картам Карно

Данный метод минимизации применим для функций с числом переменных не более 6 и удобен для ручной минимизации, когда человек видит те комбинации, которые можно объединить вместе. Рассмотрим его на конкретном примере.

Пример 2. Рассмотрим функцию

$f_1(x_1,x_2,x_3)=x_1x_2x_3 \vee x_1x_2\bar{x}_3 \vee x_1\bar{x}_2x_3 \vee x_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1x_2x_3$.

Множество переменных разобьем на две группы. Одной группе сопоставим строки таблицы, второй — столбцы, так чтобы каждой клетке соответствовала комбинация переменных из этих групп. Карта Карно для нее имеет вид табл. 5.6.

Таблица 5.6

x_1x_2/x_3	0	1
00		
01		1_5
11	1_2	1_1
10	1_4	1_3

При составлении карты Карно строки именуются всевозможными комбинациями значений переменных первой группы так, чтобы расстояние между соседними комбинациями было равно 1.

Для нашего случая *0<u>0</u>→<u>0</u>1→1<u>1</u>→10* (при каждом последующем переходе изменяется только подчеркнутый символ). Аналогично именуются столбцы таблицы.

Заполнение карты производится по таблице соответствия исходной функции. В примере конъюнкции $x_1x_2x_3$ соответствует клетка 11/1, а $x_1x_2\bar{x}_3$ клетка 11/0 и т.д. В данной таблице каждая единица имеет порядковый индекс, который соответствует порядковому номеру данной компоненты в исходной функции (расстановка этих индексов совершенно не обязательна и здесь приведена для лучшего понимания).

Для минимизации необходимо попарно “склеить” рядом стоящие единицы, имеющие хотя бы одну общую компоненту. При этом надо стремиться “склеить” в один набор как можно больше клеток. В данном примере мы можем “склеить” *11,12,13,14* вместе. Это запишется как x_1,так как содержимое всех этих клеток зависит только от x_1 и не меняется при изменении x_2 или x_3. На следующем шаге склеим *11* и *15*. В результате получим x_2x_3. Рассуждения аналогичны: при изменении x_1 изменения ячеек с *11* и *15* не происходит.

Результирующей минимальной записью исходной функции будет

$f(x_1,x_2,x_3)=x_1 \vee x_2x_3$.

Пример 3. Минимизируем функцию пяти переменных:

$f_2(x_1,x_2,x_3,x4,x5)=x_1\bar{x}_2\bar{x}_3\bar{x}4 \vee x_1\bar{x}_2\bar{x}_3x4 \vee \bar{x}_1x_2x_3x4 \vee \bar{x}_2x_1\bar{x}5$.

Карта Карно для нее приведена в табл.5.7.

Таблица 5.7

$x_4x_5\backslash x_1x_2x_3$	000	001	011	010	110	111	101	100
00							1_4	$1_{1,4}$
01								1_1
11			1_3					1_2
10			1_3				1_4	$1_{2,4}$

Если в конъюнкции переменная не присутствует, то 1 ставится во все клетки, удовлетворяющие присутствующим переменным. Так, например, первой конъюнкции соответствует две клетки: 100/00 и 100/01.

Минимизация приводит к формуле

$f_2(x_1,x_2,x_3,x_4,x_5)=x_1\bar{x}_2\bar{x}_3\vee\bar{x}_1x_2x_3x_4\vee\bar{x}_2x_1\bar{x}_5$.

Пример 4. Рассмотрим функцию

$f_3(x_1,x_2,x_3)=x_1x_2\bar{x}_3\vee x_1\bar{x}_2x_3\vee x_1\bar{x}_2\bar{x}_3\vee\bar{x}_1x_2x_3\vee\bar{x}_1x_2\bar{x}_3\vee\bar{x}_1\bar{x}_2x_3$.

Таблица 5.8

x_1x_2/x_3	0	1
00		1
01	1	1
11	1	
10	1	1

По карте Карно табл.5.8 видно, что для данной функции существует две минимальных формы:

$f(x_1,x_2,x_3)=\bar{x}_1x_3\vee x_2\bar{x}_3\vee x_1\bar{x}_2$,

$f(x_1,x_2,x_3)=\bar{x}_1x_2\vee\bar{x}_2x_3\vee x_1\bar{x}_3$.

5.5.2. Метод неопределенных коэффициентов

Метод может быть применен для минимизации функции любого числа аргументов, однако для простоты изложения и большей наглядности рассмотрим его на примере минимизации функции трех аргументов.

Представим функцию $f(x_1,x_2,x_3)$ в виде следующей ДНФ:

$$f(x_1,x_2,x_3=K_1^1x_1\vee K_1^0\bar{x}_1\vee K_2^1x_2\vee K_2^0\bar{x}_2\vee K_3^1x_3\vee K_3^0\bar{x}_3\vee$$
$$\vee K_{12}^{11}x_1x_2\vee K_{12}^{10}x_1\bar{x}_2\vee K_{12}^{01}\bar{x}_1x_2\vee K_{12}^{00}\bar{x}_1\bar{x}_2\vee$$
$$\vee K_{13}^{11}x_1x_3\vee K_{13}^{10}x_1\bar{x}_3\vee K_{13}^{01}\bar{x}_1x_3\vee K_{13}^{00}\bar{x}_1\bar{x}_3\vee$$
$$\vee K_{23}^{11}x_2x_3\vee K_{23}^{10}x_2\bar{x}_3\vee\vee K_{23}^{01}\bar{x}_2x_3\vee K_{23}^{00}\bar{x}_2\bar{x}_3\vee$$
$$\vee K_{123}^{111}x_1x_2x_3\vee K_{123}^{110}x_1x_2\bar{x}_3\vee K_{123}^{101}x_1\bar{x}_2x_3\vee$$
$$\vee K_{123}^{100}x_1\bar{x}_2\bar{x}_3\vee K_{123}^{011}\bar{x}_1x_2x_3\vee K_{123}^{010}x_1\bar{x}_2x_3\vee$$
$$\vee K_{123}^{001}\bar{x}_1\bar{x}_2x_3\vee K_{123}^{000}\bar{x}_1\bar{x}_2\bar{x}_3$$

Здесь представлены всевозможные конъюнктивные члены, которые могут входить в ДНФ функции $f(x_1,x_2,x_3)$. Коэффициенты K с различными индексами являются неопределенными и подбираются так, чтобы получающаяся после этого дизъюнктивная форма была минимальной. Если задавать всевозможные наборы значений аргументов $<x_1, x_2, x_3>$ и приравнивать полученное после этого выражение (отбрасывая нулевые конъюнкции) значению функции на выбранных наборах, получим систему 2^n уравнений для определения коэффициентов K:

Удалено: .
Удалено: f…x…x…x…x…x…x…x…x…x…x…x…x…x… ... [13]
Удалено: Пример
Удалено: f…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x…x… ... [14]
Удалено: ¶ Табл. 5.8 ... [15]
Удалено: x1x2/x3 ... [16]
Удалено: в
Удалено: f…x…x…x…x…x…x…x…x…x ... [17]
Удалено: f…x…x…x…x…x…x…x…x…x ... [18]
Удалено: f…x…x…x ... [19]

$$\left.\begin{array}{l}
K_1^1 \vee K_2^1 \vee K_3^1 \vee K_{12}^{11} \vee K_{13}^{11} \vee K_{23}^{11} \vee K_{123}^{111} = f(1,1,1) \\
K_1^1 \vee K_2^1 \vee K_3^0 \vee K_{12}^{11} \vee K_{13}^{10} \vee K_{23}^{10} \vee K_{123}^{110} = f(1,1,0) \\
K_1^1 \vee K_2^0 \vee K_3^1 \vee K_{12}^{10} \vee K_{13}^{11} \vee K_{23}^{01} \vee K_{123}^{101} = f(1,0,1) \\
K_1^1 \vee K_2^0 \vee K_3^0 \vee K_{12}^{10} \vee K_{13}^{10} \vee K_{23}^{00} \vee K_{123}^{100} = f(1,0,0) \\
K_1^0 \vee K_2^1 \vee K_3^1 \vee K_{12}^{01} \vee K_{13}^{01} \vee K_{23}^{11} \vee K_{123}^{011} = f(0,1,1) \\
K_1^0 \vee K_2^1 \vee K_3^0 \vee K_{12}^{01} \vee K_{13}^{00} \vee K_{23}^{10} \vee K_{123}^{010} = f(0,1,0) \\
K_1^0 \vee K_2^0 \vee K_3^1 \vee K_{12}^{00} \vee K_{13}^{01} \vee K_{23}^{01} \vee K_{123}^{001} = f(0,0,1) \\
K_1^0 \vee K_2^0 \vee K_3^0 \vee K_{12}^{00} \vee K_{13}^{00} \vee K_{23}^{00} \vee K_{123}^{000} = f(0,0,0)
\end{array}\right\}$$

Пусть таблично задана некоторая функция $f(x_1,x_2,x_3)$. Если набор $<x_1,x_2,x_3>$ таков, что функция на этом наборе равна 0, то в правой части соответствующего уравнения будет стоять 0. Для удовлетворения этого уравнения необходимо приравнять 0 все коэффициенты K, входящие в левую часть рассматриваемого уравнения.

В уравнениях, где справа стоят единицы, вычеркнем слева все нулевые коэффициенты. Из оставшихся коэффициентов приравняем единице коэффициенты, определяющие конъюнкции наименьшего возможного ранга, а остальные примем равными 0 (это можно сделать, так как дизъюнкция обращается в 1, если хотя бы один член ее равен *1*). Единичные коэффициенты K^i определят минимальную ДНФ.

Пример 5. $f(x_1,x_2,x_3)=x_1x_2x_3 \vee x_1x_2\bar{x}_3 \vee x_1\bar{x}_2x_3 \vee x_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2\bar{x}_3$.
Составляем систему:

$$\left.\begin{array}{l}
K_1^1 \vee K_2^1 \vee K_3^1 \vee K_{12}^{11} \vee K_{13}^{11} \vee K_{23}^{11} \vee K_{123}^{111} = 1 \\
K_1^1 \vee K_2^1 \vee K_3^0 \vee K_{12}^{11} \vee K_{13}^{10} \vee K_{23}^{10} \vee K_{123}^{110} = 1 \\
K_1^1 \vee K_2^0 \vee K_3^1 \vee K_{12}^{10} \vee K_{13}^{11} \vee K_{23}^{01} \vee K_{123}^{101} = 1 \\
K_1^1 \vee K_2^0 \vee K_3^0 \vee K_{12}^{10} \vee K_{13}^{10} \vee K_{23}^{00} \vee K_{123}^{100} = 1 \\
K_1^0 \vee K_2^1 \vee K_3^1 \vee K_{12}^{01} \vee K_{13}^{01} \vee K_{23}^{11} \vee K_{123}^{011} = 0 \\
K_1^0 \vee K_2^1 \vee K_3^0 \vee K_{12}^{01} \vee K_{13}^{00} \vee K_{23}^{10} \vee K_{123}^{010} = 0 \\
K_1^0 \vee K_2^0 \vee K_3^1 \vee K_{12}^{00} \vee K_{13}^{01} \vee K_{23}^{01} \vee K_{123}^{001} = 0 \\
K_1^0 \vee K_2^0 \vee K_3^0 \vee K_{12}^{00} \vee K_{13}^{00} \vee K_{23}^{00} \vee K_{123}^{000} = 1
\end{array}\right\}$$

Из пятого, шестого и седьмого уравнений вытекает, что
$K_1^0=K_2^0=K_2^1=K_3^0=K_3^1=K_{12}^{00}=K_{12}^{01}=K_{13}^{00}=K_{13}^{01}=K_{23}^{01}=K_{23}^{10}=K_{23}^{11}=K_{123}^{001}=$
$=K_{123}^{010}=K_{123}^{011}=0$.

Удалено: Пример
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x

В результате данная система примет вид

$$\left.\begin{aligned}
&K_1^1 \vee K_{12}^{11} \vee K_{13}^{11} \vee K_{123}^{111} = 1\\
&K_1^1 \vee K_{12}^{11} \vee K_{13}^{10} \vee K_{123}^{110} = 1\\
&K_1^1 \vee K_{12}^{10} \vee K_{13}^{11} \vee K_{123}^{101} = 1\\
&K_1^1 \vee K_{12}^{10} \vee K_{13}^{10} \vee K_{23}^{00} \vee K_{123}^{100} = 1\\
&K_{23}^{00} \vee K_{123}^{000} = 1
\end{aligned}\right\}$$

Приравняем 0 в каждом уравнении все коэффициенты, кроме тех, которые отвечают конъюнкциям, содержащим наименьше число переменных: $K_{12}^{11}=K_{12}^{01}=K_{13}^{11}=K_{13}^{10}=K_{123}^{111}=K_{123}^{110}=K_{123}^{101}=K_{123}^{100}=K_{123}^{000}=0$.

После этого получаем систему

$$\left.\begin{aligned}
&K_1^1 = 1\\
&K_1^1 = 1\\
&K_1^1 = 1\\
&K_1^1 \vee K_{23}^{00} = 1\\
&K_{23}^{00} = 1
\end{aligned}\right\}$$

Отсюда находим МДНФ для данной функции: $f(x_1,x_2,x_3)=x_1 \vee \bar{x}_2\ \bar{x}_3$.

Описанный метод эффективен лишь для минимизации функций, число аргументов в которых не больше 5-6. Это связано с тем, что число уравнений равно 2^n.

5.5.3. Метод Квайна — Мак-Класки

Более эффективным является выписывание для функции n переменных не всех возможных конъюнкций, а только тех, которые могут присутствовать в ДНФ, эквивалентной минимизируемой функции. Этот прием позволяет уменьшить таблицу и перебор, который мы проводим с целью нахождения МДНФ.

Предположим, что минимизируемая функция задана в СДНФ. Для простоты будем называть элементарные конъюнкции ранга n, входящие в СДНФ минимизируемой функции, *минитермами* ранга n. Метод состоит из последовательного выполнения следующих этапов.

1. *Нахождение первичных импликант*. Все минитермы данной функции сравнивают между собой попарно. Если минитермы m_i и m_j таковы, что они имеют вид ax_i и $a\bar{x}_i$, то вписывается конъюнкция a, являющаяся минитермом $(n\text{-}1)$-го ранга. Будем говорить, что минитерм a покрывает минитерм ax и $a\bar{x}$ (покрывается ими). Минитермы n-го ранга, для которых произошло склеивание, отмечаются. После построения всех минитермов $(n\text{-}1)$-го ранга вновь сравнивают их попарно, выписывают минитермы $(n\text{-}2)$-го ранга, отмечают склеивающиеся минитермы $(n\text{-}1)$-го

ранга и т.д. Этап заканчивается, когда вновь полученные минитермы *i*-го ранга уже не склеиваются между собой. Все неотмеченные минитермы называются *первичными* или *простыми импликантами*.

Пример 6

Удалено: Пример

$$f(x_1,x_2,x_3,x_4)=\bar{x}_1\ \bar{x}_2x_3x_4\vee\bar{x}_1x_2\ \bar{x}_3\ \bar{x}_4\vee\bar{x}_1x_2\ \bar{x}_3x_4\vee$$
$$\vee\ \bar{x}_1x_2x_3x_4\vee x_1\ \bar{x}_2\ \bar{x}_3x_4\vee x_1\ \bar{x}_2x_3x_4\vee x_1x_2\ \bar{x}_3\ \bar{x}_4\vee x_1x_2x_4\ \bar{x}_3.$$

Минитермы 4-го ранга:

$x_1\ \bar{x}_2x_3x^*_4$, $\bar{x}_1x_2\ \bar{x}_3\ \bar{x}^*_4$, $\bar{x}_1\ \bar{x}_3x_2x^*_4$, $\bar{x}_1x_2x_3x^*_4$,

$x_1\ \bar{x}_2\ \bar{x}_3x^*_4$, $x_1\ \bar{x}_2x_3x^*_4$, $x_1x_2\ \bar{x}_3\ \bar{x}^*_4$, $x_1x_2\ \bar{x}_3x^*_4$.

Образуем минитермы 3-го ранга:

$\bar{x}_1x_3x_4$, $\bar{x}_2x_3x_4$, $\bar{x}_1x_2\ \bar{x}^*_3$, $x_2\ \bar{x}_3\ \bar{x}^*_4$,

$\bar{x}_1x_2x_4$, $x_2\ \bar{x}_3x^*_4$, $x_1\ \bar{x}_2x_4$, $x_1\ \bar{x}_3x_4$, $x_1x_2\ \bar{x}^*_3$.

Теперь находим минитермы 2-го ранга: $x_2\ \bar{x}_3$.

Дальнейшее склеивание невозможно, этап получения простых импликант закончен. Простыми импликантами являются минитермы:

$\bar{x}_1x_3x_4$, $\bar{x}_2x_3x_4$, $\bar{x}_1x_2x_4$, $x_1\ \bar{x}_2x_4$, $x_1\ \bar{x}_3x_4$, $x_2\ \bar{x}_3$.

2. Расстановка меток. Для данной функции $f(x_1,x_2,...,x_n)=\vee_i\lambda_i$, где λ_i - простые импликанты, полученные нами на первом этапе. Для нахождения МДНФ нужно найти минимальное подмножество из λ_i, покрывающее конъюнкции исходной СДНФ, для чего необходимо выбросить некоторое количество первичных импликант. На данном этапе составляется таблица, число строк которой равно числу полученных первичных импликант минимизируемой функции, число столбцов совпадает с числом минитермов. Если в некоторый минитерм входит какая-либо из первичных импликант, то на пересечении соответствующего столбца и строки ставится метка.

Продолжение примера 6. Составим таблицу (табл.5.9) с метками для рассматриваемой функции, в которой будет 6 строк и 8 столбцов.

Таблица 5.9

	$\bar{x}_1\ \bar{x}_2x_3x_4$	$\bar{x}_1x_2\ \bar{x}_3\ \bar{x}_4$	$\bar{x}_1x_2\ \bar{x}_3x_4$	$\bar{x}_1x_2x_3x_4$	$x_1\ \bar{x}_2\ \bar{x}_3x_4$	$x_1\ \bar{x}_2x_3x_4$	$x_1x_2\ \bar{x}_3\ \bar{x}_4$	$x_1x_2x_4\ \bar{x}_3$
$\bar{x}_1x_3x_4$	∨			∨				
$\bar{x}_2x_3x_4$	∨					∨		
$\bar{x}_1x_2x_4$			∨	∨				
$x_1\ \bar{x}_2x_4$					∨	∨		
$x_1\ \bar{x}_3x_4$					∨			∨
$x_2\ \bar{x}_3$		∨	∨				∨	∨

3. *Нахождение существенных импликант*. Если в каком-либо из столбцов составленной таблицы имеется только одна метка, то первичная

импликанта, стоящая в соответствующей строке, называется *существенной импликантой*. Существенная импликанта не может быть исключена из правой части, так как без нее не будет получено покрытие всего множества *T1* данной функции. Поэтому из таблицы меток исключаются строки, соответствующие существенным импликантам, и столбцы минитермов, покрываемых этими существенными импликантами.

Продолжение примера 6. Для нашей функции существенной импликантой является $x_2\ \bar{x}_3$. Она покрывает минитермы $\bar{x}_1 x_2\ \bar{x}_3\ \bar{x}_4$, $\bar{x}_1 x_2\ \bar{x}_3\ \bar{x}_4$, $x_1 x_2\ \bar{x}_3\ \bar{x}_4$, $x_1 x_2 x_4\ \bar{x}_3$. При переходе к следующему этапу эти минитермы будут вычеркнуты.

4. *Вычеркивание лишних столбцов*. Исследуется таблица, полученная после третьего этапа. Если в ней есть два столбца, в которых используются метки в одинаковых строках, то один из них вычеркивается. Это можно сделать в силу того, что покрытие оставшегося столбца будет осуществлять покрытие выброшенного минитерма.

Продолжение примера 6. После вычеркивания существенной импликанты и минитермов, которые она покрывает, таблица меток принимает вид, как в табл.5.10:

Таблица 5.10

	$\bar{x}_1\ \bar{x}_2 x_3 x_4$	$\bar{x}_1 x_2 x_3 x_4$	$x_1\ \bar{x}_2\ \bar{x}_3 x_4$	$x_1\ \bar{x}_2 x_3 x_4$
$\bar{x}_1 x_2 x_4$	$\vee$	$\vee$		
$\bar{x}_2 x_3 x_4$	$\vee$			$\vee$
$\bar{x}_1 x_2 x_4$		$\vee$		
$x_1\ \bar{x}_2 x_4$			$\vee$	$\vee$
$x_1\ \bar{x}_3 x_4$			$\vee$	

В данном случае одинаковых столбцов нет.

5. *Вычеркивание лишних первичных импликант*. Если после выбрасывания некоторых столбцов на четвертом этапе в таблице появляются строки, в которых нет ни одной метки, то первичные импликанты, соответствующие этим строкам, исключаются из дальнейших рассмотрений, так как они не покрывают оставшиеся в рассмотрении минитермы.

6. *Выбор минимального покрытия максимальными интервалами*. Выбирается такая совокупность первичных импликант, которая включает метки во всех столбцах (по крайней мере, по одной в каждом столбце). При нескольких возможных вариантах отдается предпочтение варианту с

минимальным суммарным числом букв в простых импликантах, образующих покрытие.

Окончание примера 6. Для рассматриваемой функции выбираем покрытие из импликант $\bar{x}_1x_3x_4$ и $x_1\bar{x}_2x_4$, так как они совместно покрывают все оставшиеся после четвертого этапа минитермы. Минимальная ДНФ для этой функции имеет вид

$$f(x_1,x_2,x_3,x_4)=\bar{x}_1x_3x_4\vee x_1\bar{x}_2x_4\vee x_2\bar{x}_3.$$

5.6. Классы функций алгебры логики

В гл. 4 было введено понятие замкнутого множества, класса, порождающего множества класса и базиса. Рассмотрим эти понятия применительно к множеству ФАЛ.

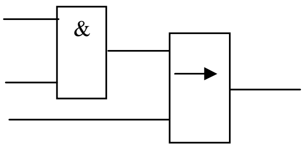
Над множеством функций F введём операцию суперпозиции функций, состоящую в замене некоторых аргументов одной из $f\in F$ на функции из *F*. В результате получается новая функция, порождённая суперпозицией. При этом будем считать, что в общем случае аргументы у функций все различны, в частном случае множества аргументов функций могут пересекаться или совпадать. Полученную функцию можно снова использовать в суперпозиции и т.д.

Пример. Пусть $F=\{x\&y,\ x\to y\}$. Результатом суперпозиции может быть функция $x\to(y\cdot z)$

Множество всех функций, которые могут быть порождены с помощью суперпозиции из функций множества *F*, назовём *классом функций*, порождённых *F*, и обозначим как $|F|$. Множество *F* называют *порождающим* множеством класса $|F|$

Порождающее множество данного класса называется *базисом*, если никакое его собственное подмножество данный класс не порождает.

Инженерная трактовка. Сопоставим множеству *F* множество элементов, реализующих функции из *F*. Тогда суперпозиции сопоставляется схема из этих элементов, множеству функций класса $|F|$ – множество всех функций, которые могут быть реализованы такими схемами.



Для рассматриваемого выше примера схема приведена на рис. 5.1.

Рис.5.1

Рассмотрим основные классы функций алгебры логики.

Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: y

5.6.1. Монотонные функции

Определение. Два набора значений двоичных переменных $\alpha=<\alpha_1,\alpha_2,\ldots,\alpha_n>$ и $\beta=<\beta_1,\beta_2,\ldots,\beta_n>$ назовём *сравнимыми* и будем писать $\alpha\geq\beta$, если $\forall_i$, $i=1,\ldots,n$ $\alpha_i \geq \beta_i$. Здесь $\geq$ понимается в обычном виде: 1>0. Если $\alpha\ngeq\beta$ и $\beta\ngeq\alpha$, наборы считаются *несравнимыми*.

Пример. Наборы α=<010111> и β=<010101> сравнимы и $\alpha\geq\beta$. Набор α и χ=<100111> несравнимы.

Определение. Функция f называется монотонной, если для любых двух наборов значений входных переменных α и β из того, что $\alpha\geq\beta$, следует, что $f(\alpha)\geq f(\beta)$.

Свойства монотонных функций.

Нулевой набор значений сравним с любым набором и является меньшим любого из них. Значит, если монотонная функция равна единице на этом наборе, то она равна единице и на любом наборе, т.е. равна константе. Точно так же, если на единичном наборе значений монотонная функция равна нулю, то она не может быть единицей ни на каком наборе, так как единичный набор больше всякого другого набора.

Пусть функция на наборе α, отличном от единичного, равна 1, и пусть значение i-ой компоненты в нём равно 0. Это значит, что на наборе, который отличается только тем, что i-ая переменная в нём равна 1, функция тоже примет единичное значение. Это означает, что конъюнкции в ДНФ, соответствующие этим наборам, можно склеить по переменной x_i. Точно так же, для набора со значением переменной 0 (т.е. с возможным значением в конъюнкции переменной с инверсией) найдётся набор со значением переменной 1, что приведёт к склеиванию по этой переменной. Следовательно, *в минимальной ДНФ монотонной функции нет переменных в инверсной форме.*

Из этого свойства можно вывести, что суперпозиция монотонных функций снова будет монотонной функцией, т.е. множества монотонных функций образует класс монотонных функций, обозначаемый как М. Базис класса М образуют обе константы и пара функций – конъюнкция и дизъюнкция, т.е. множество $\{x\cdot y,\ x\vee y,\ 0,1\}$.

Задача. Докажите, что константы должны присутствовать в базисе.

5.6.2. Самодвойственные функции

Определение. Для функции $f(x_1,x_2,\ldots,x_n)$ функция $\overline{f(\overline{x}_1, \overline{x}_2,\ldots, \overline{x}_n)}$ называется *двойственной* к ней.

Обозначим двойственную функцию как f^*.

Пример. Для функции $(x\cdot y)$ двойственной будет функция $\overline{(\overline{x}\vee \overline{y})} = x\vee y$.

Можно показать, что двойственной функцией к f^* будет функция f, значит для $x\vee y$ двойственной будет $x\cdot y$.

Двойственной к *x* будет функция, равная x, двойственной к 0 будет 1.

Определение. Функция называется *самодвойственной*, если она равна своей двойственной.

Переменная x служит примером самодвойственной функции, так же как и функция инверсии переменной.

Свойства самодвойственных функций.

1. Самодвойственная функция полностью определяется своим видом на верхней половине таблицы истинности. Действительно, если, например, значение функции на наборе $<\alpha_1, \alpha_2, \ldots, \alpha_n>$ равно 0, то значение функции на инверсном наборе $< \bar{\alpha}_1, \bar{\alpha}_2, \ldots, \bar{\alpha}_n>$ должно быть равно 1.
2. Из первого свойства вытекает, что число различных функций от n переменных равно 2^m, где $m=2^{n-1}$.
3. Построим все функции от двух переменных. Их будет 4 в соответствии с возможными значениями на верхней половине таблицы: 00,01,10,11. Эти функции приведены в таблице 5.11. Как видно из таблицы, первые две функции совпадают с переменными, две последние – с инверсиями переменных. Отсюда следует свойство: *самодвойственных функций, существенно зависящих ровно от двух переменных нет.*

Таблица 5.11

x	y	f_1	f_2	f_3	f_4
0	0	0	0	1	1
0	1	0	1	0	1
1	0	1	0	1	0
1	1	1	1	0	0

4. СДНФ самодвойственной функции будет иметь ровно 2^{n-1} конъюнкций.
5. Суперпозиция самодвойственных функций будет функция самодвойственная. Множество самодвойственных функций образуют класс, который принято обозначать как *D*. Базисом класса является функция трёх переменных $\{x\cdot \bar{y} \vee x\cdot \bar{z} \vee y\cdot \bar{z}\}$.

5.6.3. Линейные функции.

Рассмотрим класс функций, порождённый множеством $F=\{x\cdot y,\ x\oplus y,\ 1\}$.

Из того, что $x\oplus 1= \bar{x}$, следует, что в данном базисе реализуется инверсия, которая вместе с конъюнкцией даёт возможность построить любую функцию. Значит, данный базис порождает класс всех функций – класс C.

Сравним таблицы функции сложения по модулю два и дизъюнкции (табл.5.12).

Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: *x*

Таблица 5.12

a	b	$a \vee b$	$a \oplus b$
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	0

Из таблицы видно, что $a \vee b = (a \oplus b) \vee a \cdot b$.

Если a и b такие, что имеет место равенство $a \cdot b = 0$, то такие переменные называются *ортогональными*. Для ортогональных переменных $a \vee b = (a \oplus b)$.

Если рассматривать СДНФ любой функции, то можно показать, что в ней любая пара конъюнкций ортогональна. Это приводит к следующему алгоритму построения записи функции в рассматриваемом базисе.

- записать функцию в СДНФ;
- заменить в СДНФ символы дизъюнкции на символы сложения по модулю два;
- заменить все инверсии по формуле $\bar{x} = (x \oplus 1)$;
- раскрыть скобки, пользуясь свойством дистрибуции $x \cdot (y \oplus z) = x \cdot y \oplus x \cdot z$;
- сделать сокращения, используя свойство $x \oplus x = 0$, $x \oplus 0 = x$.

В результате получается запись функции в форме, которую представим в общем виде

$f = C_0 \oplus C_1 \cdot x_1 \oplus C_2 \cdot x_2 \oplus \ldots \oplus C_n \cdot x_n \oplus C(n+1) \cdot x_1 \cdot x_2 \oplus \ldots \oplus C_m \cdot x_1 \cdot x_2 \cdot \ldots \cdot x_n$, где $C_0, C_1, C_2, \ldots$, принимают значения 0 или 1.

Это представление называется *полиномом Жегалкина*, а алгебра с сигнатурой $\{F = \{x \cdot y,\ x \oplus y,\ 1\}\}$ – алгеброй Жегалкина.

Пример. Построим полином Жегалкина для функции импликации. По её таблице истинности запишем СДНФ этой функции

$$a \to b = \bar{a}\ \bar{b} \vee \bar{a}b \vee ab,$$

после замены дизъюнкций на сложение по модулю два имеем $a \to b = \bar{a}\ \bar{b} \oplus \bar{a}b \oplus ab = (a \oplus 1)(b \oplus 1) \oplus (a \oplus 1) \cdot b \oplus a \cdot b = a \cdot b \oplus a \oplus b \oplus 1 \oplus a \cdot b \oplus b \oplus a \cdot b = a \cdot b \oplus a \oplus 1$.

Определение. Функция называется *линейной*, если её полином Жегалкина не содержит конъюнкций. Общий вид линейной функции $f = C_0 \oplus C_1 \cdot x_1 \oplus C_2 \cdot x_2 \oplus \ldots \oplus C_n \cdot x_n$.

Число различных линейных функций от не более чем n переменных определяется формулой $N = 2^{n+1}$.

Суперпозиция линейных функций есть функция линейная, следовательно, множество линейных функций образуют класс линейных

Удалено: .
Удалено: b
Удалено: b
Удалено: b
Удалено: y
Удалено: x
Удалено: y
Удалено: x
Удалено: C
Удалено: C
Удалено: x
Удалено: C
Удалено: x
Удалено: C
Удалено: x
Удалено: C
Удалено: x
Удалено: x
Удалено: C
Удалено: x
Удалено: x
Удалено: x
Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: Пример
Удалено: C
Удалено: C
Удалено: x
Удалено: C
Удалено: x
Удалено: C
Удалено: x

функций, обозначаемый как L. Базисом класса L служит множество $\{x \oplus y, 1\}$.

5.6.4. Функции, сохраняющие константу

Если в функции $f(x_1, x_2, \ldots, x_n)$ вместо всех переменных поставить одну переменную x, то возможно 4 различных варианта: $f(x, x, \ldots, x) \in \{x, 0, 1, \bar{x}\}$. В зависимости от варианта функцию называют

- α-функция, если $f=x$;
- β-функция, если $f=1$;
- χ-функция, если $f=0$;
- δ-функция, если $f=\bar{x}$,

Рассмотрим множество α- и β-функций. Для них имеет место $f(1,1,\ldots,1)=1$, поэтому эти функции называют функциями, сохраняющими 1. Можно показать, что суперпозиция этих функций будет снова сохранять константу 1. Значит, множество этих функций замкнуто и образует класс функций, сохраняющих единицу. Этот класс обозначают как $C1$.

Аналогично можно показать, что множество α- и χ-функций образуют класс функций, сохраняющих константу 0, так как для них $f(0,0,\ldots,0)=0$.Этот класс принято обозначать как $C0$.

5.7. Функциональная полнота.

Определение. *Класс функций K называется предполным в C, если в C не существует класса K', чтобы имело место $K \subset K' \subset C$,* т.е. не найдётся класса, который бы включал в себя полностью данный класс и был меньше, чем *C*.

Мы выделили 5 классов функций алгебры логики. Различных классов функций гораздо больше; так, α-функции образуют свой класс, пересечение классов снова будет классом.

Пост установил, что *классы M, D, L*, *C*1 и *C*0 *являются предполными и других предполных классов в C нет*.

Результаты работ Поста позволили ответить на важный вопрос: какими свойствами должны обладать функции множества F, чтобы это множество порождало класс C. Такие множества называют *функционально полными*, поставленный вопрос известен как вопрос об условиях *функциональной полноты*.

Инженерная трактовка этого вопроса: какими свойствами должны обладать функции элементов, чтобы схемой из этих элементов можно было реализовать любые функции.

Решение задачи формулируется как теорема Поста.

Теорема. Для того, чтобы множество функций порождало класс всех функций C, необходимо и достаточно, чтобы это множество содержало, по крайней мере, одну немонотонную, одну

Удалено: x
Удалено: y
Удалено: ¶
Удалено: *x*
Удалено: *x*
Удалено: *x*
Удалено: x
Удалено: x
Удалено: x
Удалено: *x*
Удалено: *x*
Удалено: x
Удалено: x
Удалено: f
Удалено: f
Удалено: C
Удалено: C

несамодвойственную, одну нелинейную, одну не сохраняющую константу ноль и одну не сохраняющую константу единица функцию.

Необходимые условия формулируются из результатов Поста о предполных классах. Необходимо, чтобы по отношению любого из этих классов в F была функция, не принадлежащая классу. Действительно, если в рассматриваемом множестве нет, например, несамодвойственной функции, то в результате суперпозиции будут реализовываться только самодвойственные функции, т.е. нельзя получить, например, конъюнкцию, которая не является самодвойственной.

Достаточность доказывается конструктивно. Покажем, как из функций, удовлетворяющих условиям теоремы Поста, получить функции инверсию и конъюнкцию или дизъюнкцию, составляющих функционально полный базис.

Пусть множество содержит функцию f_0, не сохраняющую константу ноль, f_1, не сохраняющую константу единица, f_2 –немонотонную, f_3 – несамодвойственную и f_4 – нелинейную функцию (функции не обязательно различны).

По определению f_0 (0,0,…,0) =1. Для этой функции возможны два варианта значений f_0 (1, 1,…,1).

- Если f_0 (1, 1,…,1)=1, то функция является β-функцией, и при подстановке вместо всех аргументов одного произвольного аргумента она превращается в константу 1. Имея константу 1, из функции f_1 получаем константу 0, так как f_1(1, 1,…,1)=0.
- Если f_0 (1, 1,…,1)=0, то функция является δ-функцией, и при подстановке вместо всех аргументов одного произвольного аргумента она превращается в инверсию. В этом случае рассмотрим функцию f_3. Для неё найдется набор значений аргументов $<a_1, a_2,\dots, a_n>$, такой, что $f(a_1, a_2,\dots, a_n)= f(\bar{a}_1, \bar{a}_2,\dots, \bar{a}_n)$.

В функцию f_3 поставим произвольную переменную в прямой форме, если компонента набора a_i равна 1, и в инверсной форме, если компонента набора равна 0, Получим константу, равную $f(a_1, a_2,\dots, a_n)$. Из неё с помощью инверсии получается вторая константа.

С помощью констант из функции f_2 можно получить инверсию.

Для неё найдется два соседних набора, таких, что на меньшем наборе функция равна единице, а на большем – нулю. Если подставим в f_2 константы этого набора, где они совпадают, и произвольную переменную, где наборы отличаются, получим инверсию этой переменной.

Константы и инверсия позволяет получить конъюнкцию переменных из функции f_4. Запишем эту функцию в виде полинома Жегалкина, выделим в нём первое вхождение переменных в конъюнкцию, и все переменные, кроме переменных конъюнкции, заменим на константу 0, а переменные конъюнкции заменим произвольно на два различных аргумента, например на x и y. В результате возможны (с точностью до инверсии константы C_0 в полиноме) следующие варианты:

1. xy,
2. $xy \oplus x$,
3. $xy \oplus y$,
4. $xy \oplus x \oplus y$.

В первом случае получаем конъюнкцию, что и необходимо получить, во втором варианте полученная функция равна функции $\bar{x}y$, из которой с помощью инверсии получим конъюнкцию. То же самое можно сказать и о третьем варианте, где функция равна $\bar{y}x$. Для последнего варианта функция равна дизъюнкции.

Теорема доказана.

В табл. 5.13 показана принадлежность простейших функций к предполным классам. Здесь + означает, что функция принадлежит, x –что функция не принадлежит к классу. Здесь символом ‘ обозначена инверсия.

Из таблицы легко видеть, что функциональной полнотой обладают множества {$`x$, $x \vee y$}, {$`x$, $x \cdot y$}, {$x \cdot y$, $x \oplus y$, 1}. {$x \to y$, 1}. Особый интерес представляют две последние функции, составляющие монофункциональный базис. Такие функции, отвечающие всем условиям теоремы Поста, получили название *функций шефферовского типа*.

Теорема позволяет определить, является ли заданное множество функционально полным, если нет, то какой функции в нём не хватает. Можно решать задачу построения функции шефферовского типа от более чем двух переменных. Это должна быть δ-функция, так как она не сохраняет константы. Как δ-функция она немонотонна, и дальше нужно распределить единицы и нули в таблице так, чтобы функция была нелинейной и несамодвойственной.

Таблица 5.13

F	M	D	L	$C1$	$C0$
0	**+**	**x**	**+**	**x**	**+**
1	**+**	**x**	**+**	**+**	**x**
$‘x$	**x**	**+**	**+**	**x**	**x**
$x \vee y$	**+**	**x**	**x**	**+**	**+**
$x \cdot y$	**+**	**x**	**x**	**+**	**+**
$x \to y$	**x**	**x**	**x**	**+**	**x**
$x \oplus y$	**x**	**x**	**+**	**x**	**+**
$x \approx y$	**x**	**x**	**+**	**+**	**x**
$‘(x \cdot y)$	**x**	**x**	**x**	**x**	**x**
$`(x \vee y)$	**x**	**x**	**x**	**x**	**x**

Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: x
Удалено: x
Удалено: y
Удалено: y
Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: y
Удалено: x
Удалено: x
Удалено: x
Удалено: y
Удалено: x
Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: x
Удалено: y
Удалено: ¶

5.8. Контрольные вопросы и задания

5.8.1. Представление функций

Задача 1. Постройте таблицу значений для функций, представленных в виде следующих формул.

1. $((a\text{->}b) \oplus (b\sim c))\text{->}d$
2. $(a \oplus b)\text{->}(b\text{->}c)\ \&(d \vee bca)$;
3. $((a\text{->}b)\text{->}c) \vee \bar{a}dc) \oplus (a \oplus b)$;
4. $(a\text{->}b) \oplus (b\text{->}ac) \vee (a\text{->}c)$
5. $((a\text{->} \bar{b})\text{->}c) \vee a\ \bar{d}c) \oplus(\bar{a} \oplus b)$;
6. $((d\text{->}b)\text{->}c) \vee \bar{a}dc) \oplus (\bar{a} \oplus b)$;
7. $((d\text{->}b) \oplus(\bar{b}\sim c)) \oplus d$
8. $((a \oplus b) \oplus (b \vee d))\text{->}bd$

Удалено: ¶ Задачи ... [20]
Удалено: Вариант b…b…c…d ... [21]
Удалено: Вариант a…b…b…c…d…b…c… ... [22]
Удалено: Вариант a…b…c…a…d…c…a… ... [23]
Удалено: Вариант a…b…b…a…c…a…c ... [24]
Удалено: Вариант a…b…c…a…d…c…a… ... [25]
Удалено: Вариант d…b…c…a…d…c…a… ... [26]
Удалено: Вариант d…b…b…c…d ... [27]
Удалено: Вариант b…b…d…b…d ... [28]

5.8.2. Разложение функций

Задача 1. Для функций, описанных выше, постройте представление их в виде СДНФ.

Задача 2. Для функций, описанных в первой задаче, постройте разложение Шеннона по переменной *b* и определите существенную зависимость от переменной *c*.

5.8.3. Минимизация ФАЛ

Задача 1. Для функций, описанных в разд. 5.8.1, постройте СДНФ. Используя операции склеивания и поглощения, найдите минимальные ДНФ. Найдите минимальные ДНФ через карты Карно.

Задача 2. Для функций задачи 1 получите ДНФ следующим образом. Каждую из элементарных функций представьте в виде ДНФ, затем эти формулы подставьте в описания функций, раскройте скобки и сделайте необходимые преобразования, используя свойства функций: коммутативность, дистрибутивность, правило де Моргана и др.

Задача 3. Используя карты Карно, запишите в виде ДНФ самую сложную функцию от 4 переменных. Используя вынесение за скобки (свойство дистрибуции операций конъюнкции и дизъюнкции), запишите эту функцию в скобочной форме.

5.8.4. Классы функций

Монотонные функции

Задача 1. Выпишите все монотонные функции, существенно зависимые от трёх переменных. Если в одну группу отнести функции, получающиеся одна из другой переименованием переменных, то сколько групп будет в этом множестве? (*Пример*. Функции ($xy \vee z$) , ($xz \vee y$) и ($yz \vee x$) будут в одной группе).

Удалено: Пример

Задача 2. Являются ли монотонными следующие функции:

1. $((a \to b) \oplus (b \sim c)) \to d$
2. $(a \oplus b) \to (b \to c) \wedge (\bar{d} \vee bca)$;
3. $((a \to b) \to c) \vee \bar{a}dc) \oplus (a \oplus b)$;
4. $(\bar{a} \to b) \oplus (b \to ac) \vee (a \to c)$;
5. $((a \to \bar{b}) \to c) \vee a\bar{d}c) \oplus (\bar{a} \oplus b)$;
6. $((d \to b) \to c) \vee \bar{a}dc) \oplus (\bar{a} \oplus b)$;
7. $((d \to b) \oplus (\bar{b} \sim c)) \oplus d$;
8. $((a \oplus b) \oplus (b \vee d)) \to bd$.

Удалено: Вариант … b…b…c…d ... [29]

Удалено: Вариант a…b…b…c…d…b…c… ... [30]

Удалено: Вариант … a…b…c…a…d…c…a… ... [31]

Удалено: Вариант … a…b…b…a…c…a…c ... [32]

Удалено: Вариант … a…b…c…a…d…c…a… ... [33]

Удалено: Вариант d…b…c…a…d…c…a… ... [34]

Удалено: Вариант … d…b…b…c…d ... [35]

Удалено: Вариант … b…b…d…b…d ... [36]

Самодвойственные функции

Задача 3. Проверьте, являются ли самодвойственными функции задачи 2.

Задача 4. Перечислите все самодвойственные функции, существенно зависящие от трёх переменных.

Так же, как в задаче 1, определите, сколько групп в этом множестве.

Линейные функции.

Задача 5. Постройте полиномы Жегалкина для простейших функций.

Задача 6. Постройте полином Жегалкина для следующих функций:.

1. $((a \to b) \& (b \sim c)) \to d$
2. $(a \vee \bar{b}) \to (b \to c) \wedge (\bar{d} \vee bca)$;
3. $((a \to b) \to c) \vee \bar{a}dc) \vee (a \oplus b)$;
4. $((a \to \bar{b}) \to c) \vee \bar{a}\bar{d}c) \vee (\bar{a} \oplus b)$;
5. $((d \to b) \to c) \vee \bar{a}dc) \oplus (\bar{a} \vee b)$;
6. $((d \to b) \oplus (\bar{b} \sim c)) \vee \bar{d}$
7. $((a \oplus b) \oplus (b \vee d)) > (\bar{b} \cdot \bar{d})$

Удалено: b…b…c…d ... [37]

Удалено: a…b…b…c…d…b c…a ... [38]

Удалено: a…b…c…a…d…c a…b ... [39]

Удалено: a…b…c…a…d…c a…b ... [40]

Удалено: d…b…c…a…d…c a…b ... [41]

Удалено: d…b…b…c… ... [42]

Удалено: b…b…d…b… ... [43]

5.8.5. Функциональная полнота

1. Постройте функции шефферовского типа, существенно зависящие от трёх переменных.

2. Являются ли функции, приведённые ниже, функциями шефферовского типа? Если нет, то какую функцию для каждой из них необходимо добавить, чтобы она образовала функционально полный базис?

1: *f*=<01101001>;
2: *f*=<00010011>;
3: *f*=<00101011>;
4: *f*=<10010110>;
5: *f*=<11001010>;
6: *f*=<01110100>;
7: *f*=><00010111>;
8: *f*=<10011100>;
9: *f*=<11010000>

Удалено: *Задача*
Удалено: ¶ *Задача*
Удалено: функция
Удалено: выше
Удалено: функцией
Удалено: к
Удалено: ней
Удалено: и
Удалено: и
Удалено: Вариант
Удалено: Вариант
Удалено: Вариант
Удалено: Вариант
Удалено: Вариант
Удалено: Вариант
Удалено: Вариант
Удалено: Вариант
Удалено: Вариант

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Акимов, Олег Евгеньевич. Дискретная математика: логика, группы, графы / О.Е. Акимов. - М.: Лаборатория Базовых Знаний, 2001. - 352 с.
2. Горбатов, Вячеслав Афанасьевич. Дискретная математика: Учебник для студентов втузов / В. А. Горбатов, А. В. Горбатов, М. В. Горбатова. - М.: АСТ : Астрель, 2003. - 447 с.
3. Горбатов, Вячеслав Афанасьевич. Фундаментальные основы дискретной математики. Информационная математика: Учебник для втузов / В.А. Горбатов. - М.: Наука. Физматлит, 2000. - 544 с.
4. Горбатов, Вячеслав Афанасьевич. Основы дискретной математики: Учеб. пособие для вузов. - М.: Высш.шк., 1986. - 311с.
5. Карпов, Юрий Глебович. Теория автоматов: Учебник для студентов вузов / Ю.Г. Карпов. - М.; СПб.; Н. Новгород и др.: Питер, 2002. - 224 с.
6. Кострикин, Алексей Иванович. Введение в алгебру: Учебник для студентов ун-тов, обучающихся по специальности "Математика" и "Прикладная математика". Ч. 3: Основные структуры алгебры / А.И. Кострикин. - М.: ФИЗМАТЛИТ, 2000. - 272 с.
7. Курош, Александр Геннадьевич. Курс высшей алгебры: Для ун-тов. - 11-е изд., стер. - М.: Наука, 1975. - 431с.
8. Новиков Ф. А. Дискретная математика для программистов: Для студентов вузов / Ф.А. Новиков. - СПб.; М.; Харьков; Минск: Питер, 2001. - 304 с.
9. Романовский, Иосиф Владимирович. Дискретный анализ: Учеб. пособие по прикладной математике и информатике / И.В. Романовский. - 2-е изд., испр. - СПб.; М.: ФИЗМАТЛИТ : Невский диалект : Лаборатория Базовых Знаний, 2001. - 240 с.
10. Яблонский, Сергей Всеволодович. Введение в дискретную математику: Учеб. пособие для студентов вузов, обучающихся по спец. "Прикладная математика" / С. В. Яблонский. - 3-е изд., стер. - М.: Высшая математика, 2001. - 384 с.

Стр. 53: [1] Удалено Mary 05.12.2003 12:00:00

Стр. 55: [2] Удалено Mary 05.12.2003 11:52:00

Табл. 3.11

Табл. 3.12

	1	2	3	4	5	6	s
1		1		1			2=*1·2*=2·1
2				1		1	2=1·2=*2·1*
3							0=x·0=0·x
4		1	1	1		1	4=*2·2*=1·4=4·1
5				1			1=*1·1*
6							0=x·0=0·x
t	0=0·x =x·0	2=1·2 =*2·1*	1=*1·1*	4=*2·2* 1·4=4·1	0=x·0 =0·x	2=*1·2* =2·1	

Стр. 65: [3] Удалено Mary 17.12.2003 11:07:00

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
1	0	0	0	1	1	1	1	0	0
2	0	0	0	0	1	0	0	0	0
3	0	0	0	0	1	0	1	0	0
4	0	0	0	0	1	0	1	1	0
5	0	1	0	0	0	0	0	0	1
6	0	0	0	1	1	0	0	0	0
7	0	1	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	1
9	0	0	0	0	1	0	0	0	0

Стр. 77: [4] Удалено **Mary** **17.12.2003 13:12:00**

Пример

Стр. 77: [4] Удалено **Mary** **17.12.2003 11:29:00**

Рассмотрим

Стр. 77: [4] Удалено **Mary** **17.12.2003 11:29:00**

ю

Стр. 77: [4] Удалено **Mary** **17.12.2003 11:29:00**

Стр. 77: [4] Удалено **Mary** **17.12.2003 11:29:00**

Стр. 77: [4] Удалено **Mary** **17.12.2003 11:30:00**

,

Стр. 77: [4] Удалено **Mary** **17.12.2003 11:29:00**

найдём её разложение

Стр. 77: [4] Удалено **Mary** **05.12.2003 13:28:00**

Стр. 77: [4] Удалено **Mary** **17.12.2003 11:26:00**

.

Стр. 77: [4] Удалено **Mary** **17.12.2003 11:27:00**

)

Стр. 77: [4] Удалено **Mary** **17.12.2003 11:27:00**

Стр. 77: [4] Удалено	**Mary**	**17.12.2003 11:27:00**

Стр. 77: [4] Удалено	**Mary**	**05.12.2003 13:28:00**

Стр. 77: [4] Удалено	**Mary**	**17.12.2003 11:27:00**

Стр. 77: [4] Удалено	**Mary**	**17.12.2003 11:30:00**

Стр. 77: [5] Удалено	**Mary**	**17.12.2003 13:12:00**

Пример

Стр. 77: [5] Удалено	**Mary**	**05.12.2003 13:33:00**

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено	**Mary**	**17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [6] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [7] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [7] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [7] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:41:00**

f

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

X

Стр. 77: [8] Удалено **Mary** **17.12.2003 11:35:00**

x

Стр. 77: [8] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [8] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [8] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [8] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [8] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [8] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [8] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [8] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [9] Удалено	Mary	17.12.2003 12:12:00

a

Стр. 77: [9] Удалено	Mary	17.12.2003 12:13:00

b

Стр. 77: [9] Удалено	Mary	17.12.2003 12:12:00

a

Стр. 77: [9] Удалено	Mary	17.12.2003 12:13:00

b

Стр. 77: [9] Удалено	Mary	17.12.2003 12:12:00

a

Стр. 77: [9] Удалено	Mary	17.12.2003 12:12:00

a

Стр. 77: [9] Удалено	Mary	17.12.2003 12:14:00

b

Стр. 77: [9] Удалено	Mary	17.12.2003 12:12:00

a

Стр. 77: [9] Удалено	Mary	17.12.2003 12:12:00

a

Стр. 77: [9] Удалено	Mary	17.12.2003 12:12:00

a

Стр. 77: [9] Удалено	Mary	17.12.2003 12:14:00

b

Стр. 77: [9] Удалено	Mary	17.12.2003 12:12:00

a

Стр. 77: [9] Удалено	Mary	17.12.2003 12:14:00

b

Стр. 77: [9] Удалено	Mary	17.12.2003 12:12:00

a

Стр. 77: [10] Удалено	Mary	17.12.2003 11:41:00

f

Стр. 77: [10] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:35:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Стр. 77: [10] Удалено	Mary	17.12.2003 11:36:00

x

Табл. 5.6

x1x2/x3	*0*	*1*
00		
01		15
11	12	11
10	14	13

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:41:00**

f

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 77: [11] Удалено **Mary** **17.12.2003 11:36:00**

x

Стр. 78: [12] Удалено **Mary** **17.12.2003 11:44:00**

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:41:00**

f

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:37:00**

x

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:37:00**

x

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:37:00**

x

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:37:00**

x

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:37:00**

x

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:37:00**

x

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:37:00**

x

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:37:00**

x

Стр. 79: [13] Удалено **Mary** **17.12.2003 11:37:00**

x

Стр. 79: [13] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [13] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [13] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [13] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [13] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [13] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:41:00**

f

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено	**Mary**	**17.12.2003 11:37:00**

X

Стр. 79: [14] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [14] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [14] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [14] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [14] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [14] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [15] Удалено Mary 17.12.2003 11:45:00

Стр. 79: [15] Удалено Mary 17.12.2003 11:45:00

Табл. 5.8

Стр. 79: [16] Удалено Mary 17.12.2003 11:44:00

x1x2/x3	*0*	*1*
00		1
01	1	1
11	1	
10	1	1

Стр. 79: [17] Удалено Mary 17.12.2003 11:41:00

f

Стр. 79: [17] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [17] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [17] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [17] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [17] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [17] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [17] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [17] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [17] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [18] Удалено Mary 17.12.2003 11:41:00

f

Стр. 79: [18] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [18] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [18] Удалено Mary 17.12.2003 11:37:00

x

Стр. 79: [18] Удалено	Mary	17.12.2003 11:37:00

x

Стр. 79: [18] Удалено	Mary	17.12.2003 11:37:00

x

Стр. 79: [18] Удалено	Mary	17.12.2003 11:37:00

x

Стр. 79: [18] Удалено	Mary	17.12.2003 11:37:00

x

Стр. 79: [18] Удалено	Mary	17.12.2003 11:37:00

x

Стр. 79: [18] Удалено	Mary	17.12.2003 11:37:00

x

Стр. 79: [19] Удалено	Mary	17.12.2003 11:41:00

f

Стр. 79: [19] Удалено	Mary	17.12.2003 11:38:00

x

Стр. 79: [19] Удалено	Mary	17.12.2003 11:38:00

x

Стр. 79: [19] Удалено	Mary	17.12.2003 11:38:00

x

Стр. 91: [20] Удалено	Mary	17.12.2003 12:09:00

Стр. 91: [20] Удалено	Mary	05.12.2003 12:18:00

Задачи

Стр. 91: [21] Удалено	Mary	17.12.2003 12:08:00

Вариант

Стр. 91: [21] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [21] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [21] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [21] Удалено	Mary	17.12.2003 12:14:00

d

Стр. 91: [22] Удалено	Mary	17.12.2003 12:08:00

Вариант

Стр. 91: [22] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [22] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [22] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [22] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [22] Удалено	Mary	17.12.2003 12:15:00

d

Стр. 91: [22] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [22] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [22] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [23] Удалено	Mary	17.12.2003 12:08:00

Вариант

Стр. 91: [23] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [23] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [23] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [23] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [23] Удалено	Mary	17.12.2003 12:15:00

d

Стр. 91: [23] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [23] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [23] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [24] Удалено	Mary	17.12.2003 12:08:00

Вариант

Стр. 91: [24] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [24] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [24] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [24] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [24] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [24] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [24] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [25] Удалено	Mary	17.12.2003 12:09:00

Вариант

Стр. 91: [25] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [25] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [25] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [25] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [25] Удалено	Mary	17.12.2003 12:15:00

d

Стр. 91: [25] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [25] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [25] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [26] Удалено	Mary	17.12.2003 12:09:00

Вариант

Стр. 91: [26] Удалено	Mary	17.12.2003 12:15:00

d

Стр. 91: [26] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [26] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [26] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [26] Удалено	Mary	17.12.2003 12:15:00

d

Стр. 91: [26] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [26] Удалено	Mary	17.12.2003 12:10:00

a

Стр. 91: [26] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [27] Удалено	Mary	17.12.2003 12:09:00

Вариант

Стр. 91: [27] Удалено	Mary	17.12.2003 12:15:00

d

Стр. 91: [27] Удалено	Mary	17.12.2003 12:12:00

b

Стр. 91: [27] Удалено	Mary	17.12.2003 12:13:00

b

Стр. 91: [27] Удалено	Mary	17.12.2003 12:14:00

c

Стр. 91: [27] Удалено	Mary	17.12.2003 12:15:00

d

Стр. 91: [28] Удалено	Mary	17.12.2003 12:09:00

Вариант

Стр. 91: [28] Удалено	Mary	17.12.2003 12:13:00

b

Стр. 91: [28] Удалено	**Mary**	**17.12.2003 12:13:00**

b

Стр. 91: [28] Удалено	**Mary**	**17.12.2003 12:15:00**

d

Стр. 91: [28] Удалено	**Mary**	**17.12.2003 12:13:00**

b

Стр. 91: [28] Удалено	**Mary**	**17.12.2003 12:15:00**

d

Стр. 92: [29] Удалено	**Mary**	**17.12.2003 12:16:00**

Вариант

Стр. 92: [29] Удалено	**Mary**	**17.12.2003 12:16:00**

Стр. 92: [29] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [29] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [29] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [29] Удалено	**Mary**	**17.12.2003 12:18:00**

d

Стр. 92: [30] Удалено	**Mary**	**17.12.2003 12:16:00**

Вариант

Стр. 92: [30] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [30] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [30] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [30] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [30] Удалено	**Mary**	**17.12.2003 12:18:00**

d

Стр. 92: [30] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [30] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [30] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [31] Удалено	Mary	17.12.2003 12:16:00

Вариант

Стр. 92: [31] Удалено	Mary	17.12.2003 12:16:00

Стр. 92: [31] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [31] Удалено	Mary	17.12.2003 12:20:00

b

Стр. 92: [31] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [31] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [31] Удалено	Mary	17.12.2003 12:18:00

d

Стр. 92: [31] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [31] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [31] Удалено	Mary	17.12.2003 12:20:00

b

Стр. 92: [32] Удалено	Mary	17.12.2003 12:16:00

Вариант

Стр. 92: [32] Удалено	Mary	17.12.2003 12:16:00

Стр. 92: [32] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [32] Удалено	Mary	17.12.2003 12:20:00

b

Стр. 92: [32] Удалено	Mary	17.12.2003 12:20:00

b

Стр. 92: [32] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [32] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [32] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [32] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:16:00**

Вариант

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:16:00**

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:18:00**

d

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [33] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [34] Удалено	**Mary**	**17.12.2003 12:16:00**

Вариант

Стр. 92: [34] Удалено	**Mary**	**17.12.2003 12:19:00**

d

Стр. 92: [34] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [34] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [34] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [34] Удалено	**Mary**	**17.12.2003 12:19:00**

d

Стр. 92: [34] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [34] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [34] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [35] Удалено	**Mary**	**17.12.2003 12:16:00**

Вариант

Стр. 92: [35] Удалено	**Mary**	**17.12.2003 12:17:00**

Стр. 92: [35] Удалено	**Mary**	**17.12.2003 12:19:00**

d

Стр. 92: [35] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [35] Удалено	**Mary**	**17.12.2003 12:20:00**

b

Стр. 92: [35] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [35] Удалено	**Mary**	**17.12.2003 12:19:00**

d

Стр. 92: [36] Удалено	**Mary**	**17.12.2003 12:16:00**

Вариант

Стр. 92: [36] Удалено	**Mary**	**17.12.2003 12:17:00**

Стр. 92: [36] Удалено	**Mary**	**17.12.2003 12:21:00**

b

Стр. 92: [36] Удалено	**Mary**	**17.12.2003 12:21:00**

b

Стр. 92: [36] Удалено	**Mary**	**17.12.2003 12:19:00**

d

Стр. 92: [36] Удалено	**Mary**	**17.12.2003 12:21:00**

b

Стр. 92: [36] Удалено	**Mary**	**17.12.2003 12:19:00**

d

Стр. 92: [37] Удалено	**Mary**	**17.12.2003 12:21:00**

b

Стр. 92: [37] Удалено	**Mary**	**17.12.2003 12:21:00**

b

Стр. 92: [37] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [37] Удалено	**Mary**	**17.12.2003 12:19:00**

d

Стр. 92: [38] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [38] Удалено	**Mary**	**17.12.2003 12:21:00**

b

Стр. 92: [38] Удалено	**Mary**	**17.12.2003 12:21:00**

b

Стр. 92: [38] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [38] Удалено	**Mary**	**17.12.2003 12:19:00**

d

Стр. 92: [38] Удалено	**Mary**	**17.12.2003 12:21:00**

b

Стр. 92: [38] Удалено	**Mary**	**17.12.2003 12:21:00**

c

Стр. 92: [38] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [39] Удалено	**Mary**	**17.12.2003 12:19:00**

a

Стр. 92: [39] Удалено	**Mary**	**17.12.2003 12:21:00**

b

Стр. 92: [39] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [39] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [39] Удалено	Mary	17.12.2003 12:19:00

d

Стр. 92: [39] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [39] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [39] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [40] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [40] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [40] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [40] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [40] Удалено	Mary	17.12.2003 12:19:00

d

Стр. 92: [40] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [40] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [40] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [41] Удалено	Mary	17.12.2003 12:19:00

d

Стр. 92: [41] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [41] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [41] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [41] Удалено	Mary	17.12.2003 12:19:00

d

Стр. 92: [41] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [41] Удалено	Mary	17.12.2003 12:19:00

a

Стр. 92: [41] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [42] Удалено	Mary	17.12.2003 12:19:00

d

Стр. 92: [42] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [42] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [42] Удалено	Mary	17.12.2003 12:21:00

c

Стр. 92: [42] Удалено	Mary	17.12.2003 12:19:00

d

Стр. 92: [43] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [43] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [43] Удалено	Mary	17.12.2003 12:19:00

d

Стр. 92: [43] Удалено	Mary	17.12.2003 12:21:00

b

Стр. 92: [43] Удалено	Mary	17.12.2003 12:19:00

d