Оглавление

Часть 1

Множества и бинарные отношения

Глава 1

Множества

1.1 Определения и примеры

Множество

Множество *– это набор (совокупность, коллекция)* ***различимых*** *объектов, обладающих общим для них всех характеристическим свойством. Объекты, составляющие множество, называются его* ***элементами****.*

Термины *множество* и *элемент* не определяются, так как нет более фундаментальных и элементарных понятий с помощью которых можно было бы дать такое определение. Следовательно и отношение "*быть элементом множества*" не определяется. Примерами других неопределяемых терминов могут служить: число, точка, линия (множество точек) в математике; пространство, время, масса в физике.

Для того чтобы показать, что совокупность некоторых объектов образует множество, обозначения этих объектов заключаются в фигурные скобки $\{\ldots\}$.

Примеры 1.1.

Примеры конечных множеств:

1. Множество цифр шестнадцатеричной системы счисления: $\{0, 1, \ldots, 9, A, B, \ldots, F\}$.
2. Множество цифр двоичной системы счисления: $\{0, 1\}$.
3. Множество букв латинского алфавита:

   $\{a, b, \ldots, z, A, B, \ldots, Z\}$.

Примеры бесконечных множеств:

4. Множество натуральных чисел: $\{1, 2, 3, \ldots\}$.
5. Множество четных натуральных числа: $\{2, 4, 6, \ldots\}$.
6. Множество целых чисел: $\{0, -1, 1, -2, 2, \ldots\}$. ◄

Элементы множества обычно обозначаются латинскими строчными буквами, а сами множества – прописными. Для указания принадлежности элемента множеству используется знак $\in$ (обратный к нему $\notin$ или $\overline{\in}$ – не принадлежит): $a \in A$, $a \overline{\in} B$, $a \notin C$.

Определение. *Два множества A и B* ***равны*** *(обозначается $A = B$) тогда и только тогда, когда они состоят из одних и тех же элементов.*

Замечание. Выражение “тогда и только тогда, когда"эквивалентно выражению “необходимо и достаточно" и равнозначно логической эквивалентности. Это определение следует понимать следующим образом:

$\Rightarrow$ (*необходимость*): если множества равны, то они состоят из одних и тех же элементов;

$\Leftarrow$ (*достаточность*): если множества состоят из одних и тех же элементов, то они равны. ►

Пример 1.2.

$A = \{1, 2, 3\}; B = \{2, 3, 1\}; D = \{1, 2, 4\};$

$A = B$ – элементы множеств могут быть записаны в любом порядке;

$A \neq D$ – множества содержат одинаковое число элементов, но в множестве A нет элемента 4, а в множестве D нет элемента 3. ◄

Определение. *Множество, не содержащее ни одного элемента, называется* ***пустым*** *множеством и обозначается $\varnothing$ ($\{\}$).*

Примеры 1.3.

1. $\{0\} \neq \varnothing$: множество $\{0\}$ содержит один элемент – число 0;
2. $\varnothing \neq 0$: $\varnothing$ – множество, а 0 не является множеством;
3. $\{\varnothing\} \neq \varnothing$: множество $\{\varnothing\}$ содержит один элемент, а именно пустое множество $\varnothing$. ◄

Определение. *Множество A* ***содержится*** *во множестве B (обозначение $A \subseteq B$) тогда и только тогда, когда каждый элемент*

A является элементом B.

Пример 1.4. Пусть $A = \{1, 2, 3\}$, $\mathbb{N} = \{1, 2, 3...\}$, тогда $A \subseteq \mathbb{N}$. ◀

Иногда вместо "A содержится в B" говорят: *A включается* в B, или B содержит A, или B включает A. Если A содержится в B, то говорят, что A – ***подмножество*** B или, что то же самое, B – ***надмножество*** A. Если $A \subseteq B$ и существует такой $x \in B$, что $x \notin A$, то говорят, что A является ***собственным подмножеством*** B (обозначают $A \subset B$), или, что то же самое, B – ***собственное надмножество*** A. Если A не является подмножеством B, то это обозначают так: $A \not\subseteq B$.

Примеры 1.5.

1. Пусть $A = \{a, b, c, d\}; B = \{b, c, a\}$, тогда $B \subseteq A$ и $B \subset A$.
2. Пусть $A = \{1, 3, 4\}, B = \{3, 1, 4\}$, тогда $A \subseteq B$ и $B \subseteq A$. ◀

З а м е ч а н и е. Предполагается, что для любого множества A $\varnothing \subseteq A$. ▶

В дальнейшем нам часто придется доказывать равенство двух множеств. Для этого можно использовать непосредственно определение 1, однако в ряде случаев факт равенства двух множеств проще доказать основываясь на следующей теореме.

Теорема 1.1 *Множества A и B равны тогда и только тогда, когда $A \subseteq B$ и $B \subseteq A$.*

Д о к а з а т е л ь с т в о (от противного).

1. $\Rightarrow$ (*необходимость*). Пусть $A = B$. Предположим, что $A \not\subseteq B$. Следовательно существует такой x, что $x \in A$, $x \notin B$, однако это противоречит тому, что множества A и B равны. Аналогично показывается, что предположение $B \not\subseteq A$ также ведет к противоречию. Следовательно, если $A = B$, то $A \subseteq B, B \subseteq A$.

2. $\Leftarrow$ (*достаточность*). Пусть $A \subseteq B$ и $B \subseteq A$. Предположим, что $A \neq B$. Из определения 1 равенства множеств следует, что при этом либо A содержит элемент, не принадлежащий B, либо B содержит элемент, не принадлежащий A. Рассмотрим первый случай, то есть предположим, что существует такой x, что $x \in A$, $x \notin B$, однако это противоречит усло-

вию $A \subseteq B$. Второй случай также приводит к противоречию. Следовательно, если $A \subseteq B$ и $B \subseteq A$, то $A = B$. ☒

Универсальное множество

При определении понятий *содержится*, *подмножество*, *равенство* множеств неявно предполагалась возможность проверить для любого элемента, входит ли он в данное множество. Вообще говоря, определение конкретных множеств строится так, что в самом определении ограничивается класс возможных объектов. Это соглашение удобно ввести явным образом и считать, что заранее фиксирован класс исходных объектов. Говоря далее одновременно о нескольких множествах, мы имеем в виду, что в них входят только объекты, принадлежащие этому классу – *универсальному множеству* (*универсуму*).

Если все множества, рассматриваемые в рамках определенной ситуации или определенного рассуждения, являются подмножествами некоторого множества U, то это множество называют ***универсальным*** множеством (***универсумом***) для данного рассуждения.

Примеры 1.6.

1. В элементарной теории чисел U – множество целых чисел или множество рациональных чисел.

2. Если мы рассматриваем множества, состоящие из строчных букв латинского алфавита, то $U = \{a, b, c, ..., z\}$. ◄

Понятие универсума очень расплывчато, однако более точного определения при том уровне строгости, на котором ведется изложение, дать невозможно. Каждый раз, когда возникает необходимость определить универсум, следует проанализировать с какого сорта объектами мы будем работать и определить множество, включающее все такие объекты. При этом необходимо, чтобы полученный универсум был наименьшим множеством, содержащим данные объекты, другими словами никакое его собственное подмножество не должно быть универсумом.

Более или менее конструктивный прием при задании универсума заключается в выборе наименьшего множества, которое содержит все обсуждаемые множества как свои подмножества и не является элементом самого себя.

Представление множеств

Множество можно задать (представить) одним из следующих способов.

• Перечислением всех его элементов: $A = \{a, b, c, \ldots, z\}$ (для бесконечных множеств: $\mathbb{N} = \{1, 2, 3, \ldots\}$). Такой способ задания множеств соответствует представлению в языках программирования списками или массивами.

• С помощью характеристического предиката:

$A = \{x|P(x)\}$, где A – множество, x – элемент универсального множества U, $P(x)$ – предикат, определенный на элементах U. $\{x|P(x)\}$ читается как "множество всех x, таких, что $P(x)$ истинно".

• С помощью характеристической функции (***функции принадлежности***). Пусть задано универсальное множество U. Тогда множество A на универсуме U определяется с помощью функции принадлежности $\mu_A(x)$:

для любого элемента $x \in U$ $\mu_A(x) = \begin{cases} 1, & \text{если } x \in A; \\ 0, & \text{если } x \notin A. \end{cases}$

Таким образом, на данном универсуме U множество A задается совокупностью пар $A = \{\langle x, \mu_A(x)\rangle |\ x \in U\}$.

Примеры 1.7.

1. Одно и то же множество представлено перечислением и различными высказываниями:

$B = \{0, 1\}; B = \{b|b$ – число двоичной системы счисления$\}$; $B = \{b|b = 0$ или $b = 1\}$; $B = \{b|b^2 - b = 0\}$.

2. $E = \{x|x < 1$ и $x > 2\}$; если $U = \mathbb{N}$, то $E = \varnothing$.

3. Пусть $U = \{a, b, c\}$. Зададим множества A_i с помощью функций принадлежности:

$$\mu_{Ai}(x) = \begin{array}{|c|c|c||l|} \hline a & b & c & \quad A_i \\ \hline 0 & 0 & 0 & A_0 = \varnothing \\ 0 & 0 & 1 & A_1 = \{c\} \\ 0 & 1 & 0 & A_2 = \{b\} \\ 1 & 0 & 0 & A_3 = \{a\} \\ 0 & 1 & 1 & A_4 = \{b, c\} \\ 1 & 0 & 1 & A_5 = \{a, c\} \\ 1 & 1 & 0 & A_6 = \{a, b\} \\ 1 & 1 & 1 & A_7 = \{a, b, c\} \\ \hline \end{array}$$

$A_0 = \{\langle a, 0\rangle, \langle b, 0\rangle, \langle c, 0\rangle\}$, $A_1 = \{\langle a, 0\rangle, \langle b, 0\rangle, \langle c, 1\rangle\}$,
$A_2 = \{\langle a, 0\rangle, \langle b, 1\rangle\ \langle c, 0\rangle\}, \cdots$, $A_7 = \{\langle a, 1\rangle, \langle b, 1\rangle, \langle c, 1\rangle\}$. ◄

Функция принадлежности может быть как дискретной, так и непрерывной.

Пример 1.8.
$U = \{x | 0 \leqslant x \leqslant 100\}$ или $U = [0, 100]$;
$A = \{x | x \in U \ \wedge \ x \geqslant 50\}$;

$$\mu_A(x) = \begin{cases} 1, & \text{если } x \geqslant 50; \\ 0, & \text{если } x < 50. \end{cases}$$

На рисунке 1.1. приводится график этой функции принадлежности:

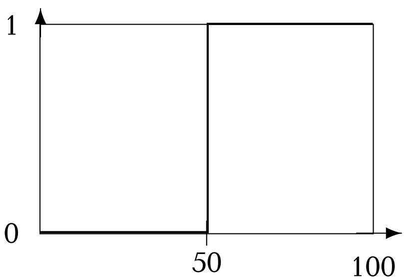


Рис. 1.1. Функция принадлежности

З а м е ч а н и е. Интересное расширение понятия множества было введено в 1965 г. Л.Заде – ***нечеткое*** множество. Нечеткое множество задается с помощью функции принадлежности, которая может принимать значения в интервале от 0 до 1: $A = \{\langle x, \mu_A(x)\rangle\}, x \in U$, где $\mu_A(x) \in [0,1]$ – значение из интервала $[0,1]$.

Пример 1.9.
Пусть снова $U = [0, 100]$, а

$$\mu_A(x) = \begin{cases} 0, & 0 \leqslant x \leqslant 50 \\ \dfrac{k(x-50)^2}{1+k(x-50)^2}, & 50 < x \leqslant 100 \end{cases}$$

Приблизительный график этой функции принадлежности приведен на рисунке 1.2. ◀

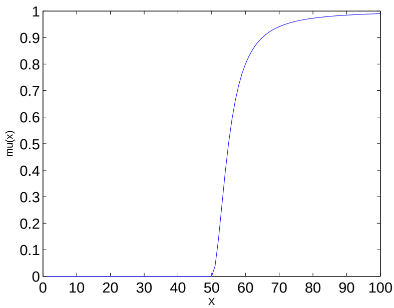


Рис. 1.2. Функция принадлежности нечеткого множества

С помощью задания нечетких множеств можно формализовать такие размытые (нечеткие) понятия как *большой*, *малый*, *старый*, *молодой* и отношений типа *немного больше*, *примерно равны* и т.д. ▶

Диаграммы Эйлера–Венна

Для иллюстрации отношений между множествами используется графическое представление, известное как *диаграмма Эйлера–Венна* или *круги Эйлера*. Некоторая замкнутая фигура (обычно прямоугольник) представляет универсальное множество, внутри которого изображаются замкнутые кривые, представляющие подмножества универсума.

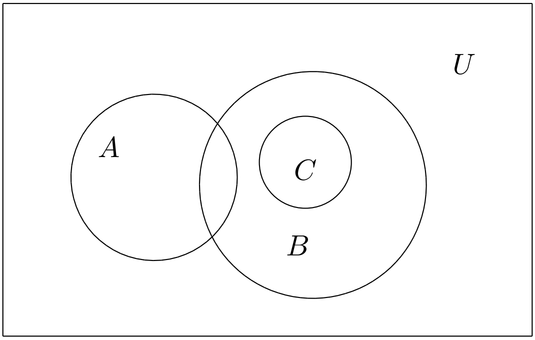


Рис. 1.3. Диаграмма Эйлера–Венна

На рисунке 1.3. изображены универсальное множество U в виде прямоугольника и множества $A, B, C \subseteq U$ в виде овалов, при этом $C \subseteq B$.

Мощность множества

Мощность множества – это обобщение понятия "число элементов" на произвольные множества. Это обобщение опирается на понятие ***взаимно однозначного соответствия***: такого соответствия между двумя множествами A и B, при котором каждому элементу из множества A сопоставляется единственный элемент из множества B, и каждому элементу из B сопоставляется ровно один элемент из A.

Установить взаимно однозначное соответствие между двумя конечными множествами можно тогда и только тогда, когда оба множества состоят из одинакового числа элементов. Как обобщение этого факта определяют количественную эквивалентность или ***равномощность*** бесконечных множеств.

Определение. *Два множества* ***равномощны*** *тогда и только тогда, когда между ними можно установить взаимно однозначное соответствие.*

Мощность множества A обозначается через $|A|$. Множество, равномощное множеству натуральных чисел $\mathbb{N} = \{1, 2, 3, \ldots\}$, называется ***счётным*** множеством. Мощность счетного множества имеет особое обозначение: $|\mathbb{N}| = \aleph_0$ (читается "алеф-нуль"). $\aleph_0$ – наименьшая мощность, которую может иметь бесконечное множество.

Множество четных натуральных чисел $\{2, 4, 6, \ldots\}$, множество целых чисел $\{\ldots, -2, -1, 0, 1, 2, \ldots\}$ – счетные множества.

Мощность множества действительных чисел называется ***мощностью континуума*** и обозначается $\mathfrak{c}$ или $|2^{\aleph_0}|$.

Мощность континуума имеют:

– множество всех действительных чисел;

– множество всех подмножеств счетного множества;

– множество всех комплексных чисел и множества точек 2-, 3-, n-мерного пространства.

Мощность булеана множества

Множество, элементами которого являются множества, называется ***семейством*** *множеств.* Семейства множеств обычно обозначают прописными рукописными буквами латинского алфавита.

Примеры 1.10.
1. $\mathcal{A} = \{\{1, 2, 3\}, \{2, 3, 4\}, \{1, 7\}\}$.

2. $\mathcal{B} = \{C | C$ – множество букв, входящих в некоторое слово русского языка$\}$ = {{о,к,н,о},{т,о,к},{п,л,о,т},...} ◄

Определение. *Семейство всех подмножеств данного множества A называется* ***булеаном*** *A.*

Обозначения *булеана*: $\mathcal{B}(A)$ или 2^A.

Пример 1.11.
Пусть $A = \{a, b, c\}$, тогда
$\mathcal{B}(A) = \{\varnothing, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c, \}\}$. ◄

Теорема 1.2 *Пусть A – конечное множество и $|A| = n$. Тогда*

$$|\mathcal{B}(A)| = 2^n.$$

Д о к а з а т е л ь с т в о . Доказательство проведем индукцией по числу элементов в A.

Базис индукции. $n = 0$, то есть $A = \varnothing$. Так как A не содержит элементов, то $\varnothing$ является его единственным подмножеством. Следовательно $\mathcal{B}(A) = \{\varnothing\}$. $|\mathcal{B}(A)| = 1 = 2^0$.

Предположение индукции. Предположим, что теорема справедлива для множества $A' = \{x_1, \ldots, x_{n-1}\}$, $|A'| = n - 1$: $|\mathcal{B}(A')| = 2^{n-1}$.

Шаг индукции. Рассмотрим множество $A = \{x_1, \ldots, x_n\}$, полученное из множества A' добавлением x_n. Все подмножества A' являются одновременно подмножествами A. Кроме того в A входят все подмножества, полученные из подмножеств A' добавлением в них элемента x_n. Следовательно, A содержит вдвое больше подмножеств чем A':

$$|\mathcal{B}(A)| = 2 \cdot |\mathcal{B}(A')|.$$

Заключение индукции. Таким образом,

$$|\mathcal{B}(A)| = 2 \cdot 2^{n-1} = 2^n. \quad \boxtimes$$

Логическое отступление

В этом разделе приводится минимум сведений из математической логики, необходимый для дальнейших рассуждений.Понятие ***высказывание*** является фундаментальным в логике и, подобно понятиям множества и его элементов, не определяется через другие. Высказывание – это утвердительное (повествовательное) предложение, которое может быть либо истинным, либо ложным.

Истина или ***ложь*** (общепринятые обозначения 1 и 0 соответственно), приписанная некоторому высказыванию, называется *истинностным* или *логическим* значением высказывания.

Символы $p, q, r, \ldots$, которые используются для обозначения высказываний, называются *атомарными формулами* или ***атомами***. Из атомов строятся составные высказывания – ***формулы*** с помощью логических *операций* (связок):

$\neg$ – отрицание (не) ,

$\wedge$ – конъюнкция (и) ,

$\vee$ – дизъюнкция (или) ,

$\to$ – импликация (если то),

$\leftrightarrow$ – эквиваленция (эквивалентно) .

Истинностные значения формул определяются с помощью ***таблиц истинности*** логических операций:

p	$\neg p$
0	1
1	0

p	q	$p \wedge q$	$p \vee q$	$p \to q$	$p \leftrightarrow q$
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	0
1	1	1	1	1	1

Истинностные значения формулы вычисляются по истинностным значениям атомов, входящих в эту формулу, и таблицам истинности логических операций. Порядок выполнения операций в формуле определяется с помощью скобок: операции в скобках имеют бо́льший приоритет. Кроме того, сами операции упорядочиваются по приоритетам (в порядке убывания): $\neg, \wedge, \vee, \to, \leftrightarrow$.Поэтому некоторые скобки в формулах можно опускать, например: $(p \vee q)$ можно заменить на $p \vee q$, $(p \to (q \wedge r))$ – на $p \to q \wedge r$, $p \to (q \wedge (\neg r))$ – на $p \to q \wedge \neg r$.

Интерпретация формулы F – это назначение истинностных значений атомам, содержащимся в формуле.

Пример 1.12. Определим истинностное значение формулы

$p \wedge (q \vee r)$

в такой интерпретации: $p = 1, q = 0, r = 1$.

$1 \wedge (0 \vee 1) = 1.$

В другой интерпретации:$p = 0, q = 0, r = 1$ получим

$0 \wedge (0 \vee 1) = 0.$ ◀

Формула F ***общезначима*** тогда и только тогда, когда она истинна во всех возможных интерпретациях. Для краткости будем иногда обозначать общезначимую формулу так: $F = 1$.

Формула F ***противоречива*** (*невыполнима*) тогда и только тогда, когда она ложна во всех возможных интерпретациях. Для краткости будем иногда обозначать противоречивую формулу так: $F = 0$.

Формулу, которая не является ни общезначимой ни противоречивой называют ***нейтральной***.

Очевидно, что отрицание общезначимой формулы противоречиво, а отрицание противоречивой формулы общезначимо.

Пример 1.13. Формула $(p \to q) \wedge p \to q$ общезначима,
формула $(p \to q) \wedge p \wedge \neg q$ – противоречива,
формула $(p \to q) \wedge p$ – нейтральна.
Рассмотрим все возможные интерпретации этих формул:

p	q	$(p \to q) \wedge p \to q$	$(p \to q) \wedge p \wedge \neg q$	$(p \to q) \wedge p$
0	0	1	0	0
0	1	1	0	0
1	0	1	0	0
1	1	1	0	1

Две формулы F и G называются ***эквивалентными*** (или F эквивалентна G; обозначается $F = G$) тогда и только тогда, когда истинностные значения F и G совпадают в каждой интерпретации.

Пример 1.14.
Проверим эквивалентность формул $p \to q$ и $\neg p \vee q$:

p	q	$p \to q$	$\neg p \vee q$
0	0	1	1
0	1	1	1
1	0	0	0
1	1	1	1

Истинностные значения формул совпадают во всех интерпретациях, следовательно, $p \to q = \neg p \vee q$. ◄

Пусть F, G и H – формулы. С помощью таблиц истинности можно проверить эквивалентность формул:

1.$a) F \vee G = G \vee F$; $b) F \wedge G = G \wedge F$;

2. $a) (F \vee G) \vee H = F \vee (G \vee H)$;

$b)$ $(F \wedge G) \wedge H = F \wedge (G \wedge H)$;

3. $a)$ $F \vee (G \wedge H) = (F \vee G) \wedge (F \vee H)$;

$b) F \wedge (G \vee H) = (F \wedge G) \vee (F \wedge H)$;

4. $a) F \vee 0 = F$; $b) F \wedge 1 = F$;

5. $a) F \vee \neg F = 1$; $b) F \wedge \neg F = 0$.

В логике эти пары эквивалентных формул часто называют законами:

1 – коммутативные законы для дизъюнкции и конъюнкции;

2 – ассоциативные законы для дизъюнкции и конъюнкции;

3 – дистрибутивные законы: дистрибутивность дизъюнкции относительно конъюнкции и дистрибутивность конъюнкции относительно дизъюнкции.

5 $a)$ – закон исключенного третьего.

5 $b)$ – закон противоречия.

Пример 1.15. Доказать эквивалентность формул:

$(p \vee q) \wedge (p \vee r) \wedge (p \vee s) = p \vee (q \wedge r \wedge s)$.

Применяя к левой части последовательно дистрибутивный закон 3$a)$, а затем ассоциативный закон 2 $b)$, получим

$(p \vee (q \wedge r)) \wedge (p \vee s) = (p \vee (q \wedge r) \wedge s) = p \vee (q \wedge r \wedge s)$. ◀

При исследовании свойств множеств и операций над ними переходят к высказываниям о принадлежности элементов тем или другим множествам. Например, $A \subseteq B$ означает, что из принадлежности любого элемента x множеству A следует принадлежность x множеству B. Этот факт запишем в виде

$A \subseteq B \Longrightarrow \forall x (x \in A \Rightarrow x \in B)$.

Наоборот, если из принадлежности любого элемента x множества A следует принадлежность x множеству B, то $A \subseteq B$. Этот факт запишем в виде

$(\forall x \in A \Rightarrow x \in B) \Longrightarrow A \subseteq B$.

Объединив оба этих высказывания получим

$$A \subseteq B \Longleftrightarrow \forall x (x \in A \Rightarrow x \in B). \tag{1.1}$$

Эта запись не что иное, как сокращенная запись определения подмножества. Здесь

$\Longrightarrow, \Rightarrow$ – обозначение *логического следования*;

$\Longleftrightarrow, \Leftrightarrow$ – логическая *эквивалентность*, часто читается "тогда и только тогда, когда" или "необходимо и достаточно";

$\forall_{x \in A} P(x)$ – *квантор всеобщности*; читается "для всех (для каждого)

x из A $P(x)$". Если $x \in U$, то вместо $\forall_{x \in U} P(x)$ пишут просто $\forall x P(x)$. $\exists x P(x)$ – *квантор существования*; читается "существует $x \in U$ такой, что $P(x)$".
Подчеркнем, что выражение (1.1) на самом деле является аббревиатурой, а не логической формулой в строгом смысле. В дальнейшем такого рода сокращения будут часто использоваться.

1.2 Операции над множествами

Определение. ***Объединение*** *множеств A и B определяется следующим образом:*

$A \cup B = \{x | x \in A \vee x \in B\}$,

(множество элементов $x \in U$ таких, что x принадлежит множеству A или x принадлежит множеству B).
Связка $\vee$ имеет неисключающий смысл, т.е. $x \in A \cup B$, если истинно следующее высказывание:

$$(x \in A \wedge x \in B) \vee (x \in A \wedge x \notin B) \vee (x \notin A \wedge x \in B).$$

На диаграммах (Рис. 1.4) объединение множеств $A \cup B$ представлено заштрихованными областями.

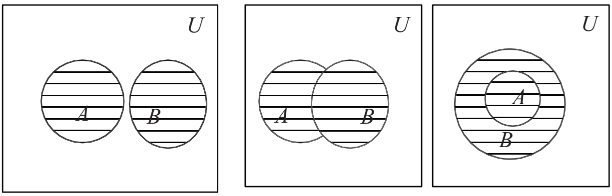


Рис. 1.4. Объединение множеств $A \cup B$

Определение. ***Пересечение*** *множеств A и B определяется следующим образом:*

$A \cap B = \{x | x \in A \wedge x \in B\}$,

(множество таких элементов $x \in U$, что x принадлежит одновременно множеству A и множеству B).

Если $A \cap B = \varnothing$, то множества A и B называют *непересекающимися*.

На диаграммах (Рис. 1.5) пересечение $A \cap B$ представлено заштрихованными областями.

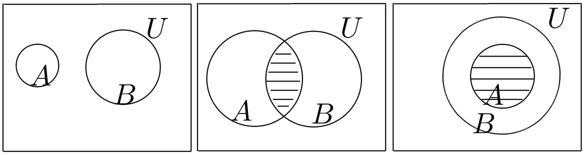


Рис. 1.5. Пересечение множеств $A \cap B$

Пример 1.16.

$A = \{1, 3, 5, 6\}, B = \{1, a, 2, 3, b\}, C = \{a, b, c\}$;
$A \cup B = \{1, 3, 5, 6, a, 2, b\}$;
$A \cap B = \{1, 3\}$;
$A \cup C = \{1, 3, 5, 6, a, b, c\}$;
$A \cap C = \varnothing$;
$B \cup C = \{1, a, 2, 3, b, c\}$;
$B \cap C = \{a, b\}$. ◀

Определение. ***Относительное дополнение*** *множества A по отношению к множеству B (**разность** множеств B и A) определяется следующим образом:* $B \setminus A = \{x | x \in B \wedge x \notin A\}$.

На диаграммах (Рис. 1.6) относительное дополнение (разность) $B \setminus A$ представлено заштрихованными областями.

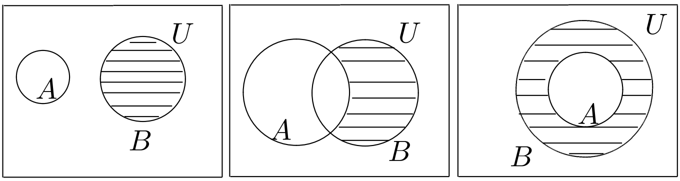


Рис. 1.6. Относительное дополнение $B \setminus A$

Определение. ***Абсолютное дополнение*** $\bar{A}$ *множества A определяется следующим образом:* $\bar{A} = U \setminus A$, *где U – универсальное множество.*

На диаграмме (Рис. 1.7) $\bar{A}$ представлено заштрихованной областью.

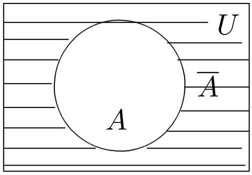


Рис. 1.7. Абсолютное дополнение $\bar{A}$

Замечание. Высказывание $x \in U = 1$ (тождественно истинно), а высказывание $x \in \varnothing = 0$ (тождественно ложно). Тогда $\bar{A} = U \setminus A = \{x | x \in U \wedge x \notin A\} = \{x | 1 \wedge x \notin A\} = \{x | x \notin A\}$. Значит, $\bar{A} = \{x | x \in \bar{A}\} = \{x | x \notin A\}$, то есть высказывания $x \in \bar{A}$ и $x \notin A$ эквивалентны. Следовательно, разность множеств $B \setminus A$ можно представить в виде

$$B \setminus A = \{x | x \in B \wedge x \in \bar{A}\} = B \cap \bar{A}\ .$$

Примеры 1.17.

Пусть $A = \{a, b, c\}, B = \{c, d, e\}$.

1. $A \setminus B = \{a, b\}$;

2. $B \setminus A = \{d, e\}$;

3. Рассмотрим множество $C = B \cap \bar{A}$. В принципе для его определения необходимо знать универсум для множества A: в C войдут все элементы, которые одновременно присутствуют в множествах B и $\bar{A}$. Но множество $\bar{A}$ содержит все элементы, которых нет в A. Следовательно C содержит все элементы из B, которых нет в A, т.е. $C = B \setminus A = \{d, e\}$. ◄

Определение. ***Симметрическая разность*** *множеств A и B определяется следующим образом:*

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

В соответствии с предыдущим замечанием симметрическую разность можно представить в виде

$$A \oplus B = (A \cap \bar{B}) \cup (B \cap \bar{A}).$$

На диаграммах (Рис. 1.8) симметрическая разность $A \oplus B$ представлена заштрихованными областями.

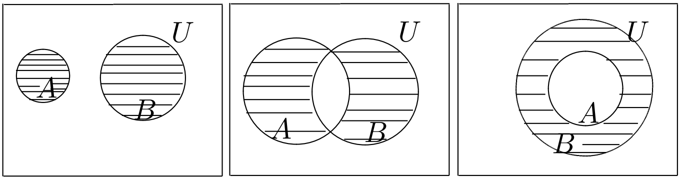


Рис. 1.8. Симметрическая разность $A \oplus B$

Пример 1.18.

$A = \{a, b, c\}, B = \{c, d, e\}$;

$A \oplus B = \{a, b, d, e\}$. ◄

Определение. *Пусть* $\mathcal{A} = \{A_{i_1}, A_{i_2}, A_{i_3}, \ldots\}$ *– семейство множеств, где* $A_{i_k} = A_{i_l}$*, если* $i_k = i_l$*. Тогда элементы* $\mathcal{A}$ *однозначно идентифицируются элементами множества* $I = \{i_1, i_2, i_3, \ldots\}$*. Поэтому можно записать* $\mathcal{A} = \{A_i | i \in I\}$*. Множество* I *называется* ***индексным*** *множеством, а его элементы* ***индексами.*** *Множество* $\mathcal{A}$ *называется* ***индексированным*** *множеством.*

Примеры 1.19.

1. $I = \{a, b, c\}, \mathcal{A} = \{A_i | i \in I\} = \{A_a, A_b, A_c\}$;
2. $I = \{2, 4, 6\}, \mathcal{B} = \{B_i | i \in I\} = \{B_2, B_4, B_6\}$. ◄

Определение. *Пусть* I *– индексное множество. Обобщим операции объединения и пересечения следующим образом:*

$$\bigcup_{i \in I} A_i = \{x | x \in A_i \text{ хотя бы для одного } i\};$$

$$\bigcap_{i \in I} A_i = \{x | x \in A_i \text{ для всех } i\}.$$

Примеры 1.20.

1. Пусть $I = \{1, 2, 3\}, A_1 = \{a, b, c\}, A_2 = \{b, d\}$, $A_3 = \{b, e\}$.

$$\bigcup_{i \in I} A_i = A_1 \cup A_2 \cup A_3 = \{a, b, c, d, e\};$$

$$\bigcap_{i \in I} A_i = A_1 \cap A_2 \cap A_3 = \{b\}.$$

2. $\bigcup_{i \in \varnothing} A_i = \varnothing$. Так как не существует ни одного A_i, то совокупность объектов, принадлежащих по крайней мере одному A_i пуста.

3. $\bigcup\limits_{i\in\{1\}} A_i = \bigcap\limits_{i\in\{1\}} A_i = A_1$. ◄

Определение. *Семейство $\mathcal{A} = \{A_i | i \in I\}$ подмножеств множества A называется* ***покрытием*** *множества A, если выполняются следующие условия:*

1) $A_i \neq \varnothing$ для всех $i \in I$;

2) $\bigcup\limits_{i\in I} A_i = A$;

Если выполняется дополнительное условие

3) $A_i \cap A_j = \varnothing$ для всех $i \in I, j \in I$, таких, что $i \neq j$,

то $\mathcal{A}$ называется ***разбиением*** *множества A.*

Множества A_i называют ***классами покрытия (разбиения)*** *множества A.*

В соответствии с этими определениями любое разбиение некоторого множества является одновременно и его покрытием.

Примеры 1.21.

1. $A = \{1, 2, 3, 4\}$.

Примером разбиения A является семейство $\{\{1, 3\}, \{2\}, \{4\}\}$, а примером его покрытия – семейство $\{\{1, 3\}, \{2, 3\}, \{1, 4\}\}$.

2. $\mathcal{A} = \{A_i | A_i$ – множество студентов факультета i $\}$;

$\mathcal{B} = \{B_i | B_i$ – множество студентов, изучающих дисциплину $i\}$;

C – множество студентов ВУЗа.

$\mathcal{A}$ – разбиение C (если считать, что один студент не может учиться на двух факультетах).

$\mathcal{B}$ – покрытие C. ◄

Алгебраические свойства операций

Как уже отмечалось выше, выражения, содержащие операции $\setminus$ и $\oplus$ можно заменить на выражения, содержащие только операции $\cup$, $\cap$, $^{-}$. Рассмотрим некоторые свойства этих операций.

Теорема 1.3 *Пусть A, B, C – подмножества универсума U. Тогда справедливы следующие равенства:*

1. *a) $A \cup B = B \cup A$; b) $A \cap B = B \cap A$;*

2. a) $A \cup (B \cup C) = (A \cup B) \cup C$;

   b) $A \cap (B \cap C) = (A \cap B) \cap C$;

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

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

4. a) $A \cup \varnothing = A$; b) $A \cap U = A$;

5. a) $A \cup \bar{A} = U$; b) $A \cap \bar{A} = \varnothing$.

Доказательство. Доказательство этих равенств опирается на определения операций над множествами, свойства логических операций (см. стр. 19) и на доказательство двустороннего включения множеств (теорема 1.1):

$A = B \Longleftrightarrow A \subseteq B \wedge B \subseteq A$.

Докажем некоторые из этих равенств.

3.a). $\Rightarrow$

$\forall x\ x \in A \cup (B \cap C) \Longrightarrow$ (определение $\cup$)
$x \in A \vee x \in (B \cap C) \Longrightarrow$ (определение $\cap$)
$x \in A \vee (x \in B \wedge x \in C) \Longrightarrow$ (дистрибутивность $\vee$)
$(x \in A \vee x \in B) \wedge (x \in A \vee x \in C) \Longrightarrow$ (определение $\cup$)
$(x \in A \cup B) \wedge (x \in A \cup C) \Longrightarrow$ (определение $\cap$)
$x \in (A \cup B) \cap (A \cup C) \Longrightarrow$
$A \cup (B \cap C) \subseteq (A \cup B) \cap (A \cup C)$

(из того, что $x \in A \cup (B \cap C)$ следует, что $x \in (A \cup B) \cap (A \cup C)$; значит $A \cup (B \cap C) \subseteq (A \cup B) \cap (A \cup C)$).

$\Leftarrow$

$\forall x\ x \in (A \cup B) \cap (A \cup C) \Longrightarrow$ (определение $\cap$)
$x \in (A \cup B) \wedge x \in (A \cup C) \Longrightarrow$ (определение $\cup$)
$(x \in A \vee x \in B) \wedge (x \in A \vee x \in C) \Longrightarrow$ (дистрибутивность $\vee$)
$x \in A \vee (x \in B \wedge x \in C) \Longrightarrow$ (определение $\cap$)
$x \in A \vee x \in (B \cap C) \Longrightarrow$ (определение $\cup$)
$x \in A \cup (B \cap C)$
$\Longrightarrow (A \cup B) \cap (A \cup C) \subseteq A \cup (B \cap C)$

(из того, что $x \in (A \cup B) \cap (A \cup C)$ следует, что $x \in A \cup (B \cap C)$; значит $(A \cup B) \cap (A \cup C) \subseteq A \cup (B \cap C)$). Оба включения и доказывают равенство.

Равенства 4*a*) и 4*b*) также доказываются с использованием свойств логических операций. Здесь F – произвольная логическая формула.

4*a*) $\Rightarrow$

$x \in A \cup \varnothing \Longrightarrow$ (определение $\cup$)

$x \in A \vee x \in \varnothing \Longrightarrow$ $(x \in \varnothing = 0, F \vee 0 = F)$

$x \in A \Longrightarrow A \cup \varnothing \subseteq A.$

$\Leftarrow$

$x \in A \Longrightarrow$ $(x \in \varnothing = 0, F \vee 0 = F)$

$x \in A \vee x \in \varnothing \Longrightarrow$ (определение $\cup$)

$x \in A \cup \varnothing \Longrightarrow A \subseteq A \cup \varnothing.$

Оба включения доказывают равенство.

4*b*) $\Rightarrow$

$x \in A \cap U \Longrightarrow$ (определение $\cap$)

$x \in A \wedge x \in U \Longrightarrow$ $(x \in U = 1, F \wedge 1 = F)$

$x \in A \Longrightarrow A \cap U \subseteq A.$

$\Leftarrow$

$x \in A \Longrightarrow$ $(x \in U = 1, F \wedge 1 = F)$

$x \in A \wedge x \in U \Longrightarrow$ (определение $\cap$)

$x \in A \cap U \Longrightarrow A \subseteq A \cap U.$

Оба включения доказывают равенство множеств. Другие равенства доказываются по таким же схемам. ⊠

Булева алгебра и алгебра множеств

В теореме 1.3 получены соотношения, характеризующие некоторые свойства операций над множествами. Другие свойства операций будут представлены на основе рассмотрения ***булевой алгебры***, которая является примером одного из важнейших понятий математики – *абстрактной алгебры*.

Понятие абстрактной алгебры

Алгеброй называется совокупность, состоящая из множества M – *носителя* алгебры, *сигнатуры* алгебры и множества *особых* элементов. Сигнатура включает в себя систему обозначений, определение операций и аксиомы алгебры. Для построения алгебры вводятся:

1. Базисные символы (примитивы, первичные символы) и знаки операций. Все остальные символы, если они нужны, определяются через примитивы.

2. Система аксиом – совокупность высказываний (утверждений) об основных отношениях между символами. Эти высказывания принимаются без доказательства. Из аксиом по правилам логики выводятся другие высказывания – *теоремы*. В принципе, любое истинное (в данной теории) высказывание является теоремой. На практике теоремами называют наиболее важные утверждения, выведенные из аксиом или из ранее выведенных теорем.

Как правило, на алгебру налагаются следующие требования.

Множество M (носитель) должно быть замкнуто относительно операций – результат операций над элементами M есть элемент M.

Требования к системе аксиом.

1. Независимость. Система аксиом называется независимой, если никакая аксиома не может быть выведена из других аксиом.

2. Непротиворечивость системы аксиом состоит в том, что в этой алгебре нельзя получить противоречие, то есть доказать некоторую теорему и вместе с тем доказать ее отрицание.

3. Полнота: система аксиом является полной, если всякое утверждение, сформулированное в данной теории может быть либо доказано (то есть является теоремой), либо опровергнуто (то есть может быть доказано его отрицание).

Перейдем теперь к *булевой алгебре*. В качестве примитивов этой теории рассматривают:

• множество $\boldsymbol{B} = \{a, b, c, \dots\}$ и его элементы – *носитель* алгебры;

• бинарные операции $+$ (операция сложения) и $\cdot$ (операция умножения; знак $\cdot$ будем, как правило, опускать), относительно которых множество $\boldsymbol{B}$ *замкнуто*, то есть результатом этих операций является всегда некоторый элемент из множества $\boldsymbol{B}$;

• унарная операция $^{-}$ (по аналогии с операцией дополнения множеств назовем эту операцию дополнением), относительно которой множество $\boldsymbol{B}$ замкнуто;

• особые элементы $\mathbf{0}$ и $\mathbf{1}$, принадлежащие $\boldsymbol{B}$. Особенность этих элементов заключается в том, что в то время как символы a, b, c могут обозначать любые элементы из $\boldsymbol{B}$, символы $\mathbf{0}$ и $\mathbf{1}$ обозначают только сами себя.

Порядок выполнения операций регламентируется приоритетами операций (в убывающем порядке): $^{-}, \cdot, +$. Порядок можно изменить с помощью скобок.

Аксиомы булевой алгебры имеют следующий вид:

$$\forall\, a, b, c \in \boldsymbol{B}$$

$$\begin{array}{llll}
1.a) & a + b = b + a; & b) & ab = ba; \\
2.a) & a + (b + c) = (a + b) + c; & b) & a(bc) = (ab)c \\
3.a) & a + (bc) = (a + b)(a + c); & b) & a(b + c) = (ab) + (ac); \\
4.a) & a + \mathbf{0} = a; & b) & a\mathbf{1} = a; \\
5.a) & a + \bar{a} = \mathbf{1}; & b) & a\bar{a} = \mathbf{0}.
\end{array}$$

З а м е ч а н и я. 1. Символы $(,), \in, \forall, \exists, =$ позаимствованы из общей теории множеств и логики и также вводятся в сигнатуру булевой алгебры. Термины бинарная и унарная операции взяты из теории отношений.

2. Результатом операций над элементами множества $\boldsymbol{B}$ является элемент этого же множества (замкнутость $\boldsymbol{B}$ относительно операций). Таким образом, в данном случае равенство "$=$"означает, что левая и правая части равенства представляют один и тот же элемент множества $\boldsymbol{B}$.

3. В рассматриваемой системе аксиомы $2a$) и $2b$) избыточны, так как могут быть выведены из других аксиом. Они включены в систему по традиции и для удобства испоьзования при выводе.

4. Система аксиом булевой алгебры непротиворечива и полна. ►

Предложенные аксиомы объединены в пары и аксиомы каждой пары подобны. Строго это подобие задает следующее

Определение. *Пусть T – высказывание в булевой алгебре. Заменим в T символы $+$, $\cdot$, **1**, **0** в соответствии со следующими правилами:*

$+$ заменяется на $\cdot$,
$\cdot$ заменяется на $+$,

***1** заменяется на **0**,*
***0** заменяется на **1**.*

*Полученное высказывание называется **двойственным** к T.*

Нетрудно убедиться, что все аксиомы с одинаковыми номерами являются двойственными по отношению друг к другу.

Рассмотрим некоторую теорему T булевой алгебры. Рассмотрим вывод T, в котором все высказывания являются аксиомами (такой вывод всегда можно построить, так как если при выводе T использовалась другая теорема, то ее можно заменить соответствующим выводом). Если заменить в этом выводе все аксиомы на двойственные им, то получится вывод другой теоремы, двойственной T. Эти рассуждения являются неформальным доказательством факта, который может быть сформулирован следующим образом:

Теорема 1.4 (***Принцип двойственности***). *Если T – теорема в булевой алгебре, то двойственное ей высказывание также является теоремой.*

Рассмотрим теперь несколько полезных теорем.

Теорема 1.5 *В булевой алгебре $\boldsymbol{B}$ для всех $\forall a, b, c \in \boldsymbol{B}$*

1. $a + a = a$

Д о к а з а т е л ь с т в о .

$$\begin{array}{lcll} a+a & = & (a+a)\mathbf{1} & \text{(аксиома } 4b\text{)} \\ & = & (a+a)(a+\bar{a}) & \text{(аксиома } 5a\text{)} \\ & = & a+(a\bar{a}) & \text{(аксиома } 3a\text{)} \\ & = & a+\mathbf{0} & \text{(аксиома } 5b\text{)} \\ & = & a & \text{(аксиома } 4a\text{)} \end{array}$$

2. $a + \mathbf{1} = \mathbf{1}$.

Д о к а з а т е л ь с т в о .

$$
\begin{aligned}
a + \mathbf{1} &= a + (a + \bar{a}) && \text{(аксиома } 5a\text{)}\\
&= (a + a) + \bar{a} && \text{(аксиома } 2a\text{)}\\
&= a + \bar{a} && \text{(теорема 1.5.1)}\\
&= \mathbf{1} && \text{(аксиома } 5a\text{).}
\end{aligned}
$$

3. $a + (ab) = a$.

Доказательство.

$$
\begin{aligned}
a + (ab) &= (a\mathbf{1}) + (ab) && \text{(аксиома } 4b\text{)}\\
&= a(\mathbf{1} + b) && \text{(аксиома } 3b\text{)}\\
&= a(b + \mathbf{1}) && \text{(аксиома } 1a\text{)}\\
&= a\mathbf{1} && \text{(теорема 1.5.2)}\\
&= a && \text{(аксиома } 4b\text{).}
\end{aligned}
$$

4. $aa = a$

Доказательство.

$$
\begin{aligned}
aa &= (aa) + \mathbf{0} && \text{(аксиома } 4a\text{)}\\
&= (aa) + (a\bar{a}) && \text{(аксиома } 5b\text{)}\\
&= a + (a\bar{a}) && \text{(аксиома } 3a\text{)}\\
&= a\mathbf{1} && \text{(аксиома } 5b\text{)}\\
&= a && \text{(аксиома } 4b\text{)}
\end{aligned}
$$

5. $a\mathbf{0} = \mathbf{0}$

Доказательство.

$$
\begin{aligned}
a\,\mathbf{0} &= a(a\bar{a}) && \text{(аксиома } 5b\text{)}\\
&= (aa)\bar{a} && \text{(аксиома } 2b\text{)}\\
&= a\bar{a} && \text{(теорема 1.5.4)}\\
&= \mathbf{0} && \text{(аксиома } 5b\text{).}
\end{aligned}
$$

6. $a(a + b) = a$.

Доказательство.

$$
\begin{aligned}
a(a + b) &= (a + \mathbf{0})(a + b) && \text{(аксиома } 4a\text{)}\\
&= a + (\mathbf{0}b) && \text{(аксиома } 3a\text{)}\\
&= a + (b\mathbf{0}) && \text{(аксиома } 1b\text{)} \quad \boxtimes\\
&= a + \mathbf{0} && \text{(теорема 1.5.5)}\\
&= a && \text{(аксиома } 4a\text{).}
\end{aligned}
$$

Замечание. Теоремы 1.5. 4 – 6 можно было доказать со ссылкой на принцип двойственности.

Теорема 1.6 *В булевой алгебре для каждого $a \in \boldsymbol{B}$ если $a + b = \mathbf{1}$ и $ab = \mathbf{0}$, то $b = \bar{a}$.*

Доказательство.

$$\begin{aligned}
b &= b+\mathbf{0} && \text{(аксиома } 4a)\\
&= b+(a\bar{a}) && \text{(аксиома } 5b)\\
&= (b+a)(b+\bar{a}) && \text{(аксиома } 3a)\\
&= (a+b)(\bar{a}+b) && \text{(аксиома } 1a)\\
&= \mathbf{1}(\bar{a}+b) && \text{(условие теоремы)}\\
&= (a+\bar{a})(\bar{a}+b) && \text{(аксиома } 5a)\\
&= (\bar{a}\ +a)(\bar{a}+b) && \text{(аксиома } 1a)\\
&= \bar{a}+(ab) && \text{(аксиома } 3a)\\
&= \bar{a}+\mathbf{0} && \text{(условие теоремы)}\\
&= \bar{a} && \text{(аксиома } 4a)
\end{aligned}$$

Аналогично доказывается, что при тех же условиях $a = \bar{b}$. ☒

Теорема 1.7 *В булевой алгебре для всех a*

$$\overline{(\bar{a})} = a.$$

Д о к а з а т е л ь с т в о . В соответствии с аксиомами $5a$ и $5b$

$$a+\bar{a} = \mathbf{1}$$

$$a\bar{a} = \mathbf{0}.$$

В соответствии с теоремой 1.6 имеем $a = \overline{(\bar{a})}$. ☒

Теорема 1.8 *В булевой алгебре* $\bar{\mathbf{0}} = \mathbf{1}$, $\bar{\mathbf{1}} = \mathbf{0}$.

Д о к а з а т е л ь с т в о .

$\mathbf{0}+\mathbf{1} = \mathbf{1}$ (теорема 1.5)

$\mathbf{0}\cdot\mathbf{1} = \mathbf{0}$ (аксиома 4б)

В соответствии с теоремой 1.6 имеем $\mathbf{1} = \bar{\mathbf{0}}$. ☒

Аналогично доказывается, что $\bar{\mathbf{1}} = \mathbf{0}$.

Теорема 1.9 *В булевой алгебре для всех $a, b \in \boldsymbol{B}$:*

1. $\overline{(a+b)} = \bar{a}\,\bar{b}$.
2. $\overline{(ab)} = \bar{a}+\bar{b}$.

Д о к а з а т е л ь с т в о . Если показать, что

1) $(a+b)+(\bar{a}\bar{b}) = \mathbf{1}$,

2) $(a+b)(\bar{a}\bar{b}) = \mathbf{0}$,

то в соответствии с теоремой 1.6 (при $a = (a+b)$ и $b = (\bar{a}\bar{b})$) будем иметь, что $\bar{a}\bar{b} = \overline{(a+b)}$, доказав тем самым теорему. Покажем справедливость соотношений 1) и 2).

1) $(a+b) + (\bar{a}\bar{b}) =$

$$
\begin{array}{lll}
= & ((a+b)+\bar{a})((a+b)+\bar{b}) & \text{(аксиома } 3a) \\
= & ((a+\bar{a})+b)(a+(b+\bar{b}) & \text{(аксиомы } 1a \text{ и } 2a) \\
= & (\mathbf{1}+b)(a+\mathbf{1}) & \text{(аксиома } 5a) \\
= & \mathbf{1}\cdot\mathbf{1} & \text{(аксиома } 1a\text{, теорема 1.5)} \\
= & \mathbf{1} & \text{(аксиома } 4b)
\end{array}
$$

2) $(a+b)(\bar{a}\bar{b}) =$

$$
\begin{array}{lll}
= & ((a+b)\bar{a})\bar{b} & \text{(аксиома } 2b) \\
= & ((\bar{a}a)+(\bar{a}b))\bar{b} & \text{(аксиомы } 1b \text{ и } 3b) \\
= & (\mathbf{0}+(\bar{a}b)\bar{b} & \text{(аксиомы } 1b \text{ и } 5b) \\
= & (\bar{a}b)\bar{b} & \text{(аксиомы } 1a \text{ и } 4a) \\
= & \bar{a}\mathbf{0} & \text{(аксиомы } 2b \text{ и } 5b) \\
= & \mathbf{0} & \text{(следствие из теоремы 1.5)}
\end{array}
$$

Второе утверждение теоремы доказывается аналогично (или с применением принципа двойственности). ☒

Высказывания, сформулированные в теореме 1.9, принято называть ***законами де Моргана***.

Определение. *В булевой алгебре отношение $a \preccurlyeq b$ выполняется тогда и только тогда , когда $a + b = b$.*

Отношение $a \succcurlyeq b$ значит то же самое, что и $b \preccurlyeq a$.

Теорема 1.10 *В булевой алгебре $a + b = b$ тогда и только тогда , когда $ab = a$.*

Доказательство. 1. ⇒ (*необходимость*) Пусть $a + b = b$.

$$
\begin{array}{lll}
a & = a(a+b) & \text{(теорема 1.5.6)} \\
 & = ab & \text{(предположение).}
\end{array}
$$

2. ⇐ (*достаточность*) Пусть $ab = a$.

$$\begin{aligned} b &= b + (ba) && \text{(теорема 1.5.3)} \\ &= b + (ab) && \text{(аксиома 1}b\text{)} \\ &= b + a && \text{(предположение)} \\ &= a + b && \text{(аксиома 1}b\text{).} \quad \boxtimes \end{aligned}$$

Если необходимо построить высказывание двойственное высказыванию $a \preccurlyeq b$, то можно заменить его на высказывание $a + b = b$, и построить двойственное ему высказывание $a\,b = b$, которое по теореме 1.10 равносильно высказыванию $b + a = a$, которое по определению эквивалентно высказыванию $a \succcurlyeq b$. Обобщив эти рассуждения можно дополнить схему подстановок в определении двойственности 21 следующим правилом:

$\preccurlyeq$ заменяется на $\succcurlyeq$,
$\succcurlyeq$ заменяется на $\preccurlyeq$.

Выше рассмотрены основные моменты абстрактной теории, называемой булевой алгеброй. Сами по эти результаты могут показаться чистой игрой ума, однако достоинство абстрактных теорий, которые изучает математика заключается в том, что если соотнести объекты абстрактной теории с объектами реального мира (дать *интерпретацию* абстрактной теории), то при определенных условиях все результаты полученные при изучении абстрактной теории можно перенести на объекты реального мира. Этими условиями являются выполнение аксиом абстрактной теории для объектов реального мира. Если это выполняется, то такая интерпретация называется *моделью* абстрактной теории. Поясним наши рассуждения на примере перехода от булевой алгебры к алгебре множеств.

Булеву алгебру можно рассматривать как набор объектов

$$\langle \boldsymbol{B},\ +, \cdot,\ ^-,\ \preccurlyeq,\ \mathbf{0}, \mathbf{1} \rangle. \tag{1.2}$$

Рассмотрим другой набор

$$\langle\ \mathcal{B}(U),\ \cup,\ \cap,\ ^-,\ \subseteq,\ \varnothing,\ U \rangle, \tag{1.3}$$

где $\mathcal{B}(U)$ – булеан универсума U. Этот набор является интерпретацией булевой алгебры, в которой между примитивами булевой алгебры и объектами теории множеств устанавливается следующее соответствие:

- множество $\boldsymbol{B}$ соответствует $\mathcal{B}(U)$, то есть множеству всех подмножеств некоторого универсума. При этом элементам множества $\boldsymbol{B}$ соответствуют множества, из некоторого универсума U;
- операции $+$ соответствует операция объединения множеств $\cup$;
- операции $\cdot$ соответствует операция пересечения множеств $\cap$;
- операции $^-$ соответствует операция абсолютного дополнения множества $^-$;
- особому элементу $\mathbf{0}$ соответствует пустое множество $\varnothing$;
- особому элементу $\mathbf{1}$ соответствует универсум U ;
- отношению $\preccurlyeq$ соответствует отношение включения множеств $\subseteq$.

Соотношения из теоремы 1.3 (алгебраические свойства операций над множествами) есть аксиомы булевой алгебры, записанные в терминах теории множеств при интерпретации (1.3), следовательно интерпретация (1.3) является моделью булевой алгебры и все результаты булевой алгебры можно перенести на теорию множеств. В частности, в теории множеств для всех элементов $\mathcal{B}(U)$ будут справедливы следующие соотношения:

$$A \cup A = A \qquad \text{(теорема 1.5.1)} \tag{1.4}$$

$$A \cap A = A \qquad \text{(теорема 1.5.4)} \tag{1.5}$$

$$A \cup U = U \qquad \text{(теорема 1.5.2)} \tag{1.6}$$

$$A \cap \varnothing = \varnothing \qquad \text{(теорема 1.5.5)} \tag{1.7}$$

$$A \cup (A \cap B) = A \qquad \text{(теорема 1.5.3)} \tag{1.8}$$

$$A \cap (A \cup B) = A \qquad \text{(теорема 1.5.6)} \tag{1.9}$$

$$\overline{(\overline{A})} = A \qquad \text{(теорема 1.7)} \tag{1.10}$$

$$\overline{\varnothing} = U \qquad \text{(теорема 1.8)} \qquad (1.11)$$

$$\overline{U} = \varnothing \qquad \text{(теорема 1.8)} \qquad (1.12)$$

$$\overline{(A \cup B)} = \overline{A} \cap \overline{B} \qquad \text{(теорема 1.9)} \qquad (1.13)$$

$$\overline{(A \cap B)} = \overline{A} \cup \overline{B} \qquad \text{(теорема 1.9)} \qquad (1.14)$$

Все эти результаты можно было получить в рамках теории множеств, доказывая соответствующие теоремы, однако основное достоинство изучения абстрактной теории заключается в том, что теория множеств является не единственной моделью булевой алгебры и, построив другую модель мы сможем и на нее перенести полученные результаты.

Пример 1.22. Упростить выражение

$$(A \cap B) \cup ((A \cup B) \cap (\overline{A} \cup \overline{B})) =$$

$$\begin{aligned}
&= (A \cap B) \cup ((A \cup B) \cap \overline{(A \cap B)}) && (\overline{(A \cap B)} = \overline{A} \cup \overline{B}) \\
&= ((A \cap B) \cup (A \cup B)) \cap ((A \cap B) \cup && \text{(3}a\text{ из теоремы 1.14)} \\
&\quad \cup \overline{(A \cap B)}) && \\
&= ((A \cap B) \cup (A \cup B)) \cap U && \text{(соотношение 1.8)} \\
&= ((A \cap B) \cup (A \cup B)) && \text{(4}a\text{ из теоремы 1.15)} \\
&= ((A \cup A \cup B) \cap (B \cup A \cup B)) && \text{(3}a\text{ из теоремы 1.15)} \\
&= ((A \cup B) \cap (A \cup B)) && \text{(соотношение 1.4)} \\
&= A \cup B && \text{(соотношение 1.5)} \quad \boxed{\star}
\end{aligned}$$

Соотношения (1.4)–(1.14) задают свойства операций объединения, пересечения и абсолютного дополнения множеств. Но кроме этих операций при изучении множеств в параграфе 1.2 нами были введены еще операции разности и симметрической разности множеств. Однако эти операции можно выразить через объединение, пересечение и абсолютное дополнение, используя следующие соотношения:

$$\begin{aligned}
A \setminus B &= A \cap \bar{B}, \\
A \oplus B &= (A \setminus B) \cup (B \setminus A), \\
&= (A \cap \bar{B}) \cup (B \cap \bar{A}).
\end{aligned}$$

Пример 1.23. Доказать равенство

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

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

$$\begin{array}{lll}
= & (A \cap \overline{B}) \cap (A \cap \overline{C}) & \text{(соотношение 1.13)} \\
= & A \cap \overline{B} \cap A \cap \overline{C} = A \cap \overline{B} \cap \overline{C} & \text{(соотношение 1.3)} \\
= & A \cap (\overline{B} \cap \overline{C}) = A \cap \overline{(B \cup C)} & \text{(соотношние 1.11)} \\
= & A \setminus (B \cup C) & \text{(соотношение 1.13)}
\end{array}$$

Декартово произведение множеств

До сих пор нам было неважно в каком порядке записаны элементы множества. Имеются, однако, объекты, для которых порядок расположения их компонент имеет существенное значение.

Пример 1.24. Координатная пара в геометрии. Точки с координатами $\langle 2, 3\rangle$ и $\langle 3, 2\rangle$ – это разные точки, а множества $\{2, 3\}$ и $\{3, 2\}$ – равны.

Определение. ***Декартовым (прямым) произведением*** $A \times B$ *множеств A и B называется множество* ***всех*** *упорядоченных пар, таких, что первый элемент пары принадлежит A, а второй B:*

$$A \times B = \{\langle a, b\rangle | a \in A \wedge b \in B\}.$$

Примеры 1.25. 1. $A = \{1, 2\}$, $B = \{a, b, c\}$.

$$A \times B = \{\langle 1, a\rangle, \langle 1, b\rangle, \langle 1, c\rangle, \langle 2, a\rangle, \langle 2, b\rangle, \langle 2, c\rangle\},$$

$$B \times A = \{\langle a, 1\rangle, \langle a, 2\rangle, \langle b, 1\rangle, \langle b, 2\rangle, \langle c, 1\rangle, \langle c, 2\rangle\},$$

$$A \times A = \{\langle 1, 1\rangle, \langle 1, 2\rangle, \langle 2, 1\rangle, \langle 2, 2\rangle\}.$$

2. $A = \{1\}, B = \{1, 2, 3\}$

$$(A \times B) \cap (B \times A) = \{\langle 1, 1\rangle, \langle 1, 2\rangle, \langle 1, 3\rangle\} \cap$$

$$\cap \{\langle 1, 1\rangle, \langle 2, 1\rangle, \langle 3, 1\rangle\} = \{\langle 1, 1\rangle\}.$$

3. $A \times (B \times C) = \{\langle a, \langle b, c\rangle\rangle | a \in A, \langle b, c\rangle \in B \times C\}$;

$(A \times B) \times C = \{\langle\langle a, b\rangle, c\rangle | \langle a, b\rangle \in A \times B, c \in C\}$. ◄

Из примера 25.1. видно, что в общем случае $A \times B \neq B \times A$, то есть декартово произведение некоммутативно.

Пример 25.3. показывает, что декартово произведение неассоциативно. Выражение $A\times B\times C$ невозможно интерпретировать без дополнительного соглашения, так как его результат зависит от порядка выполнения операций. Это соглашение заключается в следующем: считается, что умножение выполняется слева направо, то есть выражение $A\times B\times C$ интерпретируется как $(A\times B)\times C$. В общем случае декартово произведение n множеств определяется следующим образом. Пусть I_n – индексное множество, состоящее из натуральных чисел от 1 до n ($I_n=\{1,2,\ldots,n\}$). Тогда

1) $\underset{i\in I_1}{\times} A_1 = A_1$;

2) $\underset{i\in I_k}{\times} A_i = (\underset{i\in I_{k-1}}{\times} A_i)\times A_k$ $(k>1)$.

При этом элементы множества $A\times B\times C$ будем задавать просто тройкой вида $\langle a,b,c\rangle$ вместо более громоздкого обозначения в примере 25.3.

Определение. *m-той декартовой степенью множества A (обозначается как A^m) называется произведение*

$$\underbrace{A\times A\times\cdots\times A}_{m\ \text{раз}},$$

состоящее из m сомножителей.

Пример 1.26. $A^4 = A\times A\times A\times A$.

1.3 Элементы комбинаторики

"Комбинаторика (комбинаторный анализ) — раздел математики, изучающий дискретные объекты, множества (сочетания, перестановки, размещения и перечисления элементов) и отношения на них (например, частичного порядка). Комбинаторика связана со многими другими областями математики — алгеброй, геометрией, теорией вероятностей, и имеет широкий спектр применения в различных областях знаний (например в генетике, информатике, статистической физике).

Термин "комбинаторика" был введён в математический обиход Лейбницем, который в 1666 году опубликовал свой труд "Рассуждения о комбинаторном искусстве".

Иногда под комбинаторикой понимают более обширный раздел дискретной математики, включающий, в частности, теорию графов." (Википедия – Комбинаторика)

Определение. *Пусть A – множество и $|A| = n$. Упорядоченный набор из m элементов $\langle a_1, a_2, \ldots, a_m \rangle \in A^m$ называется* ***кортежем*** порядка m над A, или m-***кортежем*** над A.

Если все a_i в m-кортеже различны, то он называется ***размещением*** *порядка m над A или m-**размещением** над A. В частности, n-размещение называется **перестановкой** A.*

*Множество всех m-элементных подмножеств множества A называется множеством m-**сочетаний** над A.*

Обозначим:

$S_m(A)$ – множество всех m-кортежей над A;

$P_m(A)$ – множество всех m-размещений над A;

$P(A)$ – множество всех перестановок A;

$C_m(A)$ – множество всех m-сочетаний над A.

Пример 1.27.

1. $A = \{a, b, c\}$;

$S_2(A) = \{\langle a, a\rangle, \langle a, b\rangle, \langle a, c\rangle, \langle b, a\rangle, \langle b, b\rangle, \langle b, c\rangle,$
$\langle c, a\rangle, \langle c, b\rangle, \langle c, c\rangle\}$;

$P_2(A) = \{\langle a, b\rangle, \langle a, c\rangle, \langle b, a\rangle, \langle b, c\rangle, \langle c, a\rangle, \langle c, b\rangle\}$;

$P(A) = \{\langle a, b, c\rangle, \langle a, c, b\rangle, \langle b, a, c\rangle, \langle b, c, a\rangle,$
$\langle c, a, b\rangle, \langle c, b, a\rangle\}$.

$C_2(A) = \{\{a, b\}, \{a, c\}, \{b, c\}\}$. ◀

Очевидно, что для произвольного A выполняется соотношение
$P_k(A) \subseteq S_k(A)$, точнее $P_1(A) = S_1(A)$ и $P_k(A) \subset S_k(A)$ при $k > 1$. Если $k > n$, то $P_k(A) = \varnothing$, но $S_k(A) \neq \varnothing$, так для предыдущего примера $\langle a, a, b, b, c, a\rangle \in S_6(A)$.

Следующая теорема носит вспомогательный характер и используется далее при доказательстве теоремы 1.12.

Теорема 1.11 (Правило произведения) *Рассмотрим конечные семейства множеств* $\{A_i | i \in I\}, \{M_i | i \in I\}$. *С каждым конечным множеством* M_i *свяжем число* n_i $(n_i \leq |A_i|)$. *Множество* M_i *определим следующим образом:*

1) $M_1 \subseteq A_1$ *и* $|M_1| = n_1$*;*

2) *при* $i > 1, M_i \subseteq M_{i-1} \times A_i$*, где* M_i *конструируется соединением каждого элемента из* M_{i-1} *ровно с* n_i *элементами из* A_i *(причем не обязательно с одними и теми же, то есть если один из элементов* M_{i-1} *сочетается с какими-либо одними* n_i *элементами из* A_i*, то другой элемент* M_{i-1} *может сочетаться с другими* n_i *элементами).*

Тогда $|M_m| = \prod\limits_{i=1}^{m} n_i$.

Д о к а з а т е л ь с т в о проведем индукцией по m. .

Базис индукции. $m = 1$. По определению

$$|M_1| = n_1 = \prod_{i=1}^{1} n_i.$$

Шаг индукции. Предположим, что теорема справедлива для $m-1$ и покажем ее справедливость для m. Элементы M_m получаются сочетанием каждого из элементов M_{m-1} точно с n_m элементами A_m, то есть $|M_m| = |M_{m-1}| \cdot n_m$.

Заключение индукции. По предположению индукции

$$|M_{m-1}| = \prod_{i=1}^{m-1} n_i.$$

Отсюда $|M_m| = \prod\limits_{i=1}^{m-1} n_i \cdot n_m = \prod\limits_{i=1}^{m} n_i$, что и требовалось доказать. ☒

Примеры 1.28.

1. Пусть $A_1 = \{a, b, c\}$, $A_2 = \{1, 2, 3\}$, $A_3 = \{+, *, \#\}$,

$n_1 = 2, n_2 = 3, n_3 = 2.$

$M_1 = \{a, c\}$,

$M_2 = \{\langle\, a, 1\rangle, \langle\, a, 2\rangle, \langle\, a, 3\rangle, \langle\, c, 1\rangle, \langle\, c, 2\rangle, \langle\, c, 3\rangle\}$;

$M_3 = \{\langle\, a, 1, +\rangle, \langle\, a, 1, *\rangle, \langle\, a, 2, *\rangle, \langle\, a, 2, \#\rangle,$

$\langle\, a, 3, \#\rangle, \langle\, a, 3, +\rangle, \langle\, c, 1, +\rangle, \langle\, c, 1, \#\rangle,$

$\langle\, c, 2, *\rangle, \langle\, c, 2, +\rangle, \langle\, c, 3, *\rangle, \langle\, c, 3, \#\rangle\}$.

2. Если $n_i = |A_i|$ для всех i, то $M_m = \mathop{\times}\limits_{i \in I} A_i$ и

$$|M_m| = \prod_{i-1}^{n_m} |A_i|. \blacktriangleleft$$

Теорема 1.12 *Пусть A – множество и $|A| = n$. Тогда*

1. $|S_m(A)| = n^m$;
2. $|P_m(A)| = P(n, m)$;
3. $|C_m(A)| = C(n, m)$;

Доказательство.

1. В теореме 1.11 положим $A_1 = A_2 = \cdots = A_m = A$ и $n_1 = n_2 = \cdots = n_m = n$. Тогда $|M_m| = n^m$, но $M_m = A^m = S_m(A)$.

2. В теореме 1.11 положим $A_1 = A_2 = \cdots = A_m = A$ и $n_1 = n$. При этом $M_1 = A = P_1(A)$. Построим M_2 таким образом, чтобы оно совпадало с $P_2(A)$. Так как в размещении все элементы должны быть различны, то каждый элемент из M_1 может сочетаться только с $n-1$ элементами из A, отличными от данного элемента. Таким образом, $n_2 = n - 1$. Чтобы построить $M_3 = P_3(A)$, рассуждая аналогично, можно прийти к выводу, что $n_3 = n - 2$. Проводя аналогичные рассуждения мы получим, чтобы построить $M_m = P_m(A)$, следует взять $n_m = n - m + 1$. Таким образом

$$|P_m(A)| = |M_m| = \prod_{i=1}^{m} (n - i + 1) =$$

$$= n \cdot (n-1) \cdot \dots \cdot (n - m + 1) = P(n, m).$$

3. Число m-сочетаний над A равно числу подмножеств A, имеющих m элементов. Пусть N – число этих подмножеств. Рассмотрим перестановки каждого подмножества. Их будет $P(m, m) = m!$. Перестановки для различных подмножеств будут различны, но их число для любого подмножества будет одинаковым. Общее число таких перестановок будет $N \cdot m!$. Все эти перестановки образуют $P_m(A)$, то есть

$$N \cdot m! = |P_m(A)| = P(n, m),$$

$$|C_m(A)| = N = \frac{P(n, m)}{m!} = C(n, m).$$

Глава 2

Бинарные отношения

2.1 Определения и примеры

Пусть заданы множества M и L.
Определение. ***Бинарным отношением*** *называется подмножество*

$$R \subseteq M \times L.$$

Если R отношение, то будем писать: $x \in M, y \in L$, $\langle x, y\rangle \in R$, либо $(x, y) \in R$, либо xRy и говорить: x находится в отношении R к y, либо выполняется ***соотношение*** xRy.

Примеры 2.1.
1. S — множество групп на факультете, T — множество преподавателей на факультете.

$$R \subseteq S \times T, xRy, x \in S, y \in T;$$

xRy — группа x слушает лекции преподавателя y.
2. Отношение "быть старше":
$R \subseteq M \times M, xRy - x$ старше y.
3. Отношение "быть севернее": $R \subseteq M \times M, xRy$ — город x находится севернее города y.

4. Отношение "быть знакомым с": xRy — x знаком с y.
5. Отношение $<$: $\langle 2,5\rangle \in <, \langle 5,2\rangle \notin <$. ◄

З а м е ч а н и е. Иногда бинарным отношением называют только $R \subseteq M \times M$, а $R \subseteq M \times L$ называют *соответствием*. ►

Если $R \subseteq M \times M$, то отношение также обозначают в виде упорядоченной пары $\langle R, M \rangle$, где $R \subseteq M \times M$, и говорят — отношение R определено на множестве M. Для $R \subseteq M \times L$ обозначение $\langle R, M, L \rangle$.

В общем случае рассматриваются n-*арные* (или n-*местные*) отношения $R \subseteq M_1 \times M_2 \times \cdots M_n$, то есть множество кортежей вида $\langle x_1, x_2, \ldots, x_n \rangle$, где $x_i \in M_i$. В частности, все M_i могут совпадать.

Примеры 2.2.

1. Тернарное (трехместное) отношение $\langle x, y, z \rangle$ для $x + y = z$.
2. Расписание занятий.

День недели	Предмет	Препод.	Ауд.	Время
Понедельник	Мат.анализ	Терпугов	104	$8^{45} - 10^{20}$
Понедельник	Основы прогр.	Костюк	104	$10^{35} - 12^{10}$
...	...	...	...	...

Здесь мы подошли к способам задания отношений. Расписание — это пример табличного способа задания (точно такой же как и способ задания множества перечислением, ведь отношение — множество).

Способы задания бинарного отношения

1. Бинарное отношение можно задавать ***перечислением*** всех упорядоченных пар – элементов отношения (например, в виде двумерного массива $R[1..n, 1..2]$ в нотации языка Паскаль).

2. Множество $M \times M$ является универсальным бинарным отношением на M. Любое отношение R можно задать с помощью ***функции принадлежности***:

$$\mu_R(x, y) = \begin{cases} 0, & \langle x, y \rangle \notin R \\ 1, & \langle x, y \rangle \in R \end{cases}$$

для любой пары $\langle x, y \rangle \in M \times M$.

3. Задание отношения ***матрицей***.

Пусть $|M| = n$ и элементы $M = \{x_1, x_2, \cdots, x_n\}$ пронумерованы целыми числами от 1 до n. Построим квадратную матрицу $R = \|r_{ij}\|$ размерности $n \times n$. i-я строка матрицы соответствует i-му элементу M, j-й столбец — j-му элементу M,

$$r_{ij} = \begin{cases} 1, & \text{если } x_i R x_j \\ 0, & \text{в противном случае.} \end{cases}$$

Матрица отношения $\|r_{ij}\|$ содержит всю информацию о том, какие пары элементов из M принадлежат отношению R.

Замечание. Впредь будем считать нумерацию элементов M фиксированной. Для $n-$элементного множества возможны $n!$ различных нумераций и, соответственно, $n!$ матриц, представляющих данное отношение.

С другой стороны, если задана (0-1)матрица размерности $n \times n$, и фиксирована нумерация на M, то тем самым на M задается некоторое отношение R. ►

Матрица $\|r_{ij}\|$, все элементы которой $r_{ij} = 0$, задает ***пустое*** отношение.

Матрица, все элементы которой $r_{ij} = 1$, задает ***полное*** (универсальное) отношение $M \times M$.

Матрица $\Delta = \|\delta_{ij}\|$, где

$$\delta_{ij} = \begin{cases} 1, & \text{если } i = j \\ 0, & \text{если } i \neq j \end{cases}$$

задает ***диагональное*** отношение (***единичное***, или отношение ***равенства***).

Матрица с элементами $r_{ij} = 1 - \delta_{ij}$ задает ***антидиагональное*** отношение.

Матрицы пустого, полного, диагонального и антидиагонального отношений не зависят от выбора нумерации элементов.

4. ***Дескриптивное*** задание отношения (с помощью *характеристического предиката*). В этом случае отношение

$$R = \{\langle x, y\rangle | P(x, y)\}$$

определяется как множество пар $\langle x, y\rangle$, таких, что высказывание $P(x, y)$ — истинно.

Пример 2.3. x выше y, если рост $(x) >$ рост (y). ◄

5. Представление отношения ***графом***.

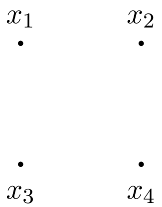
Одно из определений графа: граф $G = (M, R)$ задан, если задано множество M и бинарное отношение R на нем. Элементы множества M называются ***вершинами*** графа, а пары $\langle x, y\rangle \in R$ – ***дугами*** (пары вида $\langle x, x\rangle$ называются ***петлями***).

Для небольших размерностей удобно иллюстрировать отношения (графы) с помощью диаграммы. Элементы множества M (вершины графа) изображаются в виде точек (или небольших окружностей) на плоскости; если пара $\langle x, y\rangle \in R$, то x, y соединяют стрелкой, идущей из x в y. Эту диаграмму также называют графом. Граф, определенный таким образом, называется ***ориентированным*** (***орграфом***).

Таким образом, бинарное отношение и соответствующий ему граф являются ***представлениями*** друг друга: если задано бинарное отношение, то задан и соответствующий ему граф, и наоборот, если задан граф, то задано и соответствующее ему бинарное отношение.

Примеры 2.4.

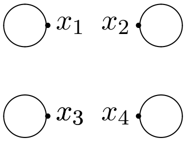
1. Пустому отношению соответствует граф без дуг и петель (пустой граф). Все элементы матрицы пустого отношения – нулевые.



$\varnothing$	x_1	x_2	x_3	x_4
x_1	0	0	0	0
x_2	0	0	0	0
x_3	0	0	0	0
x_4	0	0	0	0

Рис. 2.1. Пустое отношение (пустой граф)

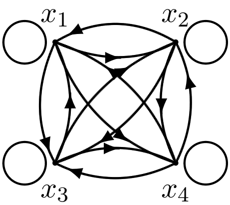
2. Диагональное отношение представляется графом, в каждой вершине которого имеется петля (заметим, что стрелку на петле можно и не изображать). Матрица отношения Δ содержит единицы на главной диагонали, остальные элементы Δ – нули.



Δ	x_1	x_2	x_3	x_4
x_1	1	0	0	0
x_2	0	1	0	0
x_3	0	0	1	0
x_4	0	0	0	1

Рис. 2.2. Диагональное (единичное) отношение

3. Полное отношение представляется *полным* графом. Все элементы матрицы полного отношения – единицы.



$M \times M$	x_1	x_2	x_3	x_4
x_1	1	1	1	1
x_2	1	1	1	1
x_3	1	1	1	1
x_4	1	1	1	1

Рис. 2.3. Полное отношение (полный граф)

Отображения

Соответствие является обобщением понятия бинарного отношения на тот случай, когда отношение является подмножеством прямого произведения различных множеств.

Определение. *Соответствие* Γ *(гамма) между множествами* M *и* L *определено, если задано отношение* $R \subseteq M \times L$, *где*

$R = \{\langle x, y\rangle | x \in M, y \in L\}$.

Γ определяет соответствие элементов множества M элементам множества L. При этом говорят, что R — отношение соответствия Γ, M — область определения, а L — область значений Γ.

Соответствие, обратное Γ, обозначается Γ^{-1}, где L — область определения, а M — область значений Γ^{-1}.

Определение. *Отображением* *множества* M ***во*** *множество* L *называется такое соответствие, которое каждому* $x \in M$ *сопоставляет по крайней мере один* $y \in L$.

Обозначение $\Gamma : M \mapsto L$ указывает на то, что Γ *отображает* множество M на (во) множество L. При этом говорят, что элемент $y \in L$ есть *образ* элемента $x \in M$, а x — прообраз (переменная, аргумент) y.

Примеры 2.5.

1. $M = \{a, b, c, d\}; L = \{A, B, C, D, E\}$;

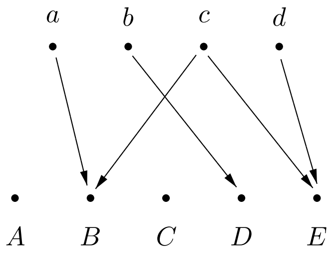


Рис. 2.4. Отображение $\Gamma : M \mapsto L$

$\Gamma a = \{B\}$, $\Gamma b = \{D\}$, $\Gamma c = \{B, E\}$, $\Gamma d = \{E\}$:
B есть образ a, D есть образ b, B есть образ c, E есть образ c, E есть образ d.

Из любого $x \in M$ на графе исходит по крайней мере одна дуга, то есть $\forall x \in M\ |\Gamma x| \geqslant 1$, значит это соответствие есть отображение.
2. Примеры записи отображений. $f : A \mapsto B$, $\alpha : \mathbb{R}^2 \mapsto \mathbb{R}$ ($\mathbb{R}$ – множество действительных чисел). ◄

Определение. *Отображение M* ***на*** *L $\Gamma : M \mapsto L$ называется* ***сюръективным (сюръекцией)****, если любой $y \in L$ имеет по крайней мере один прообраз, то есть $\forall y \in L\ |\Gamma^{-1} y| \geqslant 1$.*

В этом случае говорят, что M отображается ***на*** L.

Пример 2.6. $M = \{a, b, c, d, e, f\}; L = \{A, B, C, D\}$;
$\Gamma a = \{A\}$, $\Gamma b = \{A\}$, $\Gamma c = \{C\}$, $\Gamma d = \{B\}$, $\Gamma e = \{A, B, D\}$, $\Gamma f = \{C, D\}$;
$\Gamma^{-1} A = \{a, b, e\}$, $\Gamma^{-1} B = \{d, e\}$, $\Gamma^{-1} C = \{c, f\}$, $\Gamma^{-1} D = \{e, f\}$.

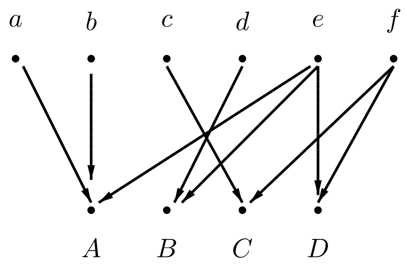


Рис. 2.5. Сюръекция

Каждый элемент L имеет прообраз, следовательно Γ — сюръекция. ◄

Определение. *Отображение $\Gamma : M \mapsto L$ называется* ***инъективным (инъекцией)****, если для каждого элемента $y \in L$ существует не более одного прообраза (либо y вообще не имеет прообраза), то есть $\forall y \in L\ |\Gamma^{-1} y| \leqslant 1$.*

В этом случае говорят, что M отображается ***в*** L.

Пример 2.7. $M = \{a, b, c, d\}; L = \{A, B, C, D, E, F\}$;
$\Gamma a = \{B\}$, $\Gamma b = \{C\}$, $\Gamma c = \{F\}$, $\Gamma d = \{D, E\}$;
$\Gamma^{-1}A = \varnothing$, $\Gamma^{-1}B = \{a\}$, $\Gamma^{-1}C = \{b\}$, $\Gamma^{-1}D = \{d\}$,
$\Gamma^{-1}E = \{d\}$, $\Gamma^{-1}F = \{c\}$.

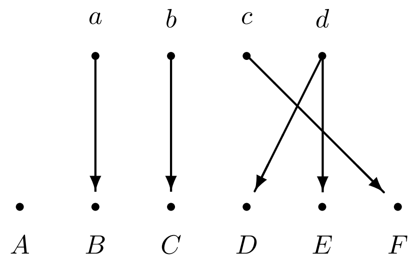


Рис. 2.6. Инъекция

В каждый $y \in L$ входит самое большее одна дуга – отображение инъективное. ◄

Определение. *Если отображение* $\Gamma : M \mapsto L$ *одновременно сюръективно и инъективно, то есть* $\forall y \in L$ $|\Gamma^{-1}y| = 1$, *то оно называется* ***биективным***.

Пример 2.8.
$M = \{a, b, c, d\}, L = \{A, B, C, D, E, F\}$;
$\Gamma a = \{A, B\}$, $\Gamma b = \{C\}$, $\Gamma c = \{D, F\}$, $\Gamma d = \{E\}$;
$\Gamma^{-1}A = \{a\}$, $\Gamma^{-1}B = \{a\}$, $\Gamma^{-1}C = \{b\}$, $\Gamma^{-1}D = \{c\}$,
$\Gamma^{-1}E = \{d\}$, $\Gamma^{-1}F = \{c\}$.

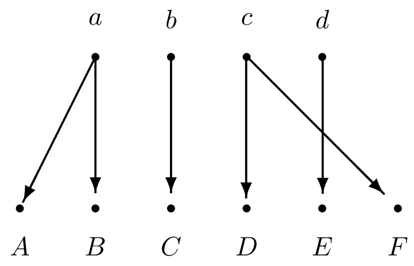


Рис. 2.7. Биекция

В каждый $y \in L$ входит одна и только одна дуга – отображение биективное. ◄

Определение. *Отображение* $\Gamma : M \mapsto L$, *такое, что* $\forall x \in M$ $|\Gamma x| = 1$, *называется* ***функцией***.

Другими словами, функцией M в L называется такое отображение, которое каждому $x \in M$ сопоставляет один и только один $y \in L$.

Функция может быть сюръективной, инъективной, или биективной.

Примеры 2.9.

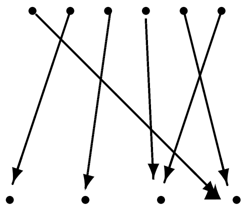


Рис. 2.8. Сюръективная функция

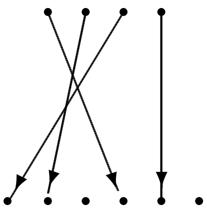


Рис. 2.9. Инъективная функция

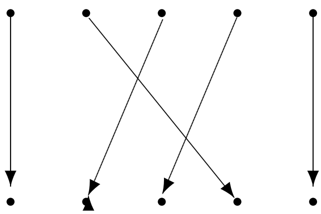


Рис. 2.10. Биективная функция

Если функция Γ биективна, то обратная ей Γ^{-1} тоже биективна. В этом случае говорят о ***взаимно однозначном соответствии***.

З а м е ч а н и е. Некоторые авторы определяют отображение как функцию, то есть $\forall x \in M \ |\Gamma x| = 1$. Мы же определили для отображения $\forall x \in M \ |\Gamma x| \geqslant 1$, для функции $\forall x \in M \ |\Gamma x| = 1$. ▶

Пример 2.10. Бинарные операции над множествами – это отображение $\mathcal{B}(U) \times \mathcal{B}(U) \mapsto \mathcal{B}(U)$; унарная операция дополнения – это отображение из $\mathcal{B}(U)$ в само себя ($\mathcal{B}(U) \mapsto \mathcal{B}(U)$).

Таким образом, бинарные операции – это тернарные отношения, унарные операции – это бинарные отношения. ◀

2.2 Операции над отношениями

В этом разделе будем считать, что все отношения заданы на одном и том же множестве M. Пусть заданы отношения $A \subseteq M \times M$ и $B \subseteq M \times M$. Так как отношения – это множества, то все операции над множествами применимы к отношениям.

Объединение отношений: $A \cup B$.

Соотношение $xA \cup By$ выполняется тогда и только тогда, когда выполнено хотя бы одно из соотношений xAy или xBy:

$xA \cup By \Leftrightarrow xAy \vee xBy$.

Пересечение отношений: $A \cap B$.

Соотношение $xA \cap By$ выполняется тогда и только тогда, когда одновременно выполнены xAy и xBy:

$xA \cap By \Leftrightarrow xAy \wedge xBy$.

Примеры 2.11.

1. Пусть M – множество вещественных чисел.
A – отношение “быть не меньше”: $x \geqslant y$;
B – отношение “быть не равным”: $x \neq y$;
$A \cap B$ – отношение “быть строго больше”: $x > y$.
2. A – отношение “быть больше”: $x > y$;
B – отношение “быть равным”: $x = y$;
$A \cup B$ – отношение “больше или равно”: $x \geqslant y$.
3. Отношение A “быть севернее”, B – "быть западнее";
$A \cap B$ – быть “северо-западнее”. ◀

На отношения переносится и понятие включения $\subseteq$ и строгого включения $\subset$. Например, $< \subset \leqslant$ (отношение $<$ является собственным подмножеством отношения $\leqslant$). Из определения включения (или подмножества) следуют такие его свойства:

если $A \subseteq B$, то из xAy логически следует xBy

$(A \subseteq B \Leftrightarrow xAy \Rightarrow xBy)$,
и, обратно, если из xAy следует xBy, то $A \subseteq B$
$(xAy \Rightarrow xBy \Leftrightarrow A \subseteq B)$.
Отсюда видно, что для любого отношения R

$\varnothing \subseteq R \subseteq U$,
где $\varnothing$ – пустое отношение, $U = M \times M$ – полное (универсальное) отношение.

Определим некоторые операции, не сводящиеся к теоретико-множественным.

Определение. *Отношение $\langle A^{-1}, M\rangle$, **обратное** к данному отношению $\langle A, M\rangle$, определяется условием:*

$xA^{-1}y \Leftrightarrow yAx$ *($xA^{-1}y$ равносильно yAx).*

Примеры 2.12.
1. Если A – отношение $>$, то A^{-1} – отношение $<$.
2. Если A – отношение "быть родителем то A^{-1} – отношение "быть ребенком". ◀

Определение. ***Произведение*** *отношений AB определяется следующим образом:*

$xABy \Leftrightarrow \exists z \in M[xAz \wedge zBy]$
(соотношение $xABy$ выполняется тогда и только тогда, когда существует такой $z \in M$, что одновременно выполняются соотношения xAz и zBy).

Примеры 2.13.
1. Если A – отношение $<$, а B – отношение $>$, то $xABy$ выполнено, если существует z такой, что $x < z$ и $z > y$. Если M – множество целых чисел, то такой z существует всегда: можно взять для примера $z = x + y + 1$. Таким образом, AB есть, в данном случае, полное отношение.
2. Если отношение A есть "быть отцом", то отношение AA есть отношение "быть дедом". ◀

Степень отношения. В последнем примере была задана *степень* отношения: $A^2 = AA$. В общем случае $A^k = \underbrace{AA \cdots A}_{k \text{ раз}}$.

Соотношение xA^ky выполнено тогда и только тогда, когда существует цепочка элементов $z_1, z_2, \ldots, z_{k-1}$, такая, что выполняются соотношения $xAz_1, z_1Az_2, \cdots, z_{k-1}Ay$. Это высказывание мы запишем следующим образом:

$$xA^ky \Leftrightarrow \exists z_1, z_2, \ldots, z_{k-1}[xAz_1 \wedge z_1Az_2 \wedge \cdots \wedge z_{k-1}Ay].$$

Определение. ***Транзитивное замыкание*** *$\widehat{A}$ отношения A определяется следующим образом:*

$$\widehat{A} = A \cup A^2 \cup A^3 \cup \cdots \cup A^k \cup \cdots .$$

Соотношение $x\widehat{A}y$ можно представить в виде

$$x\widehat{A}y \Leftrightarrow xAy \vee \exists z_1[xAz_1 \wedge z_1Ay] \vee \exists z_1, z_2[xAz_1 \wedge z_1Ay] \vee \cdots$$

$$\cdots \vee \exists z_1, z_2, \cdots z_{k-1}[xAz_1 \wedge z_1Az_2 \wedge \cdots z_{k-1Ay}] \vee \cdots .$$

Выполнение соотношения $x\widehat{A}y$ равносильно тому, что в отношении A существует одна или несколько цепочек элементов из M

$$x, z_1, z_2, \ldots, z_{k-1}, y \quad (1 \leqslant k \leqslant \infty),$$

такие, что между соседними элементами в цепочке выполнено соотношение A:

$$xAz_1 \wedge z_1Az_2 \wedge \cdots \wedge z_{k-1}Ay.$$

В частности такая цепочка может быть бесконечной, если в ней содержится цикл (хотя бы петля).

Пример 2.14. Для фрагмента отношения A, представленном на рисунке 2.11, выполняются соотношения $x\widehat{A}x$ и $x\widehat{A}y$, так как существуют цепочка $xAz_1 \wedge z_1Az_2 \wedge z_2Az_3 \wedge z_3 \wedge z_3Ax$ и несколько цепочек из x в y.

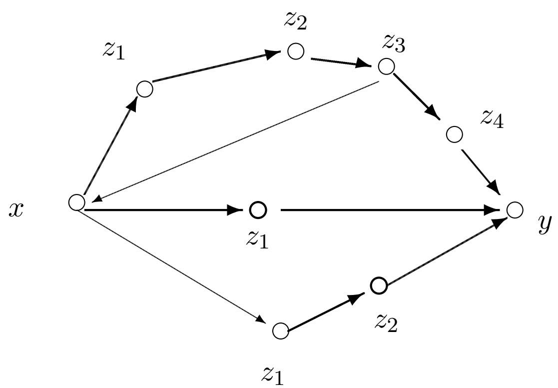


Рис. 2.11. Фрагмент отношения A

2.2.1 Выполнение операций над отношениями

Продемонстрируем на примере операции над отношениями, представленными массивами размерности $[1:N, 1:2]$.

$$A = \begin{vmatrix} ab \\ bc \\ bd \\ cb \end{vmatrix}, \qquad B = \begin{vmatrix} ba \\ bc \\ bd \\ cb \\ dc \end{vmatrix}, \qquad AB = \begin{vmatrix} aa \\ ac \\ ad \\ bb \\ bc \\ ca \\ cc \\ cd \end{vmatrix}, \qquad BA = \begin{vmatrix} bb \\ cc \\ cd \\ db \end{vmatrix}$$

Для краткости здесь пары $\langle x, y\rangle$ представлены в виде xy.

Операция ***умножения*** выполняется в точности по определению: соотношение $xABy$ выполнено, если существует такой z, что $xAz \wedge zBy$. Например, $aABd$, так как есть элемент b, такой, что $aAb \wedge bBd$, и так далее.

По существу, операция умножения отношений AB, представленных массивами, выполняется так: из отношения A выбирается очередная пара $\langle x, z\rangle$, а из отношения B выбираются все пары вида $\langle z, y\rangle$, и пары вида $\langle x, y\rangle$ включается в AB (без повторений).

Операция ***объединения*** выполняется еще проще: в $A \cup B$ включаются все пары из A и из B, за исключением повторяющихся.

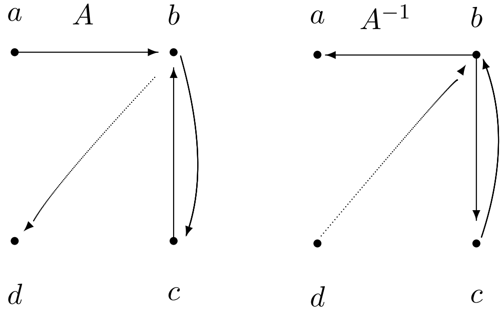
В ***пересечение*** $A \cap B$ выбираются пары, общие для обоих отношений. Для нашего примера

$$A \cup B = \begin{vmatrix} ab \\ bc \\ bd \\ cb \\ ba \\ dc \end{vmatrix}, \qquad A \cap B = \begin{vmatrix} bc \\ bd \\ cb \end{vmatrix}.$$

Обратное отношение A^{-1} получается из A заменой всех пар $\langle x, y\rangle$ на пары $\langle y, x\rangle$. Для нашего примера

$$A^{-1} = \begin{vmatrix} ba \\ cb \\ db \\ bc \end{vmatrix}.$$

На графе направления дуг заменяются на противоположные:



Дополнение. Пусть задано отношение $\langle A, M\rangle$, то есть $A \subseteq M \times M$; $M \times M = U$ – универсальное (полное) отношение.

Дополнение отношения определяется как для обычного множества:

$\overline{A} = U \setminus A = M \times M \setminus A.$

Пример 2.15. Пусть $M = \{a, b, c, d\}$ и $A = \begin{vmatrix} ab \\ bc \\ bd \\ cb \end{vmatrix}.$

Полное отношение состоит из $n^2 = 4^2 = 16$ 2-кортежей над множеством M.

$M^2 = \{\langle a,a\rangle$, $\langle a,b\rangle$, $\langle a,c\rangle$, $\langle a,d\rangle$, $\langle b,a\rangle$, $\langle b,b\rangle$, $\langle b,c\rangle$, $\langle b,d\rangle$, $\langle c,a\rangle$, $\langle c,b\rangle$, $\langle c,c\rangle$, $\langle c,d\rangle$, $\langle d,a\rangle$, $\langle d,b\rangle$, $\langle d,c\rangle$, $\langle d,d\rangle\}$.

$\overline{A} = \{\langle a,a\rangle$, $\langle a,c\rangle$, $\langle a,d\rangle$, $\langle b,a\rangle$, $\langle b,b\rangle$, $\langle c,a\rangle$, $\langle c,c\rangle$, $\langle c,d\rangle$, $\langle d,a\rangle$, $\langle d,b\rangle$, $\langle d,c\rangle$, $\langle d,d\rangle\}$. ◀

Далее предполагается, что нумерация на множестве M уже выбрана, и что матрицы, соответствующие (при данной нумерации) отношениям A и B, обозначены $\|a_{ij}\|$ и $\|b_{ij}\|$. Элементы матриц можно рассматривать как высказывания о принадлежности пары $\langle x_i, x_j \rangle$ ($x_i, x_j \in M$) отношению. Например, $a_{ij} = 1$ означает выполнение соотношения $x_i A x_j$.

Объединение отношений $C = A \cup B$, представленных матрицами $\|a_{ij}\|$, $\|b_{ij}\|$ и $\|c_{ij}\|$ выражается с помощью объединения (булева сложения) матриц, а именно

$$\|c_{ij}\| = \|a_{ij}\| \cup \|b_{ij}\|,$$

элементы которой $c_{ij} = a_{ij} \vee b_{ij}$ (поэлементное булево сложение).

Пересечение отношений $C = A \cap B$ представляется поэлементным булевым умножением соответствующих им матриц: $c_{ij} = a_{ij} \wedge b_{ij}$.

Пример 2.16.

$\|a_{ij}\| =$

A	1	2	3	4
1	0	1	1	0
2	0	1	0	0
3	0	1	0	1
4	0	0	0	0

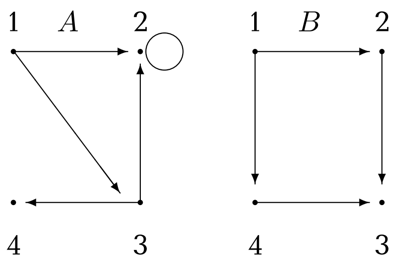
$\|b_{ij}\| =$

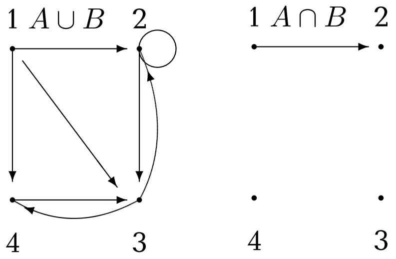
B	1	2	3	4
1	0	1	0	1
2	0	0	1	0
3	0	0	0	0
4	0	0	1	0

$A \cup B$	1	2	3	4
1	0	1	1	1
2	0	1	1	0
3	0	1	0	1
4	0	0	1	0

$A \cap B$	1	2	3	4
1	0	1	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0

Графы, представляющие соответствующие отношения, изображены на рисунке:





Произведение $C = \|c_{ij}\| = AB$ отношений $A = \|a_{ik}\|$ и $B = \|b_{kj}\|$ представляется произведением матриц:

$$c_{ij} = \bigvee_{k=1}^{n} a_{ik} \wedge b_{kj}.$$

Здесь n – порядок матрицы, число элементов множества M; операции умножения и сложения булевы двоичные.

Пример 2.17.

$\|a_{ij}\| =$

A	1	2	3	4
1	0	1	0	0
2	0	0	1	1
3	0	1	0	0
4	0	0	0	0

$\|b_{ij}\| =$

B	1	2	3	4
1	0	0	0	0
2	1	0	1	1
3	0	1	0	0
4	0	0	1	0

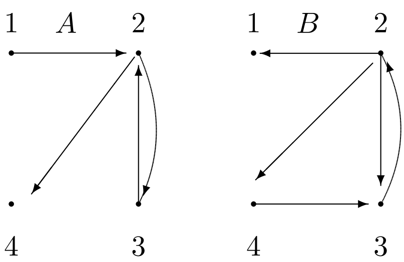
$AB =$

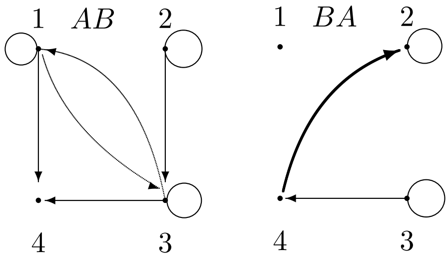
AB	1	2	3	4
1	1	0	1	1
2	0	1	1	0
3	1	0	1	1
4	0	0	0	0

$BA =$

BA	1	2	3	4
1	0	0	0	0
2	0	1	0	0
3	0	0	1	1
4	0	1	0	0

Графы, представляющие эти отношения, изображены на рисунке:





Матрица отношения $A^{-1} = A^{\mathsf{T}}$, где A^{T} – транспонированная матрица A (строки матрицы A заменяются столбцами).

Для нашего примера

$A =$

A	1	2	3	4
1	0	1	0	0
2	0	0	1	1
3	0	1	0	0
4	0	0	0	0

$A^{-1} =$

A^{-1}	1	2	3	4
1	0	0	0	0
2	1	0	1	0
3	0	1	0	0
4	0	1	0	0

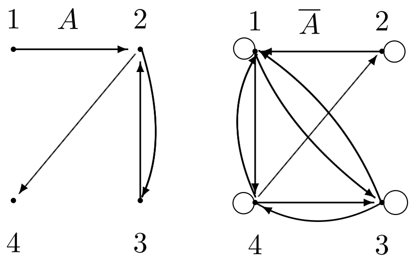
Операция вычитания над элементами $\{0, 1\}$-матрицы отношения не определена. Матрица бинарного отношения $\overline{A}$ строится из матрицы $A = \|a_{ij}\|$ следующим образом:

$$\overline{a}_{ij} = \begin{cases} 0, & \text{если } a_{ij} = 1; \\ 1, & \text{если } a_{ij} = 0; \end{cases}$$

здесь $\overline{a}_{ij}$ – элемент матрицы $\|\overline{a}_{ij}\|$ отношения $\overline{A}$.

Наконец, граф, представляющий дополнение отношения строится так: если в графе исходного отношения A нет дуги $\langle x, y \rangle$, то такая дуга должна появиться в $\overline{A}$.

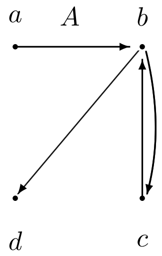
Пример 2.18.



Транзитивное замыкание $\widehat{A} = A \cup A^2 \cup \cdots \cup A^k \cup \cdots$ отношения A выполняется в соответствии с определением. Пусть

$$A = \left| \begin{array}{c} ab \\ bc \\ bd \\ cb \end{array} \right|,$$

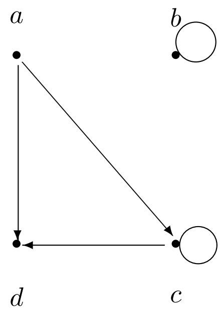
A	a	b	c	d
a	0	1	0	0
b	0	0	1	1
c	0	1	0	0
d	0	0	0	0



массив отношения A, матрица отношения A и граф отношения A, соответственно.

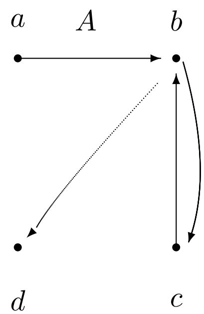
$$A^2 = \left| \begin{array}{c} ac \\ ad \\ bb \\ cc \\ cd \end{array} \right|,$$

A^2	a	b	c	d
a	0	0	1	1
b	0	1	0	0
c	0	0	1	1
d	0	0	0	0



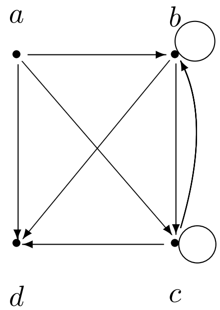
$$A^3 = \left| \begin{array}{c} ab \\ bc \\ bd \\ cb \end{array} \right|,$$

A^3	a	b	c	d
a	0	1	0	0
b	0	0	1	1
c	0	0	0	1
d	0	0	0	0



$$\bigcup_{i=1}^{3} A^i = \begin{vmatrix} ab \\ ac \\ ad \\ bb \\ bc \\ bd \\ cb \\ cc \\ cd \end{vmatrix},$$

$\cup$	a	b	c	d
a	0	1	1	1
b	0	1	1	1
c	0	1	1	1
d	0	0	0	0



Следующие степени A уже ничего не добавит к $\widehat{A}$. Действительно, видно, что $A^3 = A$. Можно проверить, что $A^4 = A^2, A^5 = A^3$ и так далее. Таким образом, в данном случае $\widehat{A} = A \cup A^2 \cup A^3$. В общем случае транзитивное замыкание $\widehat{A}$ "установится" при некоторой степени k.

Алгебраические свойства операций

Отношения – это множества и для них справедливы все соотношения, справедливые для множеств.

Пусть заданы конечное множество M мощности $|M| = n$, универсальное множество $U = M \times M$, отношения $A \subseteq U$, $B \subseteq U$, $C \subseteq U$ и Δ – единичное или диагональное отношение.

Для теоретико-множественных операций справедлива теорема 1.3.

Для других операций докажем ряд теорем, определяющих их основные свойства.

Теорема 2.1 *$AB \neq BA$ – произведение отношений некоммутативно.*

Д о к а з а т е л ь с т в о . Предположение о коммутативности опровергается примерами (см. пример 2.17). ⊠

Теорема 2.2 *$A(BC) = (AB)C = ABC$ – произведение отношений ассоциативно.*

Доказательство.

1. $\Rightarrow$ Из определения произведения следует, что выполнение соотношения $xA(BC)y$ означает, что существует такой z, что выполняются xAz и $z(BC)y$. Из этого, в свою очередь следует, что существует такой w, что выполняются соотношения xAz и zBw и wCy. Из xAz и zBw следует $xABw$, а из $xABw$ и wCy следует $x(AB)Cy$.

Такого рода цепочку рассуждений будем для краткости представлять в следующем виде:

$$\forall x, y[xA(BC)y] \;\Rightarrow\; \exists z[xAz \;\wedge\; z(BC)y] \;\Rightarrow$$
$$\Rightarrow\; xAz \;\wedge\; \exists w[zBw \;\wedge\; wCy] \;\Rightarrow$$
$$\Rightarrow\; (xAz \wedge zBw) \wedge wCy \;\Rightarrow\; x(AB)w \;\wedge\; wCy \;\Rightarrow\; x(AB)Cy,$$

где символ $\Rightarrow$ означает *следует* (логическое следствие); $\exists x[\cdots]$ – существует такой x, что высказывание в скобках $[\cdots]$ истинно; $\wedge$ – логическая связка “и".

2. $\Leftarrow$ Аналогично,

$$x(AB)Cy \;\Rightarrow\; \exists z[x(AB)z \;\wedge\; zCy] \;\Rightarrow$$

$$\Rightarrow\; \exists w[xAw \;\wedge\; wBz \;\wedge\; zCy] \;\Rightarrow\; xAw \;\wedge\; w(BC)y \Rightarrow xA(BC)y.$$

Из двух доказанных утверждений следует справедливость теоремы. ⊠

Из ассоциативности произведения следует, что порядок выполнения операций произволен и в расстановке скобок нет необходимости: вместо $A(BC)$ или $(AB)C$ можно писать просто ABC.

Следующие две теоремы определяют "дистрибутивные" свойства произведения относительно объединения и пересечения.

Теорема 2.3 $(A \cup B)C = (AC) \;\cup\; (BC)$
(умножение распределяется относительно объединения).

Доказательство. 1. $\Rightarrow$ $(A \cup B)C \;\Rightarrow$
$$\Rightarrow\; \forall x, y[x(A \cup B)Cy] \;\Rightarrow\; \exists z[x(A \cup B)z \;\wedge\; zCy] \;\Rightarrow$$
$$\Rightarrow\; (xAz \vee xBz) \;\wedge\; zCy \;\Rightarrow\; (xAz \;\wedge\; zCy) \;\vee\; (xBz \;\wedge\; zCy)$$
$$\Rightarrow\; x(AC)y \;\cup\; x(BC)y \;\Rightarrow\; (AC) \;\cup\; (BC).$$

2. $\Leftarrow$ $(AC) \,\cup\, (BC) \;\Rightarrow\; \forall x, y\ [x(AC) \cup (BC)y] \;\Rightarrow$
$\Rightarrow\; x(AC)y \vee x(BC)y \;\Rightarrow\; \exists z[xAz \,\wedge\, zCy] \;\vee \exists w[xBw \,\wedge\, wCy] \;\Rightarrow$
(так как $A \subseteq A \cup B$; $B \subseteq A \cup B$; $A \subseteq B \Leftrightarrow xAy \;\Rightarrow\; xBy$), выполняются соотношения
$\Rightarrow\; [x(A \cup B)z \;\wedge\; zCy] \;\vee\; [x(A \cup B)w \;\wedge\; wCy] \;\Rightarrow$
$\Rightarrow\; x(A \cup B)Cy \;\Rightarrow\; (A \cup B)C.$ ⊠

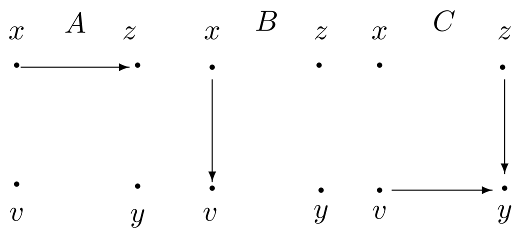
Теорема 2.4 $(A \cap B)C \subseteq (AC) \cap (BC)$
*(умножение **не** распределяется относительно пересечения).*

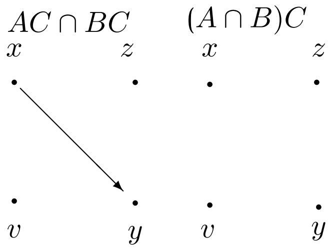
Доказательство. 1. $\Rightarrow$ $(A \cap B)C \;\Rightarrow\; \forall x, y[x(A \cap B)Cy]$ $\Rightarrow\; \exists z[x(A \cap B)z \;\wedge\; zCy] \;\Rightarrow\; [xAz \;\wedge\; zCy]\ \&\ [xBz \;\wedge\; zCy]$ $\Rightarrow\; [x(AC)y \;\wedge\; x(BC)y] \;\Rightarrow\; [x(AC) \cap (BC)y] \;\Rightarrow\; (A \cap B)C \subseteq (AC) \cap (BC).$

2. $\Leftarrow$ Обратное включение в общем случае не выполняется. Действительно,

$$(AC) \cap (BC) \;\Rightarrow\; \forall x, y[xACy \;\wedge\; xBCy] \;\Rightarrow$$

$$\Rightarrow \begin{cases} \exists z \quad [xAz \;\wedge\; zCy] \\ \wedge \\ \exists v \quad [xBv \wedge vCy] \end{cases}$$





Как показано на рисунке, может оказаться, что соотношения $xAz \wedge zCy$, $xBz \wedge zCy$ и соотношения $xAv \wedge vCy$, $xBv \wedge vCy$ не выполняются одновременно, т.е. $A \cap B = \varnothing$. ⊠

Теорема 2.5 $\Delta A = A\Delta = A$; $A\varnothing = \varnothing A = \varnothing$.

Доказательство. Докажем лишь первое утверждение:
$\Delta A \Rightarrow \forall x, y[x\Delta Ay] \Rightarrow \exists z[x\Delta z \wedge zAy] \Rightarrow$
(из определения $\Delta:$ $z = x$) $x\Delta x \wedge xAy \Rightarrow xAy \Rightarrow A$.
$A\Delta \Rightarrow \forall x, y[xA\Delta y] \Rightarrow \exists z[xAz \wedge z\Delta y] \Rightarrow xAy \wedge y\Delta y \Rightarrow xAy \Rightarrow A$. ⊠

Несколько следующих теорем отражают свойства обращения отношений.

Теорема 2.6 $(AB)^{-1} = B^{-1}A^{-1}$.

Доказательство. 1. $\Rightarrow$ $(AB)^{-1} \Rightarrow \forall x, y[x(AB)^{-1}y] \Rightarrow y(AB)x \Rightarrow \exists z[yAz \wedge zBx] \Rightarrow xB^{-1}z \wedge zA^{-1}y \Rightarrow xB^{-1}A^{-1}y \Rightarrow B^{-1}A^{-1}$.

2. $\Leftarrow$ $B^{-1}A^{-1} \Rightarrow \forall x, y[xB^{-1}A^{-1}y] \Rightarrow$
$\Rightarrow \exists z[xB^{-1}z \wedge zA^{-1}y] \Rightarrow yAz \wedge zBx \Rightarrow$
$\Rightarrow y(AB)x \Rightarrow x(AB)^{-1}y \Rightarrow (AB)^{-1}$. ⊠

Теорема 2.7 $(A \cup B)^{-1} = A^{-1} \cup B^{-1}$.

Доказательство. 1. $\Rightarrow$ $(A \cup B)^{-1} \Rightarrow$
$\Rightarrow \forall x, y[x(A \cup B)^{-1}y] \Rightarrow y(A \cup B)x \Rightarrow yAx \vee yBx \Rightarrow$
$\Rightarrow xA^{-1}y \vee xB^{-1}y \Rightarrow x(A^{-1} \cup B^{-1}y \Rightarrow A^{-1} \cup B^{-1}$.

2. $\Leftarrow$ $A^{-1} \cup B^{-1} \Rightarrow \forall x, y[xA^{-1} \cup B^{-1}y] \Rightarrow$
$\Rightarrow xA^{-1}y \vee xB^{-1}y \Rightarrow yAx \vee yBx \Rightarrow$
$\Rightarrow y(A \cup B)x \Rightarrow x(A \cup B)^{-1}y \Rightarrow (A \cup B)^{-1}$ ⊠

Теорема 2.8 $(A \cap B)^{-1} = A^{-1} \cap B^{-1}$.

Доказательство. 1. $\Rightarrow$ $(A \cap B)^{-1} \Rightarrow$
$\Rightarrow \forall x, y[x(A \cap B)^{-1}y] \Rightarrow y(A \cap B)x \Rightarrow yAx \wedge yBx \Rightarrow$
$\Rightarrow xA^{-1}y \wedge xB^{-1}y \Rightarrow x(A^{-1} \cap B^{-1})y \Rightarrow A^{-1} \cup B^{-1}$.

2. $\Leftarrow$ $A^{-1} \cap B^{-1} \Rightarrow \forall x, y[xA^{-1} \cap B^{-1}y] \Rightarrow$
$\Rightarrow xA^{-1}y \wedge xB^{-1}y \Rightarrow yAx \wedge yBx \Rightarrow y(A \cap B)x \Rightarrow$
$\Rightarrow x(A \cap B)^{-1}y \Rightarrow (A \cap B)^{-1}$. ⊠

Теорема 2.9 $(A^{-1})^{-1} = A$.

Доказательство. 1. $\Rightarrow$ $(A^{-1})^{-1} \Rightarrow \forall x, y[x(A^{-1})^{-1}y]$
$\Rightarrow yA^{-1}x \Rightarrow xAy \Rightarrow A$.

2. $\Leftarrow$ $A \Rightarrow \forall x, y[xAy] \Rightarrow yA^{-1}x \Rightarrow$
$\Rightarrow x(A^{-1})^{-1}y \Rightarrow (A^{-1})^{-1}$. ⊠

Отметим некоторые свойства отношения включения.

Теорема 2.10 *Если $A \subseteq B$, то $A^{-1} \subseteq B^{-1}$.*

Доказательство. 1. $\Rightarrow$ Из $A \subseteq B$ следует: xAy влечет xBy, т.е. $xAy \Rightarrow xBy$ (см.стр. 51). Для обратного отношения $yA^{-1}x \Rightarrow yB^{-1}x$. Это значит $A^{-1} \subseteq B^{-1}$ (свойство включения).

2. $\Leftarrow$ $A^{-1} \subseteq B^{-1} \Rightarrow \forall x, y[yAx \Rightarrow yBx] \Rightarrow A \subseteq B$. ⊠

Теорема 2.11 *Если $A \subseteq B$, то $AC \subseteq BC$ и $CA \subseteq CB$.*

Доказательство. Из свойства отношения (стр. 51) $A \subseteq B$ следует $\forall x, z[xAz \Rightarrow xBz]$. Далее,

$$xACy \Rightarrow [xAz \wedge zCy \Rightarrow xBz \wedge zCy] \Rightarrow AC \subseteq BC.$$

Второе утверждение доказывается совершенно аналогично. ⊠

Теорема 2.12 $A \subseteq \widehat{A}$.

Доказательство. Следует непосредственно из определения транзитивного замыкания:

$$\widehat{A} = A \cup A^2 \cup \cdots \cup A^k \cup \cdots. \boxtimes$$

Теорема 2.13 *Если $A \subseteq B$, то $\widehat{A} \subseteq \widehat{B}$.*

Доказательство. Используя теорему 2.11 получим следующую цепочку включений:
$A \subseteq B \;\Rightarrow\; AA \subseteq AB$ (умножили A и B на A слева). Далее, $A \subseteq B \;\Rightarrow\; AB \subseteq BB$ (умножили A и B на B справа). Сопоставляя полученные включения $AA \subseteq AB \subseteq BB$, заключаем: $A^2 \subseteq B^2$. Продолжая подобные действия, для любой степени k получим $A^k \subseteq B^k$ и, следовательно, $\widehat{A} \subseteq \widehat{B}$. ⊠

Приведем без доказательства очевидное свойство:

Теорема 2.14 $\widehat{\widehat{A}} = \widehat{A}$.

2.3 Свойства отношений

Пусть заданы конечное множество M мощности $|M| = n$ и отношения $A, B, C \subseteq M \times M$. Через Δ обозначено, как обычно, диагональное (единичное) отношение.

Определение. *Отношение A называется **рефлексивным,** если*

$\boxed{\Delta \subseteq A}$: $\forall x \in M\; [xAx]$.

К рефлексивным относятся отношения типа $\leqslant, \geqslant$, “быть похожим на", “быть равным", т.е. Δ – также рефлексивное отношение.

Матрица, представляющая рефлексивное отношение, имеет все единицы на главной диагонали. В графе, представляющем рефлексивное отношение, при каждой вершине имеется петля.

Определение. *Отношение A называется **антирефлексивным**, если*

$\boxed{\Delta \cap A = \varnothing}$: $\forall x, y \in M[xAy \;\Rightarrow\; x \neq y]$ *или* $\forall x[\neg xAx]$.

К антирефлексивным относятся отношения типа $<$, $>$, "быть моложе", “быть предком", "быть потомком".

Главная диагональ матрицы, представляющей антирефлексивное отношение, содержит только нули, а в соответствующем графе отсутствуют петли.

Определение. *Отношение A называется **симметричным**, если*

$\boxed{A \subseteq A^{-1}}$: $\forall x, y \in M[xAy \;\Rightarrow\; yAx]$.

Симметричные отношения – это отношения типа =, "быть одинаковым с", "быть похожим на", "быть родственником с".

Элементы матрицы, представляющей симметричное отношение, расположенные симметрично относительно главной диагонали, равны между собой: $a_{ij} = a_{ji}$. В соответствующем графе, если существует дуга, идущая из из вершины x в вершину y, то существует и дуга, идущая из y в x. Часто граф симметричного отношения изображают как *неориентированный* – пара симметричных дуг заменяется *звеном*.

Определение. *Отношение A называется* ***антисимметричным****, если*
$\boxed{A \cap A^{-1} \subseteq \Delta,}$ *т.е. xAy и yAx выполняются одновременно только тогда, когда $x = y$:*

$(\forall x, y \in M\ [xAy\ \wedge\ xA^{-1}y\ \Rightarrow\ x = y])$.

Примерами антисимметричных отношений могут служить $\leqslant$, $\geqslant$. Для элементов матрицы, представляющей антисимметричное отношение, $a_{ij}a_{ji} = 0$, если $i \neq j$. В соответствующем графе любую пару вершин соединяет не более одной дуги, но могут быть и петли.

Определение. *Отношение A называется* ***асимметричным****, если*
$\boxed{A \cap A^{-1} = \varnothing,}$ *т.е. из двух соотношений xAy, $xA^{-1}y$ (yAx) по крайней мере одно не выполняется.*

Примерами асимметричных отношений могут служить $<$, $>$.

В матрице, представляющей асимметричное отношение, $a_{ij}a_{ji} = 0$, т.е. если элемент $a_{ij} = 1$, то $a_{ji} = 0$, но возможно $a_{ij} = a_{ji} = 0$. В соответствующем графе любую пару вершин соединяет не более одной дуги, петли отсутствуют.

Определение. *Отношение A называется* ***транзитивным****, если*
$\boxed{A^2 \subseteq A}$ *, т.е. если выполнены соотношения xAz и zAy, выполнится и xAy :*
$(\forall x, y, z \in M\ [xAz\ \wedge\ zAy\ \Rightarrow\ xAy)$.

Из определения по индукции следует такое свойство: если выполнены соотношения $xAz_1, z_1Az_2, \ldots, z_{n-1}Ay$, выполнится и xAy. На графе, представляющем транзитивное отношение, любая ориентированная цепочка из x в y замыкается дугой

(x, y). Примеры транзитивного отношения: "быть предком", отношения $<$, $>$ на числовой оси и т.д.

Определение. *Отношение A называется* ***антитранзитивным****, если*

$\boxed{A \cap A^k = \varnothing}$ *при всех $k > 1$,*

т.е. если выполнена цепочка соотношений

$x = z_0 A z_1 \ \wedge\ z_1 A z_2 \ \wedge\ \cdots \ \wedge\ z_{k-1} A z_k = y$,

то xAy невозможно, и наоборот, если выполнено соотношение xAy, то цепочка $x = z_0 A z_1 \ \wedge\ z_1 A z_2 \ \wedge\ \cdots \ \wedge\ z_{n-1} A z_k = y$ невозможна.

Пример: отношение непосредственного следования.

Теорема 2.15 *Отношение A симметрично тогда и только тогда, когда*

$$A = A^{-1}.$$

Доказательство. 1. $\Rightarrow$ По определению симметричного отношения $A \subseteq A^{-1}$.

2. $\Leftarrow$ Используя теорему (2.10) ($A \subseteq B \ \Rightarrow\ A^{-1} \subseteq B^{-1}$), для включения $A \subseteq A^{-1}$ получим $A^{-1} \subseteq (A^{-1})^{-1} = A$ (последнее включение следует из теоремы 2.9). ⊠

Теорема 2.16 *Если отношение A асимметрично, то оно антирефлексивно.*

Доказательство (от противного). Предположим, что $\exists x[xAx]$; тогда выполнится и $xA^{-1}x$, а это значит, что $A \cap A^{-1} \neq \varnothing$, что противоречит определению асимметричности. ⊠

Теорема 2.17 *Если отношение A транзитивно, то*

$$A = \widehat{A}.$$

Доказательство. 1. $\Rightarrow$ Включение $A \subseteq \widehat{A}$ следует из определения транзитивного замыкания $\widehat{A} = A \cup A^2 \cup \cdots \cup A^k \cup \cdots$.

2. $\Leftarrow$ Покажем, что при любом n $A^n \subseteq A$.

При $n = 2$ $A^2 \subseteq A$ по определению транзитивности. Предположим, что при некотором $n-1$ $A^{n-1} \subseteq A$. Из предположения индукции и теоремы 2.11 получим $A^n = A^{n-1}A \subseteq AA \subseteq A$. Теперь справедливо включение

$\widehat{A} = A \cup A^2 \cup \cdots \cup A^k \cup \cdots \subseteq A \cup A \cup \cdots \cup A \cup \cdots$.

Оба включения выполнены, следовательно $A = \widehat{A}$. ☒

Теорема 2.18 *Если $A = \widehat{A}$, то A транзитивно.*

Доказательство. $A^2 \subseteq \widehat{A} = A \cup A^2 \cup \cdots$. Так как $A = \widehat{A}$, то $A^2 \subseteq A$ (определение транзитивности). ☒

Теорема 2.19 *Если отношение A рефлексивно и транзитивно, то*

$$A = A^2.$$

Доказательство. 1. $\Rightarrow$ Из рефлексивности

$[xAy \;\Rightarrow\; xAx \;\wedge\; xAy] \;\Rightarrow\; A \subseteq A^2$.

2. $\Leftarrow$ Из транзитивности $[xAy \;\wedge\; yAy \;\Rightarrow\; xAy] \;\Rightarrow\; A^2 \subseteq A$. ☒

2.4 Эквивалентность и толерантность

В этом разделе рассматриваются такие свойства бинарных отношений, которые формализуют понятия равенства и сходства.

Эквивалентность Отношение эквивалентности является математической абстракцией таких понятий как *равенство*, *одинаковость*, *взаимозаменяемость*.

Определение. *Отношение A на множестве M называется отношением **эквивалентности** (**эквивалентностью**), если оно рефлексивно, симметрично и транзитивно.*

Примеры 2.19.

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

2. Для целых чисел “равенство по модулю 2" также является примером эквивалентности.
3. "Равенство" чисел в различных системах счисления – эквивалентность. ◀

Напомним, что *покрытие* множества M – это семейство $\{M_i | i \in I\}$ непустых подмножеств множества M, такое, что

$$M = \bigcup_{i \in I} M_i.$$

Если при этом $M_i \cap M_j = \varnothing, i \neq j$, то $\{M_i | i \in I\}$ – *разбиение* множества M.
Сами подмножества M_i называются ***классами*** данного покрытия или разбиения.

Теорема 2.20 *Пусть задано разбиение $\{M_i | i \in I\}$ множества M и бинарное отношение $\langle A, M\rangle$ определено как "принадлежать общему классу разбиения". Тогда A – эквивалентность.*

Доказательство. Пусть дано разбиение $\{M_i | i \in I\}$ множества M. Так как $M = \bigcup\limits_{i \in I} M_i$, то $\forall x \in M \;\Rightarrow\; x \in M_i$, т.е. каждый элемент x множества M содержится в некотором классе.

1. Очевидно, соотношение $\forall x[xAx]$ выполняется, т.е. A рефлексивно (x и x принадлежат одному классу разбиения).

2. Далее, $xAy \;\Rightarrow\; yAx$ – если пара элементов x, y входит в один класс, то и пара y, x входит в тот же класс, т.е. A симметрично.

3. Пусть теперь выполнены соотношения xAy и yAz. Это значит, что $x, y \in M_i$, а $y, z \in M_j$. Получили, что $M_i \cap M_j \neq \varnothing$. Это может быть лишь в том случае, когда M_i и M_j совпадают, т.е. это один и тот же класс. Значит,

$$x, y \in M_i \;\wedge\; y, z \in M_i \;\Rightarrow\; x, z \in M_i,$$

т.е. выполняется соотношение xAz. А это означает, что отношение A транзитивно. ☒

Теорема 2.21 *Если отношение $\langle A, M\rangle$ – эквивалентность, то существует разбиение $\{M_i | i \in I\}$ множества M такое, что соотношение xAy выполняется тогда и только тогда, когда x и y принадлежат общему классу разбиения.*

Доказательство. Пусть отношение $\langle A, M\rangle$ рефлексивно, симметрично и транзитивно. Построим для каждого элемента $x \in M$ подмножество $M_x = \{z | xAz\}$.

Лемма *Для любых* x, y *либо* $M_x = M_y$, либо $M_x \cap M_y = \varnothing$.

Доказательство. Пусть $M_x \cap M_y \neq \varnothing$. Нужно доказать, что $M_x = M_y$.

Пусть $z \in M_x \cap M_y$, тогда по определению M_x и M_y должны выполняться соотношения xAz и yAz. Из симметричности A следует zAy, а из транзитивности: $xAz \wedge zAy \Rightarrow xAy$. Далее, для любого $w \in M_y$ выполняется yAw. Из xAy и yAw следует xAw, т.е. $w \in M_x$. Это означает, что $M_y \subseteq M_x$.

С другой стороны, для любого $v \in M_x$ выполняется xAv. Из симметричности $xAy \Rightarrow yAx$. Из транзитивности

$yAx \wedge xAv \Rightarrow yAv$, т.е. $v \in M_y$.

Это означает, что $M_x \subseteq M_y$. В итоге, $M_x = M_y$. ⊠

Из леммы и рефлексивности A следует, что множества вида M_x образуют разбиение множества M. Это разбиение называется соответсвующим данному отношению A. Покажем, что такое разбиение (и классы разбиения) удовлетворяют условиям теоремы.

1. $\Rightarrow$ Пусть отношение A рефлексивно, симметрично и транзитивно, и пусть для некоторых $x, y \in M$ выполняется соотношение xAy. Это означает, что $y \in M_x$ (по определению M_x). Но и $x \in M_x$ (из рефлексивности xAx отношения A). Следовательно, оба элемента x и y входят в M_x.

2. $\Leftarrow$ Пусть некоторые элементы u и v содержатся в M_x: $u, v \in M_x$. Для доказательства теоремы нужно показать, что выполняется соотношение uAv.

Действительно, из определения M_x следует xAu и xAv. Из симметричности A следует uAx. Из транзитивности A следует: $uAx \wedge xAv \Rightarrow uAv$. Теорема доказана. ⊠

Замечание 1. Отметим, что доказательство теоремы (2.21) проведено конструктивно. Вначале мы построили подмножества вида $M_x = \{z | xAz\}$, соответствующие отношению A, а затем доказали, что они являются классами разбиения.

З а м е ч а н и е 2. Множество является ***максимальным*** по некоторому свойству, если оно не содержится во множестве, обладающем этим же свойством. Классы эквивалентности, полученные в теореме 2.21, являются максимальными подмножествами попарно эквивалентных элементов множества M. ▶

Мы получили разбиение множества M на подмножества $\{M_i | i \in I\}$ *эквивалентных* друг другу элементов – ***классы эквивалентности***. Множество классов эквивалентности по отношению A принято обозначать M/A (читается: фактор-множество множества M по отношению A).

Эталоны и эквивалентность

Пусть дано разбиение $\{M_i | i \in I\}$ множества M. Выберем в каждом подмножестве M некоторый элемент $e \in M_i$ и назовем его *эталоном* для всякого $x \in M_i$. Отношение $eЭx$ назовем "быть эталоном" или "e – *эталон* для x". Тогда отношение эквивалентности Θ можно определить так:

соотношение $x\Theta y$ выполняется, если x и y имеют общий эталон e, т.е. $eЭx \wedge eЭy$. Таким образом, отношение эквивалентности можно определить с помощью отношения "быть эталоном" и, наоборот, любое отношение "быть эталоном" определяет некоторую эквивалентность.

Пусть Θ – отношение эквивалентности, а $Э$ – такое отношение "быть эталоном", что $x\Theta y$ выполняется тогда и только тогда, когда x и y имеют общий эталон, т.е. выполнение соотношения $x\Theta y$ равносильно существованию z, такого, что $eЭx$ и $eЭy$. Так как $eЭx$ есть $xЭ^{-1}e$, то можно записать: $\Theta = Э^{-1}Э$, т.е. отношение эквивалентности выражается алгебраически через более простое отношение "быть эталоном".

Из определения отношения $Э$ "e является эталоном для x" следуют его свойства:

1. $\forall x \exists e [eЭx]$ – для всякого элемента x существует эталон e.
2. $eЭe$ – любой эталон есть эталон для самого себя.
3. Эталон единствен, т.е. из $e_1Эx$ и $e_2Эx$ следует $e_1 = e_2$.

Эти свойства можно считать аксиомами отношения “быть эталоном". Заметим, что эти аксиомы теперь введены безотносительно к разбиению. Покажем теперь, что из аксиом следует определение эталона с помощью разбиения. Для этого сначала по отношению $Э$ построим новое отношение Θ, определяемое правилом:

$x\Theta y$, если x, y имеют общий эталон, т.е. если существует такой e, что $eЭx$ и $eЭy$; при этом $\Theta = Э^{-1}Э$.

Покажем, что Θ – отношение эквивалентности.
1. Из свойства 1, у каждого x есть эталон и, следовательно, $x\Theta x$, т.е. Θ рефлексивно.
2. Симметричность Θ очевидна: если x, y имеют общий эталон, то и y, x имеют общий эталон.
3. Если $x\Theta y$ и $y\Theta z$, то это значит, что x, y имеют общий эталон, а y не может иметь эталона, отличного от эталона для z (единственность эталона). Значит, $x\Theta z$.

Итак, доказано, что Θ – отношение эквивалентности. Но тогда по теореме (2.21) существует разбиение $\{M_i | i \in I\}$ множества M на классы эквивалентности.

Очевидно, что каждый класс эквивалентности состоит из всех элементов, имеющих общий эталон x. По свойству 2 отношения "быть эталоном" $xЭx$, т.е. x обязательно принадлежит некоторому классу M_i. Таким образом, отношение $Э$, определенное аксиоматически свойствами 1 – 3, всегда может быть задано разбиением с выбранными представителями (эталонами) в каждом классе.

Пример 2.20. $M = \{a, b, c, d, e, f, g\}$;

$Э = \{\langle g, a\rangle, \langle g, e\rangle, \langle d, b\rangle, \langle d, c\rangle, \langle d, f\rangle, \langle d, d\rangle, \langle g, g\rangle\}$;

$\Theta = Э^{-1}Э = \{\langle a, a\rangle, \langle a, e\rangle, \langle a, g\rangle, \langle b, b\rangle, \langle b, c\rangle, \langle b, f\rangle, \langle b, d\rangle, \langle c, b\rangle, \langle c, c\rangle, \langle c, d\rangle, \langle c, f\rangle, \langle d, b\rangle, \langle d, c\rangle, \langle d, d\rangle, \langle d, f\rangle, \langle e, a\rangle, \langle e, e\rangle, \langle e, g\rangle, \langle f, b\rangle, \langle f, c\rangle, \langle f, d\rangle, \langle f, f\rangle$;

Графы, представленные на рисунке, соответствуют отношениям эквивалентности и “быть эталоном".

На графе эквивалентности пары дуг противоположных ориентаций представлены звеньями.

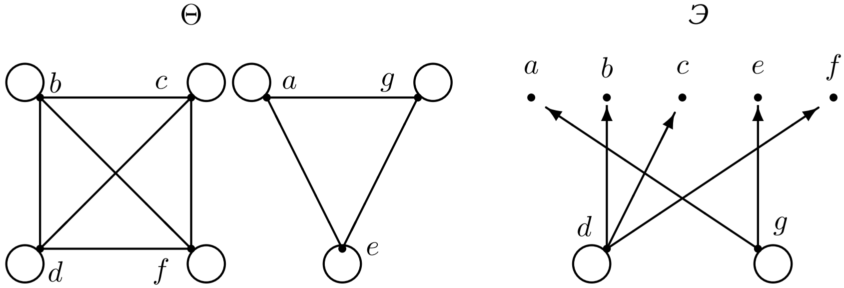


Рис. 2.12. Отношение "быть эталоном" и эквивалентность

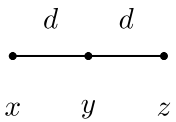
Толерантность

Отношение толерантности формализует понятие сходства, близости.

Определение. *Отношение* $\langle T, M \rangle$ *называется отношением* ***толерантности*** *(**толерантностью**), если оно рефлексивно и симметрично.*

Примеры 2.21.

1. Отношение “близости": будем считать, что два объекта "близки", если расстояние между ними не превышает некоторого порога d. Пусть, например, x близок к y, а y близок к z (см. рисунок); при этом x и z могут быть и не близкими (транзитивность не выполняется).



2. Пусть $\{M_i | i \in I\}$ – некоторое семейство непустых подмножеств множества M. Если $M_i \cap M_j \neq \varnothing, i \neq j$, то отношение пересечения есть отношение толерантности. ◄

Пусть заданы отношения $T, E \subseteq M \times M$.

Теорема 2.22 *Если* T *– толерантность,* E *– эквивалентность и* $T \subseteq E$*,то* $\widehat{T} \subseteq E$*.*

Доказательство. $\widehat{E} = E$ (транзитивность эквивалентности). Из теоремы (2.13) (если $A \subseteq B$, то $\widehat{A} \subseteq \widehat{B}$) $\widehat{T} \subseteq \widehat{E} = E$. ⊠

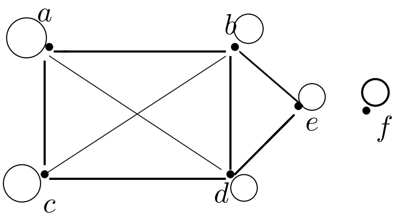
Смысл этой теоремы заключается в том, что транзитивное замыкание $\widehat{T}$ отношения толерантности T есть минимальная эквивалентность, включающая эту толерантность.

Классы толератнтности

По аналогии с классами эквивалентности (см. Замечание 2 на странице 71) введем следующее

Определение. ***Классами толерантности*** *называются максимальные (по включению) подмножества попарно толерантных элементов множества M.*

Пример 2.22. На рисунке изображен граф, представляющий отношение толерантности (пары дуг противоположных направлений заменены звеньями).



Классы толерантности: $\{a, b, c, d\}$, $\{b, d, e\}, \{f\}$. ◄

Классы толерантности образуют покрытие множества M. Действительно, каждый элемент $x \in M$ принадлежит некоторому классу толерантности (может быть – нескольким). Если элемент $x \in M$ нетолерантен никаким другим элементам, то он сам образует класс толерантности.

Число разбиений

Пусть M – конечное множество мощности $|M| = n$. Обозначим некоторое разбиение M на k подмножеств как $P(M) = \{M_i | i \in I\}$, $|P(M)| = k$. Множество разбиений M на k подмножеств обозначим как $P_k(M)$.

Число Стирлинга второго рода $S(n, k)$ определяется как число разбиений n-элементного множества на k подмножеств

$S(n,k) = |P_k(M)|$.

Свойства $S(n,k)$**.**

1. $S(0,0) = 1$ (разбиение пустого множества – пустое множество).
2. $S(n,0) = 0$, если $n > 0$; $S(n,k) = 0$, если $k > n$.
3. $S(n,n) = 1$, если $n \geq 0$.
4. $S(n,k) = S(n-1,k-1) + kS(n-1,k)$ в остальных случаях.

Доказательство. Свойства 1 – 3 очевидны; докажем свойство 4.

Множество всех разбиений на k подмножеств состоит из двух различных классов.

1. Разбиения, которые содержат одноэлементное подмножество $\{n\}$.
2. Разбиения, для которых n является элементом большего, по крайней мере, двухэлементного подмножества.

Мощность первого класса равна $S(n-1,k-1)$, т.е. равна числу разбиений множества $\{1,\dots,n-1\}$ на $k-1$ подмножеств.

Мощность другого класса составляет $kS(n-1,k)$, так как каждому разбиению множества $\{1,\dots,n-1\}$ на k подмножеств соответствует в этом классе в точности k разбиений, образованных добавлением элемента n к каждому блоку. ☒

Пример 2.23. Рассмотрим разбиение множества, состоящего из 5 элементов, на 3-х элементные подмножества.

$S(5,3) = S(4,2) + 3S(4,3)$.

$P_2(M), \|M\| = 4$		$P_3(M), \|M\| = 4$	
$\{1,2,3\},\{4\}$	$\{5\}$	$\{1,2\},\{3\},\{4\}$	$\{1,2,5\},\{3\},\{4\}$ $\{1,2\},\{3,5\},\{4\}$ $\{1,2\},\{3\},\{4,5\}$
$\{1,2,4\},\{3\}$	$\{5\}$	$\{1,3\},\{2\},\{4\}$	$\cdots$
$\{1,3,4\},\{2\}$	$\{5\}$	$\{1,4\},\{2\},\{3\}$	$\cdots$
$\{1,2\},\{3,4\}$	$\{5\}$	$\{1\},\{2,3\},\{4\}$	$\cdots$
$\{1,3\},\{2,4\}$	$\{5\}$	$\{1\},\{2\},\{3,4\}$	$\cdots$
$\{1,4\},\{2,3\}$	$\{5\}$	$\{1\},\{3\},\{2,4\}$	$\cdots$
$\{1\},\{2,3,4\}$	$\{5\}$		

Некоторые числа Стирлинга второго рода

Нерекуррентная формула для чисел Стирлинга второго рода

$$S(n,k) = \frac{1}{k!}\sum_{i=0}^{k}(-1)^i C_k^i (k-i)^n.$$

Число Белла

B_n определяется как число всех разбиений n-элементного множества

$$B_n = \sum_{k=0}^{n} S(n,k).$$

Некоторые числа Белла

Вставить табличку из Липского

Задача о наименьшем покрытии (ЗНП)

Постановка задачи в теоретико-множественной интерпретации

Дано множество $A = \{a_1, \ldots, a_n\}$ и семейство $\mathcal{A} = \{A_1, \ldots, A_m\}$ подмножеств $A_i \subseteq A$. Любое подмножество $\mathcal{A}' = \{A_{j1}, \ldots, A_{jk}\}$ семейства $\mathcal{A}$, такое, что $\bigcup_{i=1}^{k} A_{ji} = A$ образует покрытие множества A, а множества A_{ji} – классы покрытия, или покрывающие множества.

Если в дополнение к этому элементы подмножества $\mathcal{A}'$ попарно не пересекаются, т.е.

$A_{ji} \cap A_{jl} = \varnothing,\ \forall i, l \in \{1, \ldots, k\}, i \neq l$,
то $\mathcal{A}'$ является разбиением множества A.

Если каждому $A_j \in \mathcal{A}$ соответствует вес c_j, то ЗНП формулируется так: найти покрытие множества A, имеющее наименьший вес (стоимость). Для семейства $\mathcal{A}' = \{A_{j1}, \ldots, A_{jk}\}$ вес определяется как сумма $\sum_{i=1}^{k} c_{ji}$.

Аналогично формулируется и задача о *наименьшем разбиении* (ЗНР).

Построим таблицу (матрицу) $\|t_{ij}\|$ такую, что

$$t_{ij} = \begin{cases} 1, & \text{если } a_i \in A_j, \\ 0, & \text{если } a_i \notin A_j, \end{cases}$$

и введем переменную

$$x_j = \begin{cases} 1, & \text{если } A_j \in \mathcal{A}' \\ 0, & \text{если } A_j \notin \mathcal{A}' \end{cases}$$

Упрощение ЗНП.

1. Если в A существует такой элемент a_i, который не входит ни в одно из подмножеств A_j

$(\exists a_i \in A\ [a_i \notin A_j],\ j = 1, \ldots, m)$,

то задача не имеет решения.
2. Если в A существует такой элемент a_i, который принадлежит только одному подмножеству A_k и не принадлежит никакому другому подмножеству A_j

$$(\exists a_i \in A\ [a_i \in A_k\ \&\ a_i \notin A_j], j \neq k),$$

то подмножество A_k должно быть в любом решении, и задачу можно свести к задаче с меньшей размерностью, положив $A = A \setminus \{a_i\}$ и $\mathcal{A} = \mathcal{A} \setminus \{A_k\}$.

З а м е ч а н и е. В случае 1. в матрице $\|t_{ij}\|$ сумма по i-й строке равна 0, а в случае 2. – 1 (единице). ▶
3.Пусть $V_i = \{j | a_i \in A_j\}$;

если существуют $p, q \in \{1, \ldots, n\}$, такие, что $V_q \subseteq V_p$, то a_q можно удалить из A, так как любое подмножество, которое покрывает a_q, покрывает и a_p, т.е. a_p *доминирует* над a_q
4. Если для некоторого подсемейства $\mathcal{A}'' \subset \mathcal{A}$ выполняются соотношения

$$\bigcup_{A_j \in \mathcal{A}''} A_j \supseteq A_k \text{ и } \sum_{A_j \in \mathcal{A}''} c_j \leq c_k \text{ для любых } A_k \in \mathcal{A} \setminus \mathcal{A}'',$$

то A_k можно вычеркнуть, так как $\bigcup\limits_{A_j \in \mathcal{A}''} A_j$ доминирует над A_k.

Предположим, что все эти упрощения выполнены, если они возможны, и исходная ЗНП переформулирована в неприводимой форме.

Рассмотрим сначала задачу о наименьшем разбиении, а затем применим алгоритм для ЗНП.

2.4.7 Алгоритм решения ЗНР

В этом алгоритме используется метод поиска в глубину по дереву решений.

Предварительно переставим столбцы таким образом, чтобы первые столбцы матрицы содержали 1 в первых строках (образуется "блок" из таких столбцов). k-й блок состоит из таких множеств семейства $\mathcal{A}$, в которых содержится элемент k, но отсутствуют элементы с меньшими номерами $a_1, \ldots, a_{k-1}$.

	$A_1^1 A_2^1 \cdots A_l^1$	$A_{l+1}^2 \cdots$	$\cdots$	$\cdots$
a_1	$1\ 1\ \cdots\ 1$	$0 \cdots 0$	$0 \cdots 0$	
a_2	$\cdots$	$11 \cdots 1$	$0 \cdots 0$	
a_3	0 или 1	0 или 1	$11 \cdots 1$	$\cdots$
$\vdots$	0 или 1	0 или 1	0 или 1	
a_n	0 или 1	0 или 1	0 или 1	
c_j	$c_1\ c_2\ \cdots\ c_l$	$c_{l+1}\ \cdots$	$\cdots$	$\cdots$

Множества в блоке располагаются в порядке возрастания их весов и перенумеровываются так, что A_j соответствует j-му столбцу. Верхний индекс в A_j^p указывает номер блока.

Введем следующие обозначения.
$\hat{B}$ – текущее наилучшее решение, $\hat{z}$ – его вес;
B – текущее семейство $\mathcal{A}'$, z – его вес;
$E \subseteq A$ – подмножество тех элементов из A, которые покрываются подмножествами из B;
p – номер блока;
A_j^p – j-е подмножество, принадлежащее блоку p.

Алгоритм.

Шаг 1. Инициализация. Построить исходную таблицу и начать с пустого решения.

$B := \varnothing;\ E := \varnothing;\ z := 0;\ \hat{z}; = \infty;$

Шаг 2. Расширение решения.

Выбрать блок $p = \min[i|a_i \notin E]$

Пометить первое подмножество блока p (имеющего наименьший вес).

Шаг 3. Начиная с помеченной позиции блока, перебираем его подмножества A_j^p в порядке возрастания j.

1) Если нашли A_j^p такое, что

$A_j^p \cap E = \varnothing$ и $z + c_j < \hat{z}$

(где c_j – вес множества A_j^p)
то на Шаг 5
иначе (т.е. если блок исчерпан или выбрано подмножество A_j^p такое, что $z + c_j \geq \hat{z}$) на Шаг 4.

Шаг 4. Шаг возврата.

B не может привести к лучшему решению.
Если $B = \varnothing$ (т.е. блок p исчерпан), то алгоритм заканчивает работу; $\hat{B}$ – оптимальное решение;

стоп
иначе удалить из решения B последнее подмножество, включенное в B; пусть это последнее подмножество есть A_k^l; положить $p = l$;
если есть A_{k+1}^l, то удалить предшествующую метку над A_k^l;
пометить A_{k+1}^l;
на Шаг 3.
иначе на Шаг 4.

Шаг 5. Обновление решения и проверка.

$B := B \cup \{A_j^p\};\ E := E \cup A_j^p;\ z := z + c_j;$

Если найдено лучшее решение $E = A$, то положить

$\hat{B} := B;\ \hat{z} := z;$

на Шаг 4.

иначе на Шаг 2.

Пример 2.24.
$A = \{a, b, c, d, e\}$;
$\mathcal{A} = \{\{a, b, c\}$ [5], $\{a, c\}$ [3], $\{c, e\}$ [5], $\{b, e, d\}$ [3], $\{c, d\}$ [3], $\{a, d\}$ [2], $\{b\}$ [3]$\}$.

В квадратных скобках указаны веса соответствующих подмножеств.

	A_1^1	A_2^1	A_3^1	A_4^2	A_5^2	A_6^3	A_7^3
a	1	1	1	0	0	0	0
b	0	0	1	1	1	0	0
c	0	1	1	0	0	1	1
d	1	0	0	1	0	1	0
e	0	0	0	1	0	0	1
c_j	2	3	5	3	3	3	5

Итерации.

1. $p = 1$; $B = A_1^1$; $E = \{a, d\}$; $z = 2$; $E \neq A$; на Шаг 2.

2. $p = 2$; $A_5^2 \cap E = \varnothing$; $2+3 < \infty$; $B = A_1^1 \cup A_5^2$; $E = \{a, b, d\}$; $z = 5$; $E \neq A$; на Шаг 2.

3. $p = 3$; $A_7^3 \cap E = \varnothing$; $5 + 5 < \infty$; $z = 10$; $B = A_1^1 \cup A_5^2 \cup A_7^3$; $E = \{a, b, c, d, e\}$; $E = A$; $\hat{B} = \{A_1^1, A_5^2, A_7^3\}$; $\hat{z} = 10$;

Возврат.

4. $p = 1$; $B := A_2^1$; $E = \{a, c\}$; $z = 3 < 10$; $E \neq A$; на Шаг 2.

5. $p = 2$; $A_4^2 \cap E = \varnothing$; $B = A_2^1 \cup A_4^2$; $E = \{a, b, c, d, e\}$; $z = 3 + 3 = 6 < 10$;
$E = A$; $\hat{B} = \{A_2^1, A_4^2\}$; $\hat{z} = 6$;

Возврат

6. $p = 1$; $B := A_3^1$; $E = \{a, b, c\}$; $z = 5 < 6$; $E \neq A$; на Шаг 2.

7. Больше не находится A_j^l, такого, что $A_j^l \cap E = \varnothing$. На Шаг 4.

8. Удаляем из B последнее включенное в него множество, а именно A_3^1; $p := 1$; в блоке 1 больше нет подмножеств – на Шаг 4.

9. $B = \varnothing$; алгоритм заканчивает работу; $\hat{B} = \{A_2^1, A_4^2\}$, полученное на итерации 5, есть оптимальное решение;

стоп. ◄

Пример 2.25. Организации требуются переводчики с иностранных языков, приведенных в таблице. На должность переводчиков претендуют пять человек – A, B, C, D, E, каждый из которых знает несколько языков. Каких из переводчиков нужно принять на работу, чтобы обеспечить перевод со всех нужных языков и минимизировать суммарную зарплату (предполагается что оклады у всех переводчиков одинаковые)?

		A	B	C	D	E
1	Франц.	1	0	1	1	0
2	Нем.	1	1	0	0	0
3	Греч.	0	1	0	0	1
4	Итал.	1	0	0	1	0
5	Испан.	0	0	1	0	1
6	Рус.	0	1	1	0	1
7	Кит.	0	0	0	1	1

A	C	D	B	E
1	1	1	0	0
1	0	0	1	0
0	0	0	1	1
1	0	1	0	0
0	1	0	0	1
0	1	0	1	1
0	0	1	0	1

2.5 Отношения порядка

2.5.1 Строгий порядок

Определение. *Отношение $\langle A, M\rangle$ называется отношением* ***строгого порядка (строгим порядком)****, если оно антирефлексивно и транзитивно.*

Отношение строгого порядка будем также обозначать символами $\prec$ или $\succ$.

Примеры 2.26.

1. Отношения строгого порядка $<$ и $>$ для целых и действительных чисел.

2. Отношение включения $\subset$ есть отношение строгого порядка для множеств.
3. Отношения "быть предком"быть потомком" также являются примерами отношения строгого порядка. ◄

2.5.2 Свойства строгого порядка

Теорема 2.23 *Отношение строгого порядка асимметрично.*

Доказательство (от противного). Напомним, что свойство $A \cap A^{-1} = \varnothing$ является определением асимметричности.

Пусть $A \cap A^{-1} \neq \varnothing$, т.е. существуют такие x, y, что выполняются соотношения xAy и $xA^{-1}y$, или xAy и yAx. Из транзитивности отсюда следует: $xAy \ \wedge \ yAx \quad \Rightarrow \quad xAx$, что противоречит антирефлексивности. ☒

Итак, отношение строгого порядка A на M обладает следующими свойствами:
1. $\forall x \in M \ \ [\langle x, x\rangle \notin A]$.
2. $xAy \ \wedge yAz \quad \Rightarrow \ xAz$.
3. $xAy \quad \Rightarrow \ \langle y, x\rangle \notin A$ или $\langle x, y\rangle \in A \quad \Rightarrow \ \langle y, x\rangle \notin A$.

Первые два свойства – определение строгого порядка, а третье – следствие (теорема 2.23).

Граф, соответствующий отношению строгого порядка, не содержит ориентированных циклов (контуров). Ориентированный цикл (орцикл, контур) в ориентированном графе (орграфе) – это такая последовательность вершин (и дуг) $x_0, x_1, \ldots, x_k$, что $x_k = x_0$ и от вершины x_i к вершине x_{k+1} идет дуга. Частным случаем контура является петля.

Определение. *Множество M с заданным на нем отношением строгого порядка $\prec$ называется **упорядоченным** множеством и обозначается $\langle \prec, M\rangle$.*

Определение. *Отношение строгого порядка A называется **совершенным строгим порядком**, если для любой пары несовпадающих элементов $x \neq y$ из M верно либо xAy, либо yAx.*

Теорема 2.24 *Если на конечном непустом M задан совершенный строгий порядок $\prec$, то существует единственный* *элемент $x \in M$ такой, что для любого $y \in M$, не совпадающего с x ($y \neq x$), выполняется соотношение $x \prec y$.*

Элемент x, обладающий этим свойством называется *наименьшим* элементом в упорядоченном множестве $\langle \prec, M \rangle$.

Д о к а з а т е л ь с т в о . Возьмем любой $y_0 \in M$. Если y_0 – наименьший, то существование искомого элемента доказано. Если нет, то поскольку $\prec$ – совершенный строгий порядок, существует элемент $y_1 \neq y_0$ такой, что $y_1 \prec y_0$. Снова, либо y_1 наименьший, либо существует $y_2 \neq y_1$ такой, что $y_2 \prec y_1$.

Продолжим этот процесс. Пусть уже выбрано $n+1$ элементов, для которых

$$y_n \prec y_{n-1}, y_{n-1} \prec y_{n-2}, \dots, y_1 \prec y_0.$$

Из транзитивности следует, что $y_i \prec y_j$ при $i > j$. Из антирефлексивности следует, что выбранные элементы попарно не равны. Ввиду конечности M процесс выбора элементов должен оборваться на некотором конечном шаге. Элемент y_n, выбранный на последнем шаге будет, очевидно, искомым.

Итак, для любого $z \neq y_n$ выполнено $y_n \prec z$. Покажем, что этот элемент – единственный. Пусть существует другой элемент y'_n, такой, что для любого $z \neq y'_n$, $y'_n \prec z$. Тогда одновременно выполняются соотношения $y_n \prec y'_n$ и $y'_n \prec y_n$, что невозможно из-за асимметричности ⊠

З а м е ч а н и е. Если на M задан совершенный строгий порядок $\prec$, и $Q \subseteq M, Q \neq \varnothing$, то $\langle \prec, Q \rangle$ есть также совершенный строгий порядок и в Q существует единственный наименьший элемент. ▶

Следствие. *Пусть на M задано отношение совершенного строгого порядка. Тогда можно выбрать такую нумерацию элементов $M = \{x_1, \dots, x_n\}$, что соотношение $x_i \prec x_j$ будет выполняться, когда $i < j$.*

Д о к а з а т е л ь с т в о . Выберем x_1 – наименьший элемент множества M. Через M_1 обозначим $M \setminus \{x_1\}$. В M_1 снова найдем наименьший элемент x_2 и снова из M_1 удалим x_2, и так далее. Перебирая последовательно все элементы из M, получим цепочку

$$x_1 \prec x_2 \prec \ \cdots \ \prec x_n, \text{ где } x_i \prec x_j, \text{ если } i < j$$

(в силу асимметричности и транзитивности). ☒

Пусть на M задано отношение строгого порядка $\prec$.

Определение. *Элемент $x \in M$ называется минимальным (максимальным) в упорядоченном множестве $\langle \prec\ ,\ M \rangle$, если не существует никакого элемента $y \in M$, для которого $y \prec x$ (соответственно, $x \prec y$).*

На графе строгого порядка, если дугу проводить из x в y при $x \prec y$, то минимальный элемент соответствует вершине, в которую дуги не входят, а максимальный – вершине, из которой дуги не выходят.

В случае совершенного строгого порядка минимальный элемент x есть *наименьший* элемент, так как для любого $y \neq x$ $x \prec y$.

В случае просто строгого порядка может быть несколько минимальных (и максимальных) элементов, но они несравнимы (не находятся в рассматриваемом отношении порядка).

Пример 2.27. В графе строгого порядка, приведенном на рисунке, имеются три минимальных и два максимальных элемента.

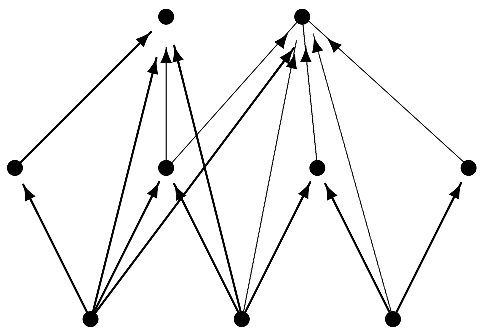


Рис. 2.13. Отношение строгого порядка

Определение. *Элементы $x, y \in M$ называются сравнимыми в данном упорядоченном множестве $\langle \prec, M \rangle$, если $x \prec y, x = y$,* *или $y \prec x$ (равенство в данном случае обозначает идентичность элементов, т.е. x и y – это один и тот же элемент).*

Определение. *Пусть на M задано отношение $\prec$ строгого порядка. Подмножество $Q \subseteq M$ называется максимальным совершенным, если*
1) отношение $\prec$ задает на Q совершенный строгий порядок;
2) на любом подмножестве $R \subseteq M, Q \subset R$ (для которого Q – собственное подмножество), отношение $\prec$ уже не является совершенным строгим порядком.

Теорема 2.25 *Пусть $\langle \prec, M \rangle$ – упорядоченное множество. Для любого элемента $y \in M$ существует максимальное совершенное подмножество $Q \subseteq M$, содержащее y.*

Доказательство (для конечного M). Пусть $Q_1 = \{y\}$. Отношение $\prec$ на Q_1 является совершенным строгим порядком (отношение пусто). Если Q_1 максимальное совершенное подмножество, то теорема доказана.

Пусть мы построили Q_n – совершенно строгий порядок с отношением $\prec$. Если Q_n максимально, то теорема доказана. Если нет, то в M существует элемент, сравнимый со всеми элементами из Q_n. Присоединим его к Q_n и получим $Q_{n+1} \supset Q_n$ с совершенным строгим порядком. Из-за конечности M этот процесс оборвется на конечном шаге и мы получим максимальное совершенное подмножество $Q \ni y$. $\boxtimes$

Теорема 2.26 *Пусть $\prec$ – отношение строгого порядка на конечном M. Для любого элемента $y \in M$ существует минимальный элемент $x \in M$, такой, что $x \prec y$ или $x = y$.*

Доказательство. Если y – минимальный элемент, то $x = y$. В противном случае существует такой элемент $z \in M$, что $z \prec y$. Если z – минимальный элемент, то $x = z$. В противном случае существует такой $u \in M$, что $u \prec z$, и так далее. Так как M конечное, то через конечное число шагов убывающая цепочка $\ldots u \prec z \prec y$ оборвется на искомом элементе. $\boxtimes$

Нестрогий порядок

Определение. *Отношение $\langle A, M\rangle$ называется отношением **нестрогого порядка** (**нестрогим порядком**), если его можно представить в виде*

$A = A_1 \cup \Delta$ *($\preccurlyeq = \prec \cup \Delta$),*

где A_1 – строгий порядок, Δ – диагональное отношение.

Нестрогий порядок часто обозначают как $\preccurlyeq$ или $\succcurlyeq$.

Из определения следует, что нестрогий порядок рефлексивен ($\Delta \subseteq A$).

Проверим его транзитивность ($A^2 \subseteq A$):

$(A_1 \cup \Delta)^2 = A_1^2 \cup A_1\Delta \cup \Delta A_1 \cup \Delta^2 \subseteq A_1 \cup A_1 \cup A_1 \cup \Delta = A_1 \cup \Delta$

(так как $\Delta A = A\Delta = A, \Delta^2 = \Delta$).

В отличие от строгого порядка $\prec$ отношение $\preceq$ не асимметрично, а только антисимметрично. Более того,

$A \cap A^{-1} = \Delta$. Действительно,

$A \cap A^{-1} = (A_1 \cup \Delta) \cap (A_1^{-1} \cup \Delta) =$

$= (A_1 \cap A_1^{-1}) \cup (A_1 \cap \Delta) \cup (A_1^{-1} \cap \Delta) \cup \Delta = \Delta$

(из антирефлексивности и асимметричности A_1).

Таким образом, отношение нестрогого порядка $\preceq$ рефлексивно, антисимметрично и транзитивно. Обратно, если отношение A рефлексивно, антисимметрично и транзитивно, то A – нестрогий порядок. Действительно,

$A = (A \setminus \Delta) \cup \Delta$, а $A \setminus \Delta$ – строгий порядок.

Итак, нестрогий порядок можно было определить аксиоматически как рефлексивное, антисиммтричное и транзитивное отношение. Ни одно из этих свойств не следует из других.

Определение. *Нестрогий порядок назовем **совершенным**, если для любой пары $\langle x, y\rangle$ выполняется одно из соотношений: $x \preccurlyeq y$, либо $y \preccurlyeq x$.*

Из антисимметричности нестрогого порядка следует, что одновременное выполнение $x \preccurlyeq$ и $y \preccurlyeq x$ означает совпадение $x = y$.

Теорема 2.27 *Если A – совершенный нестрогий порядок, то $A_1 = A \setminus \Delta$ – совершенный строгий порядок.*

Обратно, если A_1 – совершенный строгий порядок, то $A = A_1 \cup \Delta$ – совершенный нестрогий порядок.

Доказательство. 1. $\Rightarrow$ $A_1 = A \setminus \Delta = A \cap \overline{\Delta}$.

Покажем, что A_1 – совершенный строгий порядок.

1) A_1 антирефлексивно:

$\Delta \cap A_1 = \Delta \cap (A \cap \overline{\Delta}) = A \cap \Delta \cap \overline{\Delta} = \varnothing$.

2) A_1 асимметрично: $(A\cap\overline{\Delta})\cap(A\cap\overline{\Delta}^{-1} = A\cap\overline{\Delta}\cap A^{-1}\cap\overline{\Delta}^{-1} = $
$= A \cap A^{-1} \cap \overline{\Delta} \cap \Delta^{-1} \subseteq \Delta \cap \overline{\Delta} \cap \overline{\Delta}^{-1} = \varnothing$.

3) A_1 транзитивно:

$A_1 = (A_2 \cup \Delta) \cap \bar{\Delta} = (A_2 \cap \bar{\Delta}) \cup (\Delta \cap \bar{\Delta}) = A_2 \cap \bar{\Delta} = A_2$,
где A_2 – строгий порядок.

Все сравнимые в A элементы сравнимы и в A_2, т.е. A_1 – совершенный строгий порядок.

2. $\Leftarrow$ Пусть теперь A_1 – совершенный строгий порядок и $A = A_1 \cup \Delta$.

1) A рефлексивно : $\Delta \subseteq A$.

2) A антисимметрично:

$(A_1 \cup \Delta) \cap (A_1 \cup \Delta)^{-1} = (A_1 \cup \Delta) \cap (A_1^{-1} \cup \Delta^{-1}) =$
$= A_1 \cap A_1^{-1}) \cup (\Delta \cap A_1^{-1}) \cup (\Delta^{-1} \cap A_1) \cup (\Delta \cap \Delta^{-1}) = \Delta$
т.к. $A_1 \cap A_1^{-1} = \varnothing$; $\Delta \cap A_1^{-1} = \varnothing$; $\Delta^{-1} \cap A_1 = \varnothing$.

3) A транзитивно:

$(A_1 \cup \Delta)^2 = A_1^2 \cup A_1\Delta \cup \Delta A_1 \cup \Delta^2 = A_1 \cup A_1 \cup A_1 \cup \Delta = A_1 \cup \Delta$
(из рефлексивности и транзитивности A_1). ⊠

Квазипорядок является обобщением эквивалентности и нестрогого порядка одновременно.

Теорема 2.28 *Если отношение A есть одновременно эквивалентность и нестрогий порядок, то A – отношение равенства.*

Доказательство (от противного). Пусть выполнены соотношения xAy и $x \neq y$. Тогда из симметричности эквивалентности верно yAx. Из антисимметричности нестрогого порядка yAx не выполняется. Из противоречия (и рефлексивности) $y = x$. ⊠

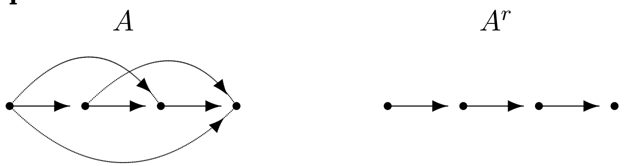
Редукция отношения

З а м е ч а н и е. Для любого из порядков A выполняется соотношение $A^2 \subseteq A$ в силу транзитивности A. ▶

Определение. ***Редукцией*** *отношения A называется отношение*

$$A^r = A \setminus A^2.$$

Пример 2.28.



Из определения A^r следует, что соотношение xA^ry выполняется тогда и только тогда, когда выполнено само соотношение xAy, но не существует такого "промежуточного" z, что xAz и zAy. Отношение xA^ry означает "непосредственное подчинение" элемента x элементу y (или непосредственное следование элемента x за элементом y).

Редукция отношения строгого порядка содержит всю информацию о нем, но изображается более простым графом.

З а м е ч а н и е. Для любого отношения A

1. $A^r \subseteq A$ – из определения A^r.
2. $(\hat{A})^r \subseteq A$: $(\hat{A})^r = \hat{A} \setminus \hat{A}^2 = A \cup A^2 \cup A^3 \cup \dots \cup A^n \cup \dots \setminus (A^2 \cup A^3 \cup \dots \cup A^n \cup \dots) \subseteq A$. ▶

Теорема 2.29 *Если A – строгий порядок на конечном M, то*

$$\widehat{A^r} = A$$

(транзитивное замыкание редукции совпадает с исходным порядком).

Д о к а з а т е л ь с т в о . 1. $A^r = A \setminus A^2 \;\Rightarrow\; A^r \subseteq A$

Далее, известно, $A \subseteq B \;\Rightarrow\; \hat{A} \subseteq \hat{B}$. Для транзитивного A: $\hat{A} = A$. Отсюда $\widehat{A^r} \subseteq \hat{A} = A$.

2. Докажем обратное включение $A \subseteq \widehat{A^r}$.

Пусть выполняется соотношение xAy. Если выполнены соотношения

$xAz_1, z_1Az_2, \ldots, z_{k-1}Az_k, z_kAy$, ($\star$)

то из-за антиантирефлексивности и транзитивности строгого порядка A и конечности M все элементы в цепочке $x, z_1, \ldots, z_k, y$ различны.

Рассмотрим все возможные цепочки элементов $z_1, \ldots, z_k$ ($k \geq 0$), такие, что выполняется ($\star$).

Так как M конечно, таких цепочек конечное число. Значит среди них есть цепочка наибольшей длины (возможно и не одна). Выберем любую из цепочек наибольшей длины. Из ($\star$) и из того, что цепочка $z_1, \ldots, z_k$ имеет наибольшую длину следует выполнение

$xA^rz_1, z_1A^rz_2, \ldots, z_{k-1}A^rz_k, z_kA^ry$ ($\star\star$)

Действительно, если, например, не выполняется $z_1A^rz_2$, то выполняется $z_1A^2z_2$, т.е. $\exists u\,[z_1Au \ \wedge \ uAz_2]$. Но тогда цепочка $z_1, u, z_2, \ldots, z_k$ имеет бо́льшую длину и обладает свойством ($\star$).

Из ($\star\star$) следует (по определению транзитивного замыкания) $x\widehat{A^r}y$, то есть $A \subseteq \widehat{A^r}$.

Получили, что из выполнения соотношения xAy следует $x\widehat{A^r}y$, т.е. $A \subseteq \widehat{A^r}$

Оба эти включения и определяют равенство. ⊠

Эта теорема означает для строгого порядка A: на конечном M по редукции можно однозначно восстановить отношение A. Более того, редукция A^r есть минимальное отношение, позволяющее восстановить A.

З а м е ч а н и е. Для нестрогого порядка $\preccurlyeq^2 = \preccurlyeq$, поэтому $\preccurlyeq^r = \preccurlyeq \setminus \preccurlyeq^2 = \varnothing$. Следовательно, о восстановлении $\preccurlyeq$ по редукции не может быть и речи. ▶

Напомним определение антитранзитивного отношения:

Определение. *Отношение B называется антитранзитивным, если*

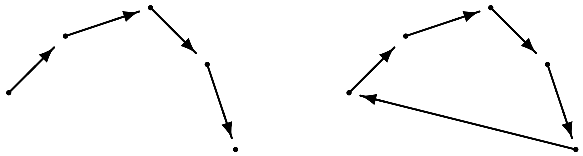
$B \cap B^n = \varnothing, \ n \geqslant 2$

Иначе говоря, если выполняется цепочка соотношений

$xBx_1, x_1Bx_2, \ldots, x_nBy$,

то невозможно выполнение xBy и наоборот.

Пример 2.29. На рисунке слева – антитранзитивное отношение, справа – не антитранзитивное, так как $B^{n+1} = B$ (n – число вершин).



◀

З а м е ч а н и е. Антитранзитивное отношение асимметрично и, следовательно, антирефлексивно. ▶

Теорема 2.30 *Если A – строгий порядок, то A^r антитранзитивно.*

Д о к а з а т е л ь с т в о (от противного). Пусть выполнено соотношение xA^ry. Предположим, что существует цепочка

$xA^rx_1, x_1A^rx_2, \ldots, x_nA^ry$.

Но тогда существует и цепочка ($A^r \subseteq A$)

$xAx_1, x_1Ax_2, \ldots, x_nAy$.

A – строгий порядок (A транзитивно), т.е. $xAx_1 \ \wedge \ x_1Ay \quad \Rightarrow \quad xA^2y$, т.е. $A^2 \subseteq A$ и $A^2 \subseteq A^r$. Но выполнение xA^ry невозможно, так как $A^r = A \setminus A^2$ (определение редукции).

Теорема доказана. ☒

Древесный порядок

Пусть задано упорядоченное множество $\langle \prec, M \rangle$ ($\prec$ – строгий порядок).

Определение. *Элемент $x_0 \in M$ называется наибольшим , если $\forall y \in M \ [y \neq x_0 \quad \Rightarrow \quad y \prec x_0]$ (если для любого элемента $y \in M$, отличного от x_0 выполнено соотношение $y \prec x_0$).*

Наибольший элемент, если он существует, единствен. Если строгий порядок на конечном множестве имеет единственный максимальный элемент, то этот элемент является наибольшим.

Определение. *Отношение строгого порядка $\prec$ на множестве M называется отношением древесного порядка (древесным порядком), если*
1) из того, что $x \prec y$ и $x \prec z$ следует, что y и z сравнимы (т.е. либо $y \prec z$, либо $z \prec y$).
2) на множестве $\langle \prec, M \rangle$ существует наибольший элемент.

Определим множество M_x следующим образом: M_x состоит из $x \in M$ и из всех y таких, что $y \prec x$ $M_x = \{x\} \cup \{y | y \prec x\}$.

Теорема 2.31 *Если A – древесный порядок на M, то на M_x отношение A также задает древесный порядок*

Доказательство. Первое условие определения древесного порядка выполняется для любого подмножества M. Наибольшим элементом в M_x является сам x. ☒

Теорема 2.32 *Если A – древесный порядок на конечном множестве M, то для всякого x, отличного от корня x_0, существует ровно один y, для которого выполнено $xA^r y$.*

Доказательство. 1. Предположим сначала, что существуют такие y и z ($y \neq z$), что $xA^r y$ и $xA^r z$. По определению древесного порядка, из выполнения xAy, xAz и $x \neq y$ следует yAz или zAy. Положим для определенности yAz. Получается, что выполнены соотношения xAy и yAz. Следовательно, невозможно $xA^r z$.

Итак, мы показали, что не может быть двух разных элементов, “непосредственно старших"чем данный x.

2. Предположим теперь, что для x не существует такого y, что $xA^r y$. Так как $A^r \subseteq A$, то из невыполнения $xA^r y$ следует и невыполнение xAy, т.е. не существует такого y, что xAy. Значит, $x = x_0$, т.е. x – максимальный и наибольший элемент. ☒

Это означает, что из каждой вершины графа редукции A^r исходит не более одной дуги, т.е. A^r является деревом: в нем не только нет орциклов, но и (неориентированных) циклов.

2.5.6 Некоторые свойства дерева

С помощью доказанных теорем можно показать, что граф, представляющий редукцию $\prec^r$ древесного порядка $\prec$ на конечном M действительно имеет древесную структуру.

(Растущее) дерево определяется следующим образом:

1) это граф без циклов;

2) из корня не исходит ни одной дуги;

3) из всех остальных вершин исходит ровно по одной дуге.

Рассмотрим свойства редукции строгого порядка.

1. Так как редукция строгого порядка антитранзитивна, то граф $\langle \prec^r, M \rangle$ не содержит циклов.
2. Так как корень – наибольший элемент, то из него не исходит ни одна дуга.
3. Из всех остальных вершин исходит ровно по одной дуге: по теореме (**??**) в древесном порядке для любого x, отличного от корня, существует единственный y, для которого выполняется $xA^r y$.
4. Любой подграф дерева – тоже дерево (поддерево).
5. Для несравнимых элементов упорядоченного множества x, y существует единственный минимальный элемент z, такой, что существуют пути из x в z и из y в z.

Замечание. Если в определении древесного порядка отказаться от существования наибольшего элемента, то для конечного M получим вместо дерева объединение нескольких попарно непересекающихся деревьев (лес). ▶

Вспомним определение совершенного строгого порядка: отношение A – строгий порядок и для любых x, y либо xAy, либо yAx.

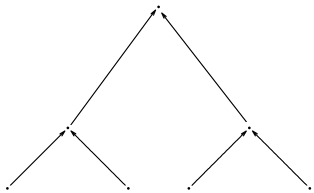
Граф редукции A^r отношения A совершенного строгого порядка обладает (кроме указанных) таким свойством: в любую вершину (кроме наименьшего элемента) заходит ровно одна дуга.

В отношении совершенного строгого порядка существуют единственный минимальный и единственный максимальный элементы, они же, соответственно, наименьший и наибольший.

Если A – совершенный строгий порядок, то его редукция A^r представляется в виде линейной цепочки. Поэтому совершенный строгий порядок называют также *линейным* порядком. Линейный порядок – частный случай древесного порядка. Иногда линейным порядком называют также и совершенный нестрогий порядок (главное условие – выполнение для любой пары x, y: либо $x \preceq y$, либо $y \preceq x$). Но для нестрогого порядка редукция пуста: $A \setminus A^2 = \varnothing$, так как для рефлексивного и транзитивного отношения $A^2 = A$.

Примеры 2.30.

1. Граф редукции древесного порядка.



2. Генеалогическое дерево.
3. Иерархические структуры:

- структуры данных;
- классы в биологии;
- классы в объектно-ориентированном программировании;
- бюрократические структуры (идеальные).

4. Линейный порядок:

- конечное подмножество натуральных чисел;
- словарный (лексикографический) порядок, если омонимы считать одним входом. ◄

З а м е ч а н и е. Пусть A – линейный порядок (строгий $\prec$ или нестрогий $\preccurlyeq$) на конечном M.

$S = \bigcup\limits_{i=1}^{m} M^i$ – множество i-кортежей над M с элементами вида:

$\langle a_1, a_2, \ldots, a_p \rangle$,

$\langle b_1, b_2, \ldots, b_q \rangle$, $p \leqslant q$

Отношение $\sqsubset$ на S называется *лексикографическим* порядком , т.е. для кортежей выполняется соотношение

$\langle a_1, a_2, \ldots, a_p \rangle \sqsubset \langle b_1, b_2, \ldots, b_q \rangle$,

если выполнено одно из условий:

1) существует такое целое j, что

$a_i = b_i$ при $i < j$, и $a_j A b_j$, (например, $a_j \prec b_j$).

2) $a_i = b_i$, $1 \leqslant i \leqslant p$ и $p \leqslant q$. ▶

2.5.7 Анализ отношений порядка

1. Проверка на строгий порядок.

Для того чтобы проверить, является ли данное отношение A строгим порядком, достаточно убедиться, что:

$\Delta \cap A = \varnothing$ – A антирефлексивно;

$A^2 \subseteq A$ – A транзитивно.

2. Проверка на линейный (совершенный строгий) порядок

Критерий 1. Пусть A – строгий порядок на конечном M, $|M| = n$.

A линейный порядок тогда и толькотогда, когда

$A^i \neq \varnothing,\ i < n,\ A^n = \varnothing.$

Д о к а з а т е л ь с т в о . Пусть A – линейный порядок.

1. (Базис индукции). Пусть $n = 2, i = 1$; выполнение $xAy \Rightarrow A^1 \neq \varnothing$. Пусть $A^2 \neq \varnothing$, т.е. выполняются соотношения $xAz \wedge zAy$. Так как A транзитивное и антирефлексивное, то x, y, z – разные. Этого не может быть, так как $n = 2$.

2. (Шаг индукции). Пусть утверждение справедливо для $n - 1$. Докажем его справедливость для n.

$A^{n-2} \neq \varnothing,\ A^{n-1} \neq \varnothing$ (по индукции).

Значит имеется цепочка

$z_1Az_2, z_2Az_3, \ldots, z_{n-2}Az_{n-1}, A^{n-1} \neq \varnothing$.

Добавление еще одной степени равносильно добавлению одной лишней вершины в графе. $A^n = \varnothing$, так как все z_i – разные.

Критерий 2. Пусть A – линейный порядок на конечном M, $|M| = n$. Образуем множества

$M_1 = \{y|yAx\},\ M_2 = \{y|xAy\}$,

где M_1 – множество элементов M, из которых дуги заходят в x, а M_2 – множество элементов M, в которые заходят дуги из x, и назовем $|M_1|$ степенью *захода* в x, а $|M_2|$ – степенью *исхода* из x.

Тогда для всех x, если

$|M_1| + |M_2| = |M| - 1$, то порядок линейный; в противном случае порядок не линейный.

Пример 2.31. Пример графа линейного порядка представлен на рисунке 2.14.

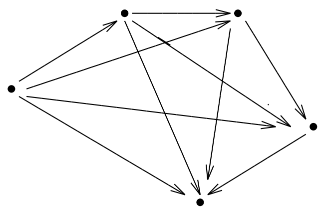


Рис. 2.14. Линейный порядок

Решетки

Пусть множество A упорядочено (частично) отношением $\preccurlyeq$ и пусть $\varnothing \subset B \subseteq A$.

Определение. *Элемент $a \in A$ называется верхней (нижней)* ***гранью*** *(границей) B тогда и только тогда, когда $b \preccurlyeq a$ $(a \preccurlyeq b)$ для всех $b \in B$.*

Наименьший (наибольший) элемент множества всех верхних (нижних) граней B называется точной верхней (нижней) гранью (границей) множества B и обозначается $\sup B$ $(\inf B)$.

Пример 2.32.

1. Пусть множество $\mathcal{B}(\{1,2,3\})$ упорядочено (частично) отношением $\subseteq$ и пусть $B_1 = \{\{1\},\{2\}\}$, $B_2 = \{\{1\},\{1,2\}\}$.

$\mathcal{B}(\{1,2,3\}) = \{\varnothing,\{1\},\{2\},\{3\},\{1,2\},\{1,3\},\{2,3\},\{1,2,3\}\}$.

Множество верхних граней для B_1 есть $\{\{1,2\},\{1,2,3\}\}$; $\sup B_1 = \{1,2\}$. Нижняя грань для B_1 есть $\varnothing$: $\inf B_1 = \varnothing$. Ни одна из граней не принадлежит B_1.

Для B_2 множество верхних граней есть $\{\{1,2\},\{1,2,3\}\}$, множество нижних граней — $\{\varnothing,\{1\}\}$; $\sup B_2 = \{1,2\}$, $\inf B_2 = \{1\}$.

2. Пусть множество $A = \{a,b,c\}$ упорядочено по отношению тождества $\equiv$. Единственные подмножества A, имеющие верхнюю и нижнюю грани есть $\{a\},\{b\},\{c\}$. Например, $\sup\{a\} = \inf\{a\} = a$. ◀

Определение. *Пусть $\preccurlyeq$ — некоторый порядок (частичный) на множестве M. Элемент $z \in M$ называется непосредственно следующим за элементом $x \in M$ тогда и только тогда, когда $x \preccurlyeq z$ и не существует такого $y \in M$, что $x \preccurlyeq y \preccurlyeq z$.*

Для строгого порядка $\prec$, например, непосредственно следующим за x является элемент z из редукции $x \prec^r z$.

Диаграммы Хассе

Пусть задано упорядоченное множество $\langle \preccurlyeq, M\rangle$. Представим каждый элемент $x \in M$ точкой на плоскости. Рассмотрим все упорядоченные пары $\langle x_i, x_j\rangle \in M \times M$. Поместим точку x_j *над* точкой x_i только тогда, когда $x_i \preccurlyeq x_j$ и соединим точки x_i, x_j линией, если x_j непосредственно следует за x_i.

В результате получится диаграмма, в которой существует маршрут от точки x_m к точке x_n, если $x_m \preccurlyeq x_n$. Такие диаграммы по имени их автора называют диаграммами Хассе.

Примеры 2.33.

1. Для упорядоченного множества $\langle \leqslant, A\rangle$, где $A = \{1,2,3,4\}$, диаграмма Хассе представлена на рис. 2.15, *а*.

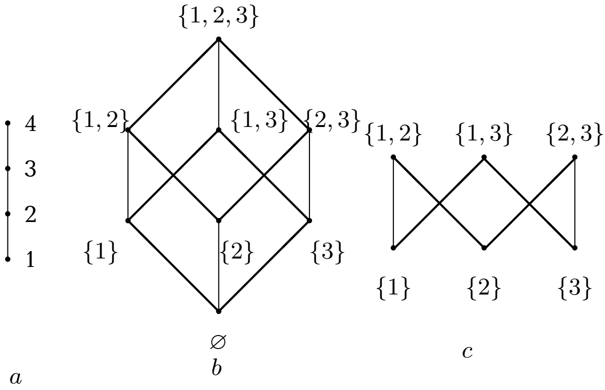


Рис. 2.15. Диаграммы Хассе

2. Семейство

$$\mathcal{B}(\{1,2,3\}) = \{\varnothing, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1,2,3\}\}$$

упорядочено отношением включения $\subseteq$. Диаграмма Хассе представлена на рис. 2.15, *b*.
3. Семейство множеств $\mathcal{A} = \{\{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}\}$ упорядочено отношением включения $\subseteq$. Диаграмма Хассе для этого упорядоченного множества представлена на рис. 2.15, *c*. ◀

Любая точка, из которой нет восходящей линии – это один из максимальных элементов множества, и если такая точка только одна, она представляет наибольший элемент множества.

Точки, из которых не исходят нисходящие линии, представляют минимальные элементы (если он один, то наименьший).Наибольший (наименьший) элемент представляет точную верхнюю (нижнюю) грань этого множества.

Решетка

Пусть $\langle \preccurlyeq, M \rangle$ – частично упорядоченное множество.

Определение. $\langle \preccurlyeq, M \rangle$ *является* ***решеткой*** *тогда и только тогда, когда каждая пара элементов* $a, b \in M$ *имеет точную верхнюю и точную нижнюю границы.*

Обозначим $\sup\{a,b\} = a \vee b, \inf\{a,b\} = a \wedge b$. Тогда определение решетки можно записать так:

$$\forall a, b \in M [\exists! i = a \wedge b, i \in M \ \& \ \exists! s = a \vee b, s \in M]$$

где $\exists!$ означает "существует один и только один".

Операции $\wedge$ и $\vee$ можно также рассматривать как отображения $M \times M$ в M, которые каждой паре $\langle a, b \rangle$ $(a, b \in M)$ ставят в соответствие элементы $a \vee b$, $a \wedge b$.

Пусть $a, b, c \in M$. Можно доказать, что решетка обладает следующими свойствами.

1. $a \vee b = b \vee a$; $a \wedge b = b \wedge a$ (коммутативность);
2. $a \vee (b \vee c) = (a \vee b) \vee c$; $a \wedge (b \wedge c) = (a \wedge b) \wedge c$ (ассоциативность);
3. $a \vee a = a$; $a \wedge a = a$ (идемпотентность);
4. $a \vee (a \wedge b) = a$; $a \wedge (a \vee b) = a$ (поглощение);

Определение. *Решетка* $\langle \preccurlyeq, M \rangle$ ***дистрибутивна***, *если для всех* $a, b, c \in M$

$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$;

$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$.

Примеры 2.34.

1. Нетрудно проверить, что решетки на рис. 2.16. дистрибутивны:

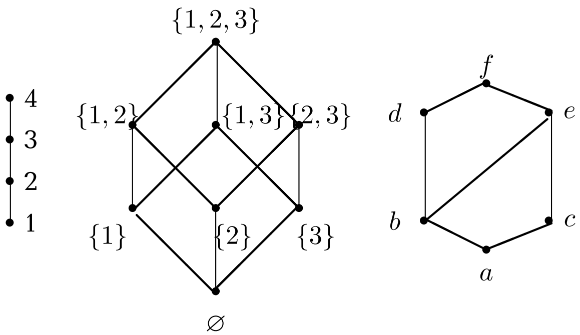


Рис. 2.16. Дистрибутивные решетки

2. Решетка $\langle \subseteq, \{\varnothing, \{1\}, \{2\}, \{3\}, \{1, 2, 3\}\} \rangle$ (рис. 2.17.) недистрибутивна. Действительно,

$\{1\} \vee (\{2\} \wedge \{3\}) = \{1\} \vee \varnothing = \{1\}$;

$(\{1\} \vee \{2\}) \wedge (\{1\} \vee \{3\}) = \{1, 2, 3\} \vee \{1, 2, 3\} = \{1, 2, 3\}$.

Заметим, что для этой решетки операции $\vee, \wedge$ не могут быть интерпретированы как теоретико-множественные операции $\cup$ и $\cap$.

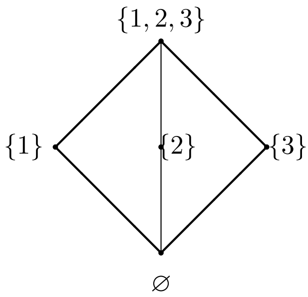


Рис. 2.17. Недистрибутивная решетка

3. Линейно упорядоченное множество является решеткой (дистрибутивной).
4. Контрпримеры: понятие отношения порядка не равносильно понятию решетки:

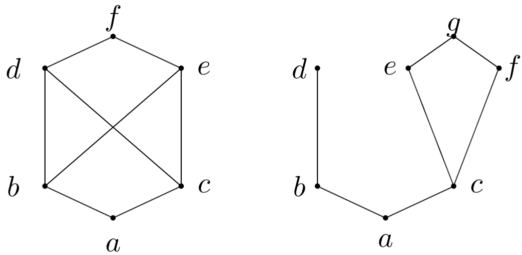


Рис. 2.18. Эти множества не являются решетками

На левой диаграмме пара элементов $\langle d, e\rangle$ имеет не единственную нижнюю грань: $d \wedge e = \{b, c\}$. На правой диаграмме пара элементов $\langle d, e\rangle$ не имеет верхней грани. ◀

Определение. Если $\langle \preccurlyeq, M\rangle$ – решетка и M имеет наименьший и наибольший элементы, то решетка называется ***ограниченной***. Обозначим через $\mathbf{1}$ верхнюю грань всей решетки, а через $\mathbf{0}$ – нижнюю грань всей решетки.

Определение. *Ограниченная решетка* $\langle \preccurlyeq, M\rangle$ *называется* ***решеткой с дополнениями***, *если для каждого* $a \in M$ *существует такой элемент* $\bar{a} \in M$, что $a \vee \bar{a} = \mathbf{1}$ и $a \wedge \bar{a} = \mathbf{0}$.

Элемент $\bar{a}$ называется ***дополнением*** элемента a.

Пример 2.35. Для решетки, приведенной на рисунке 2.19:

$\bar{\mathbf{0}} = \mathbf{1}$, $\bar{\mathbf{1}} = \mathbf{0}$, $\bar{a} = d$, $\bar{b} = d$, $\bar{c} = d$, $\bar{d} = a$, $\bar{d} = c$.

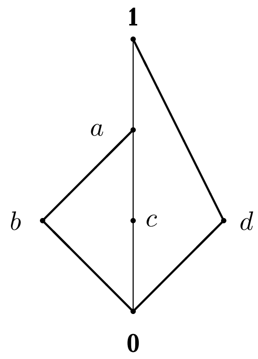


Рис. 2.19. Ограниченная решетка

Определение. *Решетка* $\langle \preccurlyeq, M \rangle$ *называется* ***решеткой с единственным дополнением****, если для каждого элемента* $a \in M$ *существует только одно дополнение.*
Примером решетки с единственным дополнением может служить

$\langle \subseteq, \mathcal{B}(\{1, 2, 3\}) \rangle$.

Булева решетка

Определение. ***Булева решетка*** *– это дистрибутивная решетка с единственным дополнением.*
(выполняются законы двойного дополнения и де Моргана: $\bar{\bar{a}} = a$, $\overline{a \vee b} = \bar{a} \wedge \bar{b}$, $\overline{a \wedge b} = \bar{a} \vee \bar{b}$).

Конечная булева решетка и решетка $\langle \subseteq, \mathcal{B}(U) \rangle$ изоморфны.

Запишем булеву решетку в виде

$\langle \boldsymbol{B},\ +,\ \cdot,\ \bar{},\ \preccurlyeq,\ \mathbf{0},\ \mathbf{1} \rangle$,

где $\boldsymbol{B} = \{a, b, c, \ldots\}$ – некоторое (конечное или бесконечное) множество элементов, в котором выделены особые элементы $\mathbf{0}$ и $\mathbf{1}$. Операции $+$, $\cdot$, $\bar{}$ удовлетворяют аксиомам булевой алгебры (включая законы де Моргана и законы поглощения $a + ab = a,\ a(a + b) = a$).

Отношение порядка $\preccurlyeq$ – нестрогое (рефлексивное, антисимметричное и транзитивное) связано с операциями следующим образом:

1. $a \preccurlyeq a + b$;
2. $ab \preccurlyeq a$;
3. $b \preccurlyeq a \quad \Rightarrow \quad a + b = a,\ ab = b$;
4. $b \preccurlyeq a \ \&\ c \preccurlyeq a \quad \Rightarrow \quad b + c \preccurlyeq a$,

$a \preccurlyeq b \ \&\ a \preccurlyeq c \quad \Rightarrow \quad a \preccurlyeq bc$;

5. $b \preccurlyeq a \quad \Rightarrow \quad b + c \preccurlyeq a + c, bc \preccurlyeq ac$;
6. $a + b = \max[a, b]$, $ab = \min[a, b]$ (если a и b сравнимы);
7. $b \preccurlyeq a \quad \Rightarrow \quad \bar{a} \preccurlyeq \bar{b}$.

Векторная решетка

Рассмотрим два m-кортежа

$$k \;=\; \langle a_1, a_2, \ldots, a_m \rangle,$$

$$l \;=\; \langle b_1, b_2, \ldots, b_m \rangle,$$

в которых a_i и b_i, $i = \overline{1, m}$ принадлежат одному и тому же совершенному порядку (вполне упорядоченному множеству), строгому или нестрогому. Отношение порядка на M обозначим символом $\geqslant$.

Будем говорить, что k доминирует l ($k \succcurlyeq l$) тогда и только тогда, когда

$$a_1 \geqslant b_1,\ a_2 \geqslant b_2,\ \ldots, a_m \geqslant b_m.$$

Для строгого порядка обозначения, соответственно, $>$ и $\succ$.

В этом случае будем говорить, что $k \succ l$ (k строго доминирует l), если

$$a_1 \geqslant b_1, \ldots, a_i > b_i, \ldots, a_m \geqslant b_m,$$

т.е. имеется по крайней мере один a_i и один b_i, для которых выполняется строгое соотношение $>$.

Говорят, что отношение доминирования индуцирует отношение порядка (совершенное или частичное) между кортежами k и l.

Пример 2.36.

$k = \langle 7, 3, 0, 5 \rangle$, $l = \langle 2, 2, 0, 4 \rangle$, $p = \langle 3, 4, 1, 4 \rangle$;

$k \succ l,\ p \succ l$, k и p – несравнимы. ◀

Пусть $M_1, M_2, \ldots, M_n$ множества, каждое из которых вполне упорядочено отношением $\prec$. Декартово произведение $M_1 \times M_2 \times \cdots \times M_n$ упорядочено и образует решетку, которая называется ***векторной*** решеткой. Отношение порядка на ней – это отношение ***доминирования***.

З а м е ч а н и е. За исключением булевой векторной решетки, каждая векторная решетка дистрибутивна, но не имеет дополнений. ▶

2.5.13 Нормированные булевы решетки

Булева решетка называется ***нормированной***, если каждому элементу $a \in \boldsymbol{B}$ сопоставляется неотрицательное число $\|a\|$ – ***норма*** элемента a, причем:

$$\left.\begin{array}{l} 1.\ \mathbf{0}\ \leq\ \|a\|\ \leq\ \mathbf{1},\ \|\mathbf{0}\| = 0,\ \|\mathbf{1}\| = 1; \\ 2.\ \text{Если } ab = \mathbf{0}, \text{то } \|a+b\| = \|a\| + \|b\|; \end{array}\right\} \text{ – аксиомы нормы,}$$

($a, b \in \boldsymbol{B}$ – произвольные элементы).

Теперь обозначим *нормированную булеву решетку*

$\langle \mathbf{B},\ +,\ \cdot,\ ^{-},\ \preccurlyeq,\ \|\|,\ \mathbf{0},\ \mathbf{1} \rangle$,

в которой к уже введенным ранее операциям добавляется *унарная* операция нормировки $\|\|$, $|\ a\| \in \mathcal{R}$, где $\mathcal{R}$ – можество действительных чисел.

2.5.14 Модели нормированной булевой решетки

1. Пусть булева решетка – это алгебра множеств и пусть U – конечное универсальное множество, $|U| = n$. Будем считать, что норма каждого $A \subseteq U$ задается числом элементов, входящих в A. Для выполнения первой аксиомы нормировки $\|U\| = 1$ будем делить $|A|/|U|$. Так, если $|A| = k$, то норма будет $\|A\| = \dfrac{k}{n}$.

Аксиома Н2 имеет совершенно ясный смысл: если множества A и B не пересекаются, т.е. $A \cap B = \varnothing$, то

$$\|A \cup B\| = \frac{|A \cup B|}{|U|} = \frac{|A| + |B|}{|U|} = \frac{|A|}{|U|} + \frac{|B|}{|U|} = \|A\| + \|B\|,$$

$\|U\| = 1.$

Такое определение нормы $\|A\|, A \subseteq U$ обобщается следующим образом. Пусть элементы $a_1, a_2, \ldots, a_n \in U$ имеют разные "веса' или "стоимости"и "вес"("цена") задается неотрицательными числами $w_1, w_2, \ldots, w_n$. Веса выбираются таким образом, чтобы

$w_1 + w_2 + \cdots + w_n = 1$

Тогда, для $A = \{a_{i_1}, a_{i_2}, \ldots, a_{i_k}\}$

$\|A\| = \|\{a_{i_1}, a_{i_2}, \ldots, a_{i_k}\}\| = w_{i_1} + w_{i_2} + \cdots + w_{i_k}.$

Если положить $w_1 = w_2 = \cdots = w_n = \dfrac{1}{n}$, то получим предыдущее определение нормы.

2. Элементы $0, 1$ решетки

$\langle\{\mathbf{0}, \mathbf{1}\}, +, \cdot, \bar{}, \preccurlyeq, \|\|, 0, 1\rangle$

уже как обычные числа можно принять за нормы соответствующих элементов:

$\|\mathbf{0}\| = 0, \ \|\mathbf{1}\| = 1.$

При этом первая аксиома нормы выполняется.

Так как

$\mathbf{0} \cdot \mathbf{0} = \mathbf{0}$ и $\|\mathbf{0}+\mathbf{0}\| = 0 = \|\mathbf{0}\| + \|\mathbf{0}\|$; $\mathbf{0} \cdot \mathbf{1} = \mathbf{0}$ и $\|\mathbf{0}+\mathbf{1}\| = 1 = \|\mathbf{0}\| + \|\mathbf{1}\|$,

то выполнена и вторая аксиома нормы.

Такое определение нормы превращает двухэлементную булеву решетку в нормированную.

Эта модель нормированной решетки используется, в частности, в исчислении высказываний и алгебре контактных схем.

3. **Вероятности**. Пусть U – некоторое конечное множество элементарных событий. Например, A – событие типа: монета упала гербом вверх, завтра пойдет дождь, эта деталь бракованная.

Определим операции над событиями:

объединение событий $A \cup B$ – произошло хотя бы одно из событий A или B;

пересечение $A \cap B$ событий – произошли оба события;

дополнение $\overline{A}$ события A – произошло событие "не A".

Особые элементы алгебры событий: достоверное событие U и невозможное событие Ω. Примеры: U – монета упала кверху "орлом"или кверху "решкой"; Ω – монета упала кверху "орлом"и кверху "решкой".

Предположим, что алгебра событий нормирована с соблюдением аксиом нормировки (алгебра событий – булева).

Норму $\|A\|$ элемента A рассмтриваемой булевой решетки обозначим через $p(A)$, а аксиомы нормы запишем в виде:

B1: $0 \leq p(A) \leq 1, p(U) = 1, p(\Omega) = 0$;

B2: если $A \cap B = \Omega$, то $p(A \cup B) = p(A) + p(B)$

(основные аксиомы теории вероятностей).

$p(A) = 1$, если событие A достоверно и $p(A) = 0$, если событие A невозможно. Число $p(A)$ характеризует вероятность того, что событие A произойдет. Таким образом, объектом теории вероятностей является совокупность элементов, образующих нормированную булеву решетку.

Глава 3

Булевы функции (БФ)

3.1 Определения и примеры

Булевы функции

Булевы функции (функции алгебры логики) определяются на множестве наборов (векторов), состоящих из нулей и единиц, и принимают значение 0 или 1.

Определение. ***Булевы константы*** *0 и 1 образуют* ***булево множество*** $\boldsymbol{B} = \{0, 1\}$. ***Булева переменная*** *– переменная, которая может принимать значение из* $\boldsymbol{B}$.

Булева функция (БФ) n ***переменных*** *– это отображение*

$$f : \boldsymbol{B}^n \mapsto B.$$

Булеву функцию можно представить в виде

$$y = f(x_1, x_2, \ldots, x_n),$$

в котором каждая булева переменная x_i и функция f могут принимать одно из двух возможных значений 0 или 1. Значение функции соответствует набору значений переменных $\langle a_1 \cdots, a_n \rangle \in \boldsymbol{B}^n$, где a_i есть фиксированное значение переменной x_i, $a_i \in \{0, 1\}$,

$$f(a_1, \cdots, a_n).$$

Булев вектор (набор) $\langle a_1 \cdots, a_n \rangle \in \boldsymbol{B}^n$ будем изображать в виде последовательности значений булевых переменных (***компонент*** вектора) без скобок и запятых (для сокращения записи) $a_1 a_2 \cdots a_n$. ***Длина*** вектора – число его компонент, а ***вес*** вектора – число единиц в нем.

Пример 3.1. Длина вектора $\vec{a} = a_1 a_2 a_3 a_4 a_5 = 10101$ равна 5, а его вес – 3. ◀

Теорема 3.1 $|\boldsymbol{B}^n| = 2^n$ *(число различных векторов длины n равно 2^n).*

Д о к а з а т е л ь с т в о .

Векторы упорядочены отношением ***доминирования*** $\preccurlyeq$: для векторов $\vec{a} = a_1 \ldots a_n$, $\vec{b} = b_1 \ldots b_n$,

$$\vec{a} \preccurlyeq \vec{b} \Leftrightarrow \forall i \ a_i \leqslant b_i \ (0 < 1).$$

Доминирование строгое ($\vec{a} \prec \vec{b}$), если хотя бы для одного i $a_i < b_i$. Если для векторов $\vec{a}$, $\vec{b}$ не выполняются ни $\vec{a} \preccurlyeq \vec{b}$, ни $\vec{b} \preccurlyeq \vec{a}$, то векторы ***несравнимы***.

Пример 3.2.

$$\begin{aligned} \vec{a} &= 1011, \\ \vec{b} &= 1001, \\ \vec{c} &= 1101, \end{aligned}$$

$\vec{b} \prec \vec{a}$, $\vec{b} \prec \vec{c}$, $\vec{a}$ и $\vec{c}$ несравнимы. ◀

Векторы разных размерностей нельзя сравнивать по отношению доминирования.

Булев вектор можно рассматривать как двоичное представление целого неотрицательного десятичного числа. Если компоненты вектора $\vec{a} = a_1 \ldots a_n$ интерпретировать как числа 0 и 1, то десятичное число $a \in \{0, 1, \ldots, 2^n - 1\}$ есть

$$a = \sum_{i=1}^{n} a_i \cdot 2^{n-i}.$$

Например, вектор $\vec{a} = 1011$ представляет число

$1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 8 + 0 + 2 + 1 = 11.$

В десятичном представлении векторы упорядочены в линейном порядке.

Множество всех булевых функций n переменных обозначим

$$P_n = \{f|f : \{0,1\}^n \mapsto \{0,1\}\}.$$

Теорема 3.2

$$|P_n| = 2^{2^n}.$$

Доказательство. Число булевых векторов длины n (число n-кортежей над двухэлементным множеством $\{0,1\}$) равно 2^n. В свою очередь, $|P_n|$ есть число 2^n-кортежей над $\{0,1\}$, то есть $|P_n| = 2^{2^n}$.

Табличное представление БФ

Булеву функцию можно представить таблицей с 2^n строками и $n+1$ столбцами. В первых n столбцах перечисляются наборы значений аргументов $a_1 \dots a_n$ в лексикографическом порядке (по возрастанию чисел, представленных векторами, от 0 до 2^{n-1}), а в $(n+1)$-м столбце – значения функции для этих аргументов.

Таблица 3.1. Табличное представление БФ

	x_1	$\cdots$	x_n	$f(x_1, \dots, x_n)$
0	0	$\dots$	0	$f(0, \dots, 0)$
1	0	$\dots$	1	$f(0, \dots, 1)$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$2^n - 1$	1	$\dots$	1	$f(1, \dots, 1)$

Слева в таблице добавлен столбец с десятичными номерами векторов.

Если интерпретировать элементы множества $\{0,1\}$ как истинностные значения "ложь" и "истина" некоторых высказываний, то булеву функцию можно считать некоторым сложным высказыванием. При этом истинностное значение f зависит от истинностных значений переменных.

С другой стороны, булеву функцию можно считать n-арной операцией на множестве $\{0, 1\}$. Таким образом, понятие логических операций $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ является частным случаем общего понятия логической операции. Таблицу, представляющую булеву функцию, можно считать таблицей истинности этой операции.

Примеры 3.3. 1. При $n = 0$ булевых функций две ($2^{2^0} = 2^1$): константа 0 и константа 1.

2. Булевы функции одной переменной.

При $n = 1$ число булевых функций равно 4 ($2^{2^1} = 2^2$):

Таблица 3.2. Булевы функции одной переменной

x	$f_0(x)$	$f_1(x)$	$f_2(x)$	$f_3(x)$
0	0	0	1	1
1	0	1	0	1

Здесь в одной таблице представлены четыре функции одной переменной. Они имеют собственные названия:

$f_0(x)$ – 0, нуль;

$f_1(x)$ – тождественная функция;

$f_2(x)$ – $\neg\, x$,отрицание, инверсия (другие обозначения: $\bar{x}, \sim x$);

$f_3(x)$ – 1, единица.

3. Булевы функции двух переменных.

При $n = 2$ число булевых функций равно 16 ($2^{2^2} = 2^4$):

Таблица 3.3. Булевы функции двух переменных

x_1	x_2	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Обозначения и названия некоторых функций двух переменных:

f_0 – 0, нуль;

f_1 – $x_1 \wedge x_2$, конъюнкция (другие обозначения: $\&, \cdot$);

f_6 – $x_1 \oplus x_2$, исключающее "или", сложение по модулю 2;

f_7 – $x_1 \vee x_2$, дизъюнкция, (другое обозначение $+$);

f_8 – $x_1 \downarrow x_2$, стрелка Пирса;

f_9 – $x_1 \leftrightarrow x_2$, эквиваленция (другие обозначения: $\sim, \equiv$);
f_{13} – $x_1 \to x_2$, импликация (другие обозначения: $\supset, \Rightarrow$);
f_{14} – $x_1|x_2$, штрих Шеффера;
f_{15} – 1, единица.

4. Мажоритарная функция (функция голосования, принимает значение 1, если "за" проголосовало большинство).

Таблица 3.4. Мажоритарная функция

	x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Булевы функции с числом переменных $n \leqslant 2$ называются ***элементарными***.

Геометрическое представление БФ

Область определения БФ n переменных множество $\boldsymbol{B}^n = \{0, 1\}^n$ можно интерпретировать геометрически как булев n-мерный куб (гиперкуб). ***Вершинами*** гиперкуба являются n-мерные булевы векторы, упорядоченные отношением доминирования. Гиперкуб, как упорядоченное множество, можно представить с помощью диаграммы Хассе. Булевой функции ставится в соответствие то подмножество $A \subseteq \boldsymbol{B}^n$ вершин куба, на которых функция равна 1:

$$f_A(x_1, \ldots, x_n) = \begin{cases} 1, & \{x_1, \ldots, x_n\} \in A \\ 0, & \text{в противном случае} \end{cases}$$

Пример 3.4.

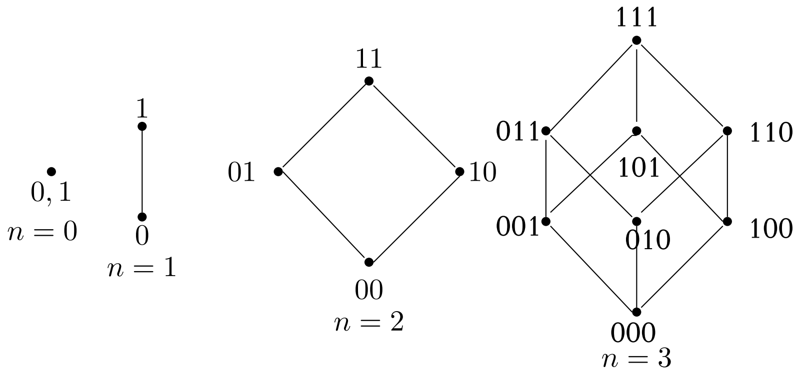


Рис. 3.1. Булевы гиперкубы.

Мажоритарной функции соответствует множество $\{011, 101, 110, 111\}$ на трехмерном кубе, сложению по модулю 2 – множество $\{01, 10\}$ на двухмерном кубе. Функции 0 соответствует пустое множество $\varnothing$, а функции 1 – множество всех вершин гиперкуба $\boldsymbol{B}^n$. ◀

Представление вектором значений. Так как множество векторов упорядочено от 0 до $2^n - 1$, то для представления функции достаточно задать значения функции в этом порядке.

Пример 3.5. Мажоритарная функция трех переменных задается вектором 00010111, сложение по модулю 2 (функция двух переменных) – вектором 0110.

Можно также перечислить десятичные номера тех векторов, на которых булева функция принимает значение 1. Для мажоритарной функции трех переменных $f = \{3, 5, 6, 7\}$, для сложения по модулю 2 (функции двух переменных) – $f = \{1, 2\}$. ◀

Представление БФ формулами

Определение. ***Суперпозицией*** *булевых функций f_0 и $f_1, \ldots, f_m$ называется функция*

$$f(x_1, \ldots, x_n) = f_0(f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n)).$$

Выберем некоторое подмножество функций n переменных $x_1, \ldots, x_n$

$$\Phi = \{f_1, \ldots, f_m\}, \quad \Phi \subseteq P_n,$$
$$P_n = \{f(x_1, \ldots, x_n) | f : \{0,1\}^n \mapsto \{0,1\}\}.$$

Множество Φ называется ***базисом***.

Определение. ***Формулой над базисом*** $\Phi = \{f_1, \ldots, f_m\}$ *называется суперпозиция* $F = f(f_1, \ldots, f_m)$, *где* $f \in \Phi$.

f называется ***главной*** (внешней) функцией (операцией), а f_i – ***подформулой***. Всякой формуле F однозначно соответствует некоторая булева функция f. Говорят, что формула F ***реализует*** функцию f и обозначают F_f.

Формулы над множеством элементарных функций называют просто *формулами*.

Пример 3.6. Рассмотрим базис $\Phi = \{\,^-, \wedge, \vee\}$. Это множество, состоящее из отрицания, конъюнкции и дизъюнкции называется ***стандартным*** базисом. Формулой над стандартным базисом будет любая переменная $x, y, x_1, x_2, \ldots, x_n$. Далее, из переменных x, y можно построить новую формулу $\vee(x, y)$ или $\wedge(x, y)$. Для записи этих формул чаще используют инфиксную нотацию $(x \vee y), (x \wedge y)$. Отрицание записывают в виде $(\bar{x})$, а не $^-(x)$.

В формулах над стандартным базисом используют ассоциативность операций $\wedge$ и $\vee$, используют соглашение о приоритете операций $(\,^-, \wedge, \vee)$ и опускают внешние скобки. Например, вместо $(^-(x \vee y) \vee ((y \wedge z) \wedge v)$ записывают $\overline{(x \vee y)} \vee y \wedge z \wedge v$. Знак конъюнкции часто заменяют на $\cdot$ или вовсе опускают, знак дизъюнкции заменяют на $+$. Тогда наша формула примет вид $\overline{(x + y)} + yzv$. Порядок выполнения операций определяется приоритетом операций и расстановкой скобок.

Построим таблицу для функции f, которую реализует формула $\bar{x} + xy + \bar{x}y$.

x	y	f
0	0	1
0	1	1
1	0	0
1	1	1

Полученная функция есть ни что иное как f_{13} из таблицы функций двух переменных – импликация. ◄

3.2 Равенство булевых функций

Определение. *Булевы функции* ***равны***

$$f_1(x_1, \ldots, x_n) = f_2(x_1, \ldots, x_n),$$

если на одинаковых наборах значений переменных они принимают одинаковые значения.

Фиктивные переменные

Если функции зависят от разного числа переменных, то об их равенстве или неравенстве нельзя судить без дополнительных соглашений.

Определение. *Переменная x_i называется* ***фиктивной*** *для булевой функции, если*

$$f(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) = f(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$$

при любых наборах значений остальных переменных.

Переменная x, не являющаяся фиктивной называется ***существенной***, при этом функция f существенно зависит от x.

Примеры 3.7. 1. Для функций одной переменной 0 и 1 переменная x является фиктивной.

2. Для функций двух переменных:

Таблица 3.5. Фиктивные переменные

x_1 x_2	0011 0101	Фик- тивные
f_0	0000	x_1, x_2
f_3	0011	x_2
f_5	0101	x_1
f_{10}	1010	x_1
f_{12}	1100	x_2
f_{15}	1111	x_1, x_2

Фиктивную переменную x_i можно удалить из таблицы следующим образом: вычеркнуть столбец x_i и все строки, в которых $x_i = 1$ (или все строки, в которых $x_i = 0$.)

Пример 3.8. Для $f_{12}(x_1, x_2)$ (x_2 – фиктивная): вычеркнув столбец x_2 и строки 1 и 3 из таблицы

	x_1	x_2	f_{12}
0	0	0	1
1	0	1	1
2	1	0	0
3	1	1	0

получим

	x_1	f
0	0	1
1	1	0

.

Полученная функция одной переменной – отрицание. ◀

Теперь можно уточнить определение равенства булевых функций.

Определение. *Булевы функции $f_1(x_1, \ldots, x_n)$ и $f_2(y_1, \ldots, y_m)$ равны (с точностью до фиктивных переменных), если f_2 может быть получена из f_1 введением или удалением фиктивных переменных.*

Таким образом, можно сравнивать функции с разным числом переменных. В дальнейшем булевы функции рассматриваются с точностью до фиктивных переменных. Это позволяет считать, что все булевы функции (в данной системе функций) зависят от одних и тех же переменных.

Уточнение. *Булевы функции f_1 и f_2 равны, если их существенные переменные равны и на каждом наборе значений переменных функции f_1 и f_2 принимают равные значения.*

Равносильные формулы

Одна и та же булева функция над данным базисом Φ может иметь множество реализаций формулами. Для сравнения формул вводится понятие ***равносильности***.

Определение. *Формулы F_1 и F_2, реализующие равные булевы функции, называются **равносильными** $F_1 = F_2$.*

Равносильность формул можно доказать с помощью построения таблиц функций, реализованных этими формулами.

Пример 3.9. Доказать равносильность $x \vee yz = (x \vee y)(x \vee z)$.

Таблица 3.6. Равносильные формулы

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

Функции, реализованные равносильными формулами $x \vee yz$ и $(x \vee y)(x \vee z)$, равны. ◄

Равносильность формул можно доказать с помощью известных равносильных преобразований. В базисе $\Phi = \{^{-}, \wedge, \vee\}$ для краткости и наглядности обозначим $\vee$ через $+$, $\wedge$ будем опускать.

Для любых формул a, b, c следующие равносильности можно проверить с помощью соответствующих таблиц. В логике некоторые из этих равносильностей называются законами.

$a + b = b + a,$	$ab = ba,$	(коммутативность);
$a + (b + c) = (a + b) + c,$	$a(bc) = (ab)c,$	(ассоциативность);
$a + bc = (a + b)(a + c),$	$a(b + c) = ab + ac,$	(дистрибутивность);
$a + 0 = a,$	$a \cdot 1 = a;$	
$a + \bar{a} = 1,$	$a\bar{a} = 0;$	

Эти 5 пар равносильностей представляют собой аксиомы абстрактной булевой алгебры (см. стр. 29). Следующие равносильности можно либо вывести из аксиом, либо проверить по таблицам истинности.

$a + a = a,$	$a\ a = a$	(идемпотентность);
$a + 1 = 1,$	$a\ 0 = 0;$	
$\overline{a + b} = \bar{a}\ \bar{b},$	$\overline{a\ b} = \bar{a} + \bar{b}$	(законы де Моргана);
$a + a\ b = a,$	$a\ (a + b) = a$	(поглощение);
$\bar{\bar{a}} = a$		(двойное отрицание);
$ab + \bar{a}b = b,$	$(a + b)(\bar{a} + b) = b$	(склеивание);
$ab + \bar{a}c =$		
	$ab + \bar{a}c + bc$	(обобщенное склеивание).

Двойственные функции

Определение. *Функция*

$$f^*(x_1, \dots, x_n) = \bar{f}(\bar{x}_1, \dots, \bar{x}_n)$$

называется ***двойственной*** *к функции* $f(x_1, \dots, x_n) \in P_n$.

Примеры 3.10.

1. $(xy)^* = \overline{(\bar{x}\bar{y})} = x + y$;
2. $(x + y)^* = \overline{(\bar{x} + \bar{y})} = xy$; ◀

Для получения таблицы функции $f(\bar{x}_1, \dots, \bar{x}_n)$ нужно перевернуть таблицу исходной функции $f(x_1, \dots, x_n)$, а для получения таблицы функции $\bar{f}(\bar{x}_1, \dots, \bar{x}_n)$ еще и инвертировать столбец значений функции $f(\bar{x}_1, \dots, \bar{x}_n)$.

Пример 3.11. Построим таблицу функции, двойственной к функции $\oplus$:

Таблица 3.7. Двойственные функции

x	y	$\oplus(x, y)$	$\oplus(\bar{x}, \bar{y})$	$\oplus^*(x, y)$
0	0	0	0	1
0	1	1	1	0
1	0	1	1	0
1	1	0	0	1

Таким образом, функция двойственная к $\oplus$ есть $\leftrightarrow$ ($\oplus^* = \leftrightarrow$). ◀

Другие пары двойственных функций: $\vee^* = \wedge$, $\wedge^* = \vee$, $0^* = 1$, $1^* = 0$, $|^* = \downarrow$, $\downarrow^* = |$.

Теорема 3.3

$$f^{**}(x_1, \dots, x_n) = f(x_1, \dots, x_n).$$

Д о к а з а т е л ь с т в о .

$$(\bar{f}^*(\bar{x}_1, \dots, \bar{x}_n))^* = \bar{\bar{f}}(\bar{\bar{x}}_1, \dots, \bar{\bar{x}}_n) = f(x_1, \dots, x_n).$$

Определение. *Функция* f *называется самодвойственной, если* $f^* = f$.

Пример 3.12. Отрицание и тождественная функция самодвойственны: $^{-*} = ^{-}$

Теорема 3.4 ***(Принцип двойственности.)*** *Функция, двойственная суперпозиции функций, равна суперпозиции двойственных функций:*

$$f^*(f_1, \ldots, f_m) = f^*(f_1^*, \ldots, f_m^*)$$

Доказательство. $f^*(f_1, \ldots, f_m) =$

$$\begin{array}{l} \bar{f}(f_1(\bar{x}_1, \ldots, \bar{x}_n), \ldots, f_m(\bar{x}_1, \ldots, \bar{x}_n)) = \\ \bar{f}(\bar{\bar{f}}_1(\bar{x}_1, \ldots, \bar{x}_n), \ldots, \bar{\bar{f}}_m(\bar{x}_1, \ldots, \bar{x}_n)) = \\ \bar{f}(\bar{f}_1^*(x_1, \ldots, x_n), \ldots, \bar{f}_m^*(x_1, \ldots, x_n)) = \\ f^*(f_1^*(x_1, \ldots, x_n), \ldots, f_m^*(x_1, \ldots, x_n)) \end{array}$$

из определения

$$f^*(x_1, \ldots, x_n) = \bar{f}(\bar{x}_1, \ldots, \bar{x}_n).$$

Замечание. Принцип двойственности сформулирован для суперпозиции любых функций. Если ограничиться только базисными функциями, то принцип двойственности переносится на формулы.

Следствие 1. Если формула F реализует функцию $f(x_1, \ldots, x_n)$, то двойственная ей формула F^* реализует двойственную функцию $f^*(x_1, \ldots, x_n)$.

Следствие 2.

$$F_1 = F_2 \Rightarrow F_1^* = F_2^*$$

Примеры 3.13.

1. $a + b = b + a \Rightarrow ab = ba$.
2. $\overline{a + b} = \bar{a}\bar{b} \Rightarrow \overline{ab} = \bar{a} + \bar{b}$. ◀

3.3 Разложение функции по переменным (разложение Шеннона)

Введем обозначение

$$x^s = \begin{cases} x, s = 1, \\ \bar{x}, s = 0. \end{cases}$$

x^s имеет следующие свойства.

x	s	x^s	Комментарий
0	0	1	$0^0 = \bar{0} = 1$
0	1	0	$0^1 = 0$
1	0	0	$1^0 = \bar{1} = 0$
1	1	1	$1^1 = 1$

$x^s = 1 \Longleftrightarrow x = s$
$1^s = s, 0^s = \bar{s}$

Теорема 3.5 *Любую функцию $f(x_1, \ldots, x_n)$ можно представить в виде*

$$f(x_1, \ldots, x_k, x_{k+1} \ldots, x_n) = \bigvee_{(s_1,\ldots,s_k)} x_1^{s_1} \cdots x_k^{s_k} f(s_1, \ldots, s_k, x_{k+1}, \ldots, x_n), \qquad (\star)$$

(дизъюнкция по всем возможным наборам $(s_1, \ldots, s_k)$).

Доказательство. Покажем, что равенство выполняется для любого набора значений переменных $a_1, \ldots, a_n$.

$$f(a_1, \ldots, a_k, a_{k+1} \ldots, a_n) = \bigvee_{(s_1,\ldots,s_k)} a_1^{s_1} \cdots a_k^{s_k} f(s_1, \ldots, s_k, a_{k+1}, \ldots, a_n),$$

Рассмотрим выражение $a_1^{s_1} \cdots a_k^{s_k}$. Если хотя бы одно из значений $a_i^{s_i} = 0$, то и $a_1^{s_1} \cdots a_k^{s_k} = 0$. Тогда и

$$a_1^{s_1} \cdots a_k^{s_k} f(s_1, \ldots, s_k, a_{k+1}, \ldots, a_n) = 0.$$

Выражение $a_1^{s_1} \cdots a_k^{s_k} = 1$ только в том случае, когда $s_1 = a_1, \ldots, s_k = a_k \quad (x^s = 1 \Longleftrightarrow x = s)$. При этом

$$f(s_1, \ldots, s_k, a_{k+1}, \ldots, a_n) = f(a_1, \ldots, a_k, a_{k+1}, \ldots, a_n).$$

Равенство $(\star)$ выполняется при любом наборе значений переменных $(x_1, \ldots, x_n)$. $\boxtimes$

Формула $(\star)$ называется ***разложением*** Шеннона.

Пример 3.14. Найти разложение Шеннона по переменным x_1, x_2 булевой функции $f(x_1, x_2, x_3) = x_1(x_2 \to x_3)$:

$$f(x_1, x_2, x_3) = \bigvee_{(s_1,s_2)} x_1^{s_1} x_2^{s_2} f(s_1, s_2, x_3) =$$

$= x_1^0 x_2^0 f(0,0,x_3) \vee x_1^0 x_2^1 f(0,1,x_3) \vee x_1^1 x_2^0 f(1,0,x_3) \vee x_1^1 x_2^1 f(1,1,x_3) =$

$= \bar{x}_1 \bar{x}_2 0 \vee \bar{x}_1 x_2 0 \vee x_1 \bar{x}_2 1 (0 \to x_3) \vee x_1 x_2 (1 \to x_3) =$

$= 0 \vee 0 \vee x_1 \bar{x}_2 \vee x_1 x_2 x_3 = x_1 \bar{x}_2 \vee x_1 x_2 x_3.$

Следствие 1.

$f(x_1, \ldots, x_k, x_{k+1} \ldots, x_n) =$

$x_k f(x_1, \ldots, x_{k-1}, 1, x_{k+1} \ldots, x_n) \ \vee \bar{x}_k f(x_1, \ldots, x_{k-1} 0, x_{k+1} \ldots, x_n).$

Доказательство. Без умаления общности положим $k = 1$. При $x_1 = 0$ получим

$f(0, x_2, \ldots, x_n) = 0 f(1, x_2, \ldots, x_n) \vee \bar{0} f(0, x_2, \ldots, x_n) =$
$= f(0, x_2, \ldots, x_n).$

При $x_1 = 1$ получим

$f(1, x_2, \ldots, x_n) = 1 f(1, x_2, \ldots, x_n) \vee \bar{1} f(0, x_2, \ldots, x_n) =$
$= f(1, x_2, \ldots, x_n).$

Следствие 2.

$f(x_1, \ldots, x_n) =$

$$\bigvee_{f(s_1,\ldots,s_k)=1} x_1^{s_1} \cdots x_n^{s_n} f(s_1, \ldots, s_n)$$

– продолжили разложение Шеннона до $k = n$ и выбрали все ненулевые $f(s_1, \ldots, s_n)$.

СДНФ

Следствие 3. *Любая булева функция (кроме 0) имеет единственное представление в виде* $f(x_1, \ldots, x_n) = \bigvee\limits_{f(s_1,\ldots,s_k)=1} x_1^{s_1} \cdots x_n^{s_n}.$

Доказательство. Из следствия 2

$$f(x_1, \ldots, x_n) = \bigvee_{(s_1,\ldots,s_k)} x_1^{s_1} \cdots x_n^{s_n} f(s_1, \ldots, s_n) =$$

$$= \bigvee_{f(s_1,\ldots,s_k)=1} x_1^{s_1} \cdots x_n^{s_n} f(s_1,\ldots,s_n) = \bigvee_{f(s_1,\ldots,s_k)=1} x_1^{s_1} \cdots x_n^{s_n}.$$

Такое представление называется ***совершенной дизъюнктивной нормальной формой*** (СДНФ).

Следствие 3. можно сформулировать в виде теоремы: *Всякая булева функция (кроме 0) имеет единственную СДНФ.*

Следствие 4. *Любая булева функция может быть выражена через отрицание, конъюнкцию и дизъюнкцию* $(^-, \cdot, \vee)$.

Доказательство. Если $f = 0$, то $0 = x\bar{x}$. Если $f \neq 0$, то смотри следствие 3.

Построение СДНФ по таблице истинности

Построение СДНФ по таблице истинности выполняется в точности в соответствии со следствием 3. Для каждой очередной строки, в которой значение функции равно 1 $(f(s_1,\ldots,s_k) = 1)$, формируется конъюнкция всех переменных: если значение переменной $x_i = 0$, то в конъюнкцию эта переменная входит в степени 0 (x_i^0), то есть с отрицанием, в противном случае – в степени 1 (x_i^1), то есть без отрицания. Дизъюнкция всех конъюнкций образует СДНФ.

Пример 3.15.

Таблица 3.8. Построение СДНФ по таблице истинности

	x_1	x_2	x_3	f	Конъюнкция
0	0	0	0	0	
1	0	0	1	1	$\bar{x}_1\bar{x}_2x_3$
2	0	1	0	0	
3	0	1	1	1	$\bar{x}_1x_2x_3$
4	1	0	0	0	
5	1	0	1	1	$x_1\bar{x}_2x_3$
6	1	1	0	0	
7	1	1	1	1	$x_1x_2x_3$

$$f(x_1,x_2,x_3) = \bar{x}_1\bar{x}_2x_3 \vee \bar{x}_1x_2x_3 \vee x_1\bar{x}_2x_3 \vee x_1x_2x_3. \blacktriangleleft$$

СКНФ

Теорема 3.6 *Любую булеву функцию (кроме 1) можно представить в виде*

$$f(x_1,\ldots,x_n) = \bigwedge_{f(s_1,\ldots,s_n)=0} (x_1^{\bar{s}_1} \vee \cdots \vee x_n^{\bar{s}_n})$$

Д о к а з а т е л ь с т в о . Любую функцию, в том числе $f^*(x_1,\ldots,x_n) = \bar{f}(\bar{x}_1,\ldots,\bar{x}_n)$ можно представить в СДНФ:

$$f^*(x_1,\ldots,x_n) = \bigvee_{f^*(s_1,\ldots,s_k)=1} x_1^{s_1}\cdots x_n^{s_n}$$

$f = f^{*^*}$. Найдем двойственную к f^* функцию f :

$$f(x_1,\ldots,x_n) = \bigwedge_{f(s_1,\ldots,s_n)=0} (x_1^{\bar{s}_1} \vee \cdots \vee x_n^{\bar{s}_n})$$

Такая форма представления функции $f(x_1,\ldots,x_n)$ называется ***совершенной конъюнктивной нормальной формой*** (СКНФ).

Эту теорему можно сформулировать в виде:

Всякая булева функция (кроме 1) имеет единственную СКНФ.

Построение СКНФ по таблице истинности

Построение СКНФ по таблице истинности выполняется в соответствии с теоремой 3.6. Для каждой очередной строки, в которой $f(s_1,\ldots,s_k) = 0$, формируется дизъюнкция: если значение переменной $x_i = 0$, то в дизъюнкцию эта переменная входит степени 1 (x_i^1), в противном случае – в степени 0 (x_i^0). Конъюнкция всех дизъюнкций образует СКНФ.

Пример 3.16. Построим СКНФ для функции из примера 3.15.

Таблица 3.9. Построение СКНФ по таблице истинности

	x_1	x_2	x_3	f	Дизъюнкция
0	0	0	0	0	$x_1 \vee x_2 \vee x_3$
1	0	0	1	1	
2	0	1	0	0	$x_1 \vee \bar{x}_2 \vee x_3$
3	0	1	1	1	
4	1	0	0	0	$\bar{x}_1 \vee x_2 \vee x_3$
5	1	0	1	1	
6	1	1	0	0	$\bar{x}_1 \vee \bar{x}_2 \vee x_3$
7	1	1	1	1	

$$f(x_1, x_2, x_3) = (x_1 \vee x_2 \vee x_3)(x_1 \vee \bar{x}_2 \vee x_3)(\bar{x}_1 \vee x_2 \vee x_3)(\bar{x}_1 \vee \bar{x}_2 \vee x_3).$$

Представление формул в СДНФ и СКНФ

Пусть функция $f(x_1, \ldots, x_n)$ представлена формулой F в некотором базисе $(f_1, \ldots, f_k)$. Любая функция может быть выражена через три базисные операции $(^-, \cdot, \vee)$. Преобразование формулы к совершенным формам выполняется за несколько шагов.

1. Каждую подформулу (и сама формулу F) выражают через базисные операции $(^-, \cdot, \vee)$, используя разложение Шеннона или равносильные преобразования.

2. Отрицание вносят непосредственно к переменным, используя инволютивность отрицания $(\bar{\bar{F}} = F)$ и законы де Моргана.

3. Применяя нужное число раз дистрибутивные законы, формулу приводят к виду $\vee\, x_1^{'} \cdots x_k^{'}$ или $\wedge x_1^{'} \vee \cdots \vee x_k^{'}$, где $x_i^{'}$ – переменная или отрицание переменной (***литерал***).

4. Приведение подобных. Используя идемпотентность конъюнкции (дизъюнкции), удаляют повторные вхождения переменных в каждую конъюнкцию (дизъюнкцию) и повторные вхождения одинаковых конъюнкций в дизъюнкцию и одинаковых дизъюнкций в конъюнкцию. В результате формула не содержит повторных переменных, конъюнкций и дизъюнкций.

Определение. *Форма представления формулы в виде* $\vee\, x_1^{'} \cdots x_k^{'}$ *называется* ***дизъюнктивной нормальной формой (ДНФ).***

Конъюнкция $x_1^{'} \cdots x_k^{'}$, в которой все переменные различны, называется ***элементарной конъюнкцией (ЭК).***

Число переменных, образующих ЭК, называется ее ***рангом****. ЭК ранга n называется* ***полной****.*

Длина *ДНФ – это число ее элементарных конъюнкций*

Ранг *ДНФ – это сумма рангов ЭК, образующих ДНФ.*

Две ЭК ***ортогональны*** *по переменной x_i, если x_i входит в одну конъюнкцию с отрицанием, а в другую без отрицания. (В СДНФ все конъюнкции попарно ортогональны).*

Две ЭК ***смежны*** *по переменной x_i, если они ортогональны только по одной переменной x_i.*

Две ЭК являются ***соседними*** *по переменной x_i, если ортогональны только по x_i, а все остальные переменные одинаковы.*

Определение. *Форма представления формулы в виде $\wedge(x_1^{'} \vee \cdots \vee x_k^{'})$ называется* ***конъюнктивной нормальной формой (КНФ)****.*

Дизъюнкция $x_1^{'} \vee \cdots \vee x_k^{'}$, в которой все переменные различны, называется ***дизъюнктом (клозом)***[1]*.*

Число переменных дизъюнкта – это его ***ранг****.* ***Полный*** *дизъюнкт (ранга n) содержит все n переменных*

Ранг *КНФ – это сумма рангов дизъюнктов, образующих КНФ.*

Число дизъюнктов в КНФ – это ее ***длина****.*

Примеры 3.17.

1. Представить в ДНФ и КНФ формулу $\overline{(x_1 \vee \bar{x}_2)}(x_3 \to x_4)$.

$\overline{(x_1 \vee \bar{x}_2)}(x_3 \to x_4) =$	закон де Моргана и $x_3 \to x_4 = \bar{x}_3 \vee x_4$,
$= \bar{x}_1 x_2(\bar{x}_3 \vee x_4) =$	КНФ, дизъюнкты $\bar{x}_1, x_2, \bar{x}_3 \vee x_4$,
$= \bar{x}_1 x_2 \bar{x}_3 \vee \bar{x}_1 x_2 x_4$	ДНФ, ЭК $\bar{x}_1 x_2 \bar{x}_3,\ \bar{x}_1 x_2 x_4$.

Ранги дизъюнктов 1, 1 и 2 соответственно, ранг КНФ равен 4 $(1+1+2)$, длина КНФ равна 3.

Ранги ЭК 3 и 3 соответственно, ранг ДНФ равен 6 $(3+3)$, длина ДНФ равна 2.

2. ЭК $\bar{x}_1 \bar{x}_2 x_3$ и $x_1 x_2 x_3$ ортогональны по переменным x_1 и x_2; $\bar{x}_1 x_2 x_3$ и $x_1 \bar{x}_2 x_3$ также ортогональны по x_1 и x_2.

[1]Клоз – *лат., англ.* clause, предложение.

ЭК $\bar{x}_1 x_2 \bar{x}_3$ и $x_2 x_3 x_4$ – смежны по x_3, а $\bar{x}_1 \bar{x}_2 x_3$ и $\bar{x}_1 x_2 x_3$ – соседние по x_2. ◀

5. Если в некоторой элементарной конъюнкции K или дизъюнкте C не содержится переменная x_j, то ее добавляют:

$$K \cdot 1 = K(x_j \vee \bar{x}_j) = Kx_j \vee K\bar{x}_j,$$

$$C \vee 0 = D \vee x_j \bar{x}_j = (C \vee \ x_j)(C \vee \bar{x}_j).$$

6. Сортировка. Используя коммутативность, переменные в каждой ЭК (каждом дизъюнкте) сортируют в лексикографическом порядке.

Пример 3.18. Дополнить ДНФ и КНФ до совершенных.
ДНФ: $ab \vee \bar{b}c = (c \vee \bar{c})ab \vee \bar{b}c(a \vee \bar{a}) =$
$cab \vee \bar{c}ab \vee \bar{b}ca \vee \bar{b}c\bar{a}$.

В полученной СДНФ упорядочиваем переменные:
$abc \vee ab\bar{c} \vee a\bar{b}c \vee \bar{a}\bar{b}c$.

КНФ: $(a \vee b)(\bar{b} \vee c) = c\bar{c} \vee (a \vee b)(\bar{b} \vee c) \vee a\bar{a} =$
$= (c \vee a \vee b)(\bar{c} \vee a \vee b)(\bar{b} \vee c) =$
$= (c \vee a \vee b)(\bar{c} \vee a \vee b)(\bar{b} \vee c) \vee a\bar{a} =$
$= (c \vee a \vee b)(\bar{c} \vee a \vee b)(\bar{b} \vee c \vee a)(\bar{b} \vee c \vee \bar{a})$.

В полученной СКНФ упорядочиваем переменные:
$(a \vee b \vee c)(a \vee b \vee \bar{c})(a \vee \bar{b} \vee c)(\bar{a} \vee \bar{b} \vee c)$.

Представление БФ полиномами Жегалкина

При рассмотрении полных систем функций нам понадобится еще одна форма разложения булевых функций – полиномы Жегалкина. Рассмотрим некоторые свойства операции $\oplus$:

$a \oplus b = b \oplus a$ (коммутативность);
$a \oplus (b \oplus c) = (a \oplus b) \oplus c$ (ассоциативность);
$a(b \oplus c) = ab \oplus ac$ (дистрибутивность);
$a \oplus 0 = a, \ a \oplus 1 = \bar{a}$,
$a \oplus b \oplus ab = a \vee b$,
$a \oplus a = 0$.

Определение. *Функция $f(x_1, \ldots, x_n)$ представлена в форме* ***полинома Жегалкина****, если она имеет вид*

$$
\begin{aligned}
&f(x_1,\ldots,x_n) =\\
&a_0 \oplus a_1x_1 \oplus \cdots \oplus a_nx_n \oplus\\
&\oplus a_{12}x_1x_2 \oplus \cdots \oplus a_{n-1,n}x_{n-1}x_n \oplus\\
&\oplus a_{123}x_1x_2x_3 \oplus \cdots \oplus a_{n-2,n-1,n}x_{n-2}x_{n-1}x_n \oplus\\
&\cdots\cdots\\
&\oplus a_{1,\ldots,n}x_1x_2\cdots x_n,
\end{aligned}
$$

$a_{i\ldots k} \in \{0,1\}$ – ***коэффициенты*** *полинома*, **длина** *полинома – это число ЭК полинома*, **степень** *полинома – наибольший из рангов ЭК, входящих в полином.*

З а м е ч а н и е.

Константа 1 считается полиномом длины 1 и степени 0, константа 0 – полиномом длины 0 и степени 0. ►

Определение. *Полином* ***линейный****, если его степень* $\leqslant 1$.

Пример 3.19. $1 \oplus x_3 \oplus x_1x_2 \oplus x_2x_4x_5x_6$ – полином Жегалкина, его длина равна 4, степень равна 4.

Полином $1 \oplus x_1 \oplus x_2 \oplus x_5$ – линейный. ◄

Теорема 3.7 *Любую булеву функцию можно единственным образом представить в виде полинома Жегалкина.*

Д о к а з а т е л ь с т в о . (Конструктивное.) Если $f = 0$, то $f = a_0 = 0$ – полином Жегалкина.

Если $f \neq 0$, то f можно представить (единственным образом) в форме СДНФ. В СДНФ операцию $\vee$ заменяют на $\oplus$. Почему можно это сделать? Для двух различных ЭК K_i и K_j

$$K_i \vee K_j = K_i \oplus K_j \oplus K_iK_j \text{ (свойство операции } \oplus.)$$

Так как все конъюнкции в СДНФ попарно ортогональны, то $K_iK_j = 0$ и $K_i \vee K_j = K_i \oplus K_j$. Переменные с отрицанием $\bar{x}$ заменяют на $x \oplus 1$. Затем раскрывают скобки и приводят подобные ($x \oplus x = 0$).

Единственность представления полиномом Жегалкина следует из единственности представления в СДНФ.

Пример 3.20. Представить СДНФ

$$f(x_1,x_2,x_3) = \bar{x}_1\bar{x}_2x_3 \vee \bar{x}_1x_2x_3 \vee x_1\bar{x}_2x_3$$

полиномом Жегалкина.

$$f(x_1, x_2, x_3) =$$
$$= (x_1 \oplus 1)(x_2 \oplus 1)x_3 \oplus (x_1 \oplus 1)x_2x_3 \oplus x_1(x_2 \oplus 1)x_3 =$$
$$= (x_1x_2 \oplus x_1 \oplus x_2 \oplus 1)x_3 \oplus x_1x_2x_3 \oplus x_2x_3 \oplus x_1x_2x_3 \oplus x_1x_3 =$$
$$= \underline{x_1x_2x_3} \oplus \underline{x_1x_3} \oplus \underline{x_2x_3} \oplus x_3 \oplus \underline{x_1x_2x_3} \oplus \underline{x_2x_3} \oplus x_1x_2x_3 \oplus \underline{x_1x_3} =$$
$$= x_1x_2x_3 \oplus x_3.$$

Подчеркнуты подобные удаляемые ЭК:
$\underline{x_1x_2x_3} \oplus \underline{x_1x_2x_3} \oplus x_1x_2x_3 = x_1x_2x_3$, $\underline{x_1x_3} \oplus \underline{x_1x_3} = 0$. ◄

3.4 Минимизация булевых функций

Импликанта

Определение. *Функция $g(x_1, \ldots, x_n)$ называется импликантой функции $f(x_1, \ldots, x_n)$, если*

$$g(x_1, \ldots, x_n) \Rightarrow f(x_1, \ldots, x_n),$$

или, что то же самое,

$$g(x_1, \ldots, x_n) \to f(x_1, \ldots, x_n) = 1.$$

Другими словами, $g(x_1, \ldots, x_n)$ есть импликанта функции $f(x_1, \ldots, x_n)$, если на всех тех векторах $a_1 \cdots a_n$, на которых, если $g(a_1, \ldots, a_n) = 1$, то $f(a_1, \ldots, a_n) = 1$. Таким образом, если K – любая ЭК из ДНФ функции $f(x_1, \ldots, x_n)$, то K есть импликанта этой функции.

Определение. *Импликанта K_i функции $f(x_1, \ldots, x_n)$ называется* ***простой****, если ее не поглощает никакая другая импликанта K_j этой функции.*

ДНФ, состоящая из всех простых импликант функции $f(x_1, \ldots, x_n)$, называется ***сокращенной*** *ДНФ этой функции.*

ДНФ называется ***минимальной****, если она имеет наименьший ранг из всех ДНФ функции $f(x_1, \ldots, x_n)$ (содержит наименьшее число литералов).*

ДНФ называется ***кратчайшей****, если она имеет наименьшую длину из всех ДНФ функции $f(x_1, \ldots, x_n)$ (содержит наименьшее число ЭК).*

*ДНФ называется **безызбыточной**, если из нее нельзя удалить ни одной переменной и ни одной импликанты так, чтобы она оставалась равносильной исходной ДНФ.*

Замечания. Любые минимальные и кратчайшие ДНФ являются безызбыточными. Кратчайшая ДНФ не обязательно минимальна, но поиск минимальной ДНФ проводится среди кратчайших. ►

Минимизация БФ

Задачи минимизации булевой функции: найти кратчайшую или минимальную ДНФ; найти все кратчайшие или все минимальные ДНФ. Каждый раз задача конкретизируется. В любом случае сначала находятся все простые импликанты функции (сокращенная ДНФ). Из сокращенной ДНФ находят нужные ДНФ.

Пусть K, K_1, K_2 – элементарные конъюнкции, входящие в СДНФ, или в уже полученные ДНФ, реализующие исходную булеву функцию $f(x_1, \ldots, x_n)$. Для минимизации ДНФ используются следующие свойства.

Поглощение: $K_1 \vee K_1K_2 = K_1$.

Склеивание/расщепление (простая склейка): $xK \vee \bar{x}K = K$.

Неполное склеивание: $xK \vee \bar{x}K = xK \vee \bar{x}K \vee K$.

Обобщенное склеивание: $xK_1 \vee \bar{x}K_2 = xK_1 \vee \bar{x} \vee K_1K_2$.

ДНФ, полученная в результате многократного применения склеек и поглощений, является сокращенной ДНФ, а все ЭК – простыми импликантами функции $f(x_1, \ldots, x_n)$.

Примеры 3.21. 1. $f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2x_3 \vee x_1\bar{x}_2\bar{x}_3 \vee x_1\bar{x}_2x_3$.

Все импликанты простые, ни одна из них не поглощает другую.

Склеиваем импликанты 1, 3 и 2, 4 по переменной x_1.

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

Снова все импликанты простые, ДНФ сокращенная.

Еще ода склейка по переменной x_3:

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

ДНФ сокращенная, кратчайшая, минимальная, равносильная исходной функции (все преобразования были равносильными). В результате склеек были удалены и фиктивные переменные x_1 и x_3.

2. Мажоритарная функция.

$f(x_1, x_2, x_3) = \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 x_3 \vee x_1 x_2 \bar{x}_3 \vee x_1 x_2 x_3$.

Склеиваем импликанты 1, 4 по переменной x_1 и импликанты 2, 4 по переменной x_2 :

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

Импликанта $x_1 x_2 \bar{x}_3$ поглощена импликантой $x_1 x_2$. ◄

Исторически идея нахождения сокращенной ДНФ путем многократных склеиваний/поглощений формулируется как теорема Квайна.

Теорема 3.8 (Квайна) *Для получения сокращенной ДНФ из СДНФ нужно выполнить все возможные неполные склеивания соседних импликант, а затем все поглощения импликант.*

Д о к а з а т е л ь с т в о . Действительно, после всех склеиваний и поглощений все оставшиеся импликанты будут простыми (никакая из импликант не поглощает другую), а ДНФ – сокращенной. ⊠

Алгоритм Квайна – МакКласки

Алгоритм строит сокращенную ДНФ из таблицы функции или буквенной записи СДНФ. Алгоритм основан на теореме Квайна. Модификация МакКласки заключается в упорядочении импликант по весам. Рассмотрим на примере последовательность шагов алгоритма.

Пример 3.22. Найти сокращенную ДНФ булевой функции.

СДНФ функции $f = \bar{a}\bar{b}\bar{c}d \vee \bar{a}\bar{b}cd \vee \bar{a}b\bar{c}d \vee \bar{a}bcd \vee abc\bar{d} \vee abcd.$

1. Импликанты СДНФ задаются таблицей, состоящей из их двоичных номеров (векторов). Векторы группируются по весам (числу единиц в векторе). Номер группы G_i равен весу векторов в группе. Таким образом, соседние импликанты находятся в соседних по номерам группах.

	#	a	b	c	d	
G_1	1	0	0	0	1	$*$
G_2	2	0	0	1	1	$*$
	3	0	1	0	1	$*$
G_3	4	0	1	1	1	$*$
	5	1	1	1	0	$*$
G_4	6	1	1	1	1	$*$

2. Выполняются неполные склеивания всех соседних импликант в соседних группах G_i и G_{i+1}. Разбиение на группы позволяет сократить число попарных сравнений при склеивании. Склеиваемые импликанты помечаются значком $*$. Результаты склеиваний (склейки) вместе с непомеченными импликантами заносятся в новую таблицу (без повторений). В склейках значком $\times$ помечается переменная, по которой выполнена склейка (вес значка $\times$ равен 0). В правом столбце показаны номера склеенных импликант из предыдущей таблицы.

	#	a	b	c	d		
G_1	1	0	0	$\times$	1	$*$	$(1,2)$
	2	0	$\times$	0	1	$*$	$(1,3)$
	3	0	$\times$	1	1	$*$	$(2,4)$
G_2	4	0	1	$\times$	1	$*$	$(3,4)$
	5	$\times$	1	1	0		$(4,6)$
G_3	6	1	1	1	$\times$		$(5,6)$

3. Если полученная таблица непуста, то с ней выполняется 2-й шаг алгоритма. Импликанты в этой таблице уже упорядочены по весам. Склеивание можно выполнять только для тех импликант, у которых $\times$ находятся в одинаковых позициях. Получаем новую таблицу.

	#	a	b	c	d		
G_1	1)	0	$\times$	$\times$	1		$(1,4)$
G_3	2)	$\times$	1	1	1		
	3)	1	1	1	$\times$		

Первая импликанта получена при склеивании импликант $(1, 4)$ и $(2, 3)$ из предыдущей таблицы, импликанты $2), 3)$ внесены как непомеченные. В этой таблице нет соседних импликант, работа алгоритма закончена. В результате получена сокращенная ДНФ $f = abc \vee \bar{a}d \vee bcd$ длины 3 и ранга 8.

Алгоритм Блейка – Порецкого

Алгоритм нахождения сокращенной ДНФ из СДНФ или равносильной ей ДНФ основан на выполнении всех возможных обобщенных склеиваний и всех возможных поглощений. Исторически идея этого алгоритма формулируется как теорема Блейка.

Теорема 3.9 (Блейка) *Сокращенную ДНФ булевой функции можно получить, выполнив все обобщенные склеивания и все возможные поглощения.*

Рассмотрим на примере работу алгоритма.

Пример 3.23. Булева функция задана ДНФ

$$f = \bar{a}bcd \vee a\bar{b}c\bar{d} \vee a\bar{b}c \vee \bar{a}b\bar{c} \vee \bar{a}\bar{c}d \vee \bar{b}\bar{c}\bar{d}.$$

Представим импликанты ДНФ троичными векторами, заменив недостающие переменные значком $\times$. В обобщенном склеивании $xK_1 \vee \bar{x}K_2 = xK_1 \vee \bar{x}K_2 \vee K_1K_2$ участвуют только смежные импликанты, ортогнальные по переменной x. Результат обобщенного склеивания K_1K_2 вносится в таблицу (без повторений) со следующим по счету номером; при этом в графу С вносятся номера склеиваемых векторов; в строку графы П вносится номер вектора, который ее поглощает.

Вектор 5 сразу поглощается вектором 6; вектор 7 получается при склеивании векторов 1 и 2, вектор 8 – из 1 и 3, вектор 9 – из 1 и 6, вектор 10 – из 3 и 4; вектор 4 поглощается вектором 10; из склеивания векторов 3 и 7 получается вектор 11, который поглощает векторы 2, 3, 7 и 8. Склеивания других векторов дают повторения: например, склеивание векторов 9 и 11 дает вектор 1.

#	a	b	c	d	С	П
1	$\times$	0	0	0		
2	0	$\times$	0	1		11
3	0	1	0	$\times$		11
4	0	1	1	1		10
5	1	0	1	0		6
6	1	0	1	$\times$		
7	0	0	0	$\times$	$1,2$	11
8	0	$\times$	0	0	$1,3$	11
9	1	0	$\times$	0	$1,6$	
10	0	1	$\times$	1	$3,4$	
11	0	$\times$	0	$\times$	$3,7$	

Непоглощенные векторы есть простые импликанты полученной сокращенной ДНФ:

$$\bar{b}\bar{c}\bar{d} \vee a\bar{b}c \vee a\bar{b}\bar{d} \vee \bar{a}bd \vee \bar{a}\bar{c}$$

#	a	b	c	d
1	$\times$	0	0	0
6	1	0	1	$\times$
9	1	0	$\times$	0
10	0	1	$\times$	1
11	0	$\times$	0	$\times$

Таблица Квайна

Алгоритмы Квайна – МакКласки и Блейка – Порецкого находят сокращенную ДНФ булевой функции. На втором этапе минимизации выполняется поиск кратчайшей или минимальной ДНФ. Для этого нужно в сокращенной ДНФ найти такие подмножества импликант, которые поглощают все импликанты СДНФ.

Матрица, строкам которой соответствуют импликанты СДНФ, а строкам импликанты сокращенной ДНФ, называется таблицей Квайна $Q = \|q_{ij}\|$. Элемент таблицы $q_{ij} = 1$, если импликанта столбца j поглощает импликанту строки i, $q_{ij} = 0$ в противном случае.

3.5 Полные системы булевых функций

Ранее было показано, что любая булева функция может быть представлена в ДНФ, в КНФ, или полиномом Жегалкина, то есть может быть выражена в базисах элементарных функций $\{\bar{\ }, \cdot, \vee\}$, или $\{1, \cdot, \oplus\}$.

Определение. *Подмножества функций* $\Phi = \{f_1, \ldots, f_m\}$, $\Phi \subseteq P_n$, *через которые может быть выражена любая булева функция, называются* ***функционально полными*** *(ФП) системами булевых функций.*

Теорема 3.10 *Пусть* Φ_1 *функционально полная система и любая функция из нее может быть представлена суперпозицией функций из множества функций* Φ_2. *Тогда* Φ_2 *– функционально полная система.*

Доказательство. Булева функция $f(x_1, \ldots, x_n)$ может быть представлена суперпозицией функций из полной системы $\Phi_1 = \{f_0, f_1, \ldots, f_m\}$

$$f(x_1, \ldots, x_n) = f_0(f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n)).$$

Но любая функция $f_0, f_1, \ldots, f_m$ может быть представлена суперпозицией функций из множества Φ_2, следовательно любая $f(x_1, \ldots, x_n)$ также может быть представлена суперпозицией функций из Φ_2. Таким образом, Φ_2 – функционально полная система. ⊠

Пример 3.24. $\Phi = \{\,\bar{\ }, \cdot, \vee\}$ – полная система; $\{\,\bar{\ }, \cdot\}$ тоже полная, так как функцию $\vee$ можно представить суперпозицией функций: $x \vee y = \overline{\bar{x}\bar{y}}$; система $\{\,\bar{\ }, \vee\}$ полная, так как функцию $\cdot$ можно представить суперпозицией $x \cdot y = \overline{\bar{x} \vee \bar{y}}$. Полную систему $\{\,\bar{\ }, \cdot\}$ можно представить суперпозицией $\bar{x} = x \downarrow x, xy = \overline{\bar{x} \vee \bar{y}} = (x \downarrow x) \downarrow (y \downarrow y)$, а полную систему $\{\,\bar{\ }, \vee\}$ суперпозицией $\bar{x} = x|x$, $x \vee y = \overline{\bar{x}\bar{y}} = (x|x)|(y|y)$. Значит системы $\{\downarrow\}$ и $\{|\}$ тоже полные. ◄

Теорема 3.10 служит основанием для выбора функционально полных систем функций. Традиционно за исходную принимают полную систему $\{^{-}, \cdot\}$ и находят такую систему функций, в которой могут быть представлены функции $^{-}$ и $\cdot$.

Рассмотрим сначала такие множества функций, которые заведомо не являются функционально полными системами.

Замкнутые классы булевых функций

Определение. *Множество* $\Phi = \{f_1, \ldots, f_m\}$ *называется* ***замкнутым классом*** *функций, если суперпозиция любых функций из* Φ *принадлежит этому же множеству.*

Множество всех булевых функций n переменных $P_n = \{f | f : \{0,1\}^n \mapsto \{0,1\}\}$ – замкнутый класс, так как суперпозиция любых булевых функций есть булева функция. Рассмотрим некоторые из замкнутых подмножеств множества P_n.

1. Класс булевых функций, ***сохраняющих константу 0***:

$$S_0 = \{f | f(0, 0, \cdots, 0) = 0\}$$

(на наборе всех нулей функция принимает значение 0).

Классу S_0 принадлежат, например, мажоритарная, $\cdot$ и $\leftrightarrow$ функции; $|$ и $\downarrow$ не принадлежат S_0.

2. Класс функций, ***сохраняющих константу 1***:

$$S_1 = \{f | f(1, 1, \cdots, 1) = 1$$

(на наборе всех единиц функция принимает значение 1).

Мажоритарная функция, $\cdot$, $\vee$ сохраняют 1, а $|$ и $\downarrow$ не сохраняют.

3. Класс ***самодвойственных*** функций

$$S_* = \{f | f = f^*\}$$

$$(f(x_1, \ldots, x_n) = f^*(x_1, \ldots, x_n) = \bar{f}(\bar{x}_1, \ldots, \bar{x}_n)).$$

Из элементарных функций самодвойственны только $\leftrightarrow$ и $^{-}$.

4. Класс ***монотонных функций***

$$S_M = \{f | \vec{a} \preccurlyeq \vec{b} \Rightarrow f(\vec{a}) \leqslant f(\vec{b})\}.$$

Функции $\cdot, \vee$, мажоритарная монотонны, а $|$ и $\downarrow$ немонотонны.

5. Класс ***линейных*** функций

$$S_L = \{f | f = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus \cdots \oplus a_n x_n\}$$

(полином Жегалкина первой степени).
Линейны функции $^{-}$ и $\leftrightarrow$, а $|, \downarrow$ и мажоритарная нелинейны.

Теорема 3.11 *Классы S_0, S_1, S_*, S_M, S_L замкнуты.*

Доказательство.

1. Суперпозиция функций из класса S_0 сохраняет 0:

$$f(0, \ldots, 0) = f_0(f_1(0, \ldots, 0), \ldots, f_k(0, \ldots, 0)) = f_0(0, \ldots, 0) = 0.$$

2. Для класса S_1 доказательство аналогмчно.
3. Класс самодвойственных функций S_* замкнут:

$$f^*(x_1, \ldots, x_n) = f_0^*(f_1^*(x_1, \ldots, x_n), \ldots, f_k^*(x_1, \ldots, x_n)) =$$

$$f_0(f_1(x_1, \ldots, x_n), \ldots, f_k(x_1, \ldots, x_n))$$

4. В суперпозиции монотонных функций

$$f_0(f_1(x_1, \ldots, x_n), \ldots, f_k(x_1, \ldots, x_n))$$

подставим векторы $\vec{a} = (a_1, \ldots, a_n)$ и $\vec{b} = (b_1, \ldots, b_n)$, такие,что $\vec{a} \preccurlyeq \vec{b}$. Пусть

$$f(\vec{a}) = f_0(f_1(\vec{a}), \ldots, f_k(\vec{a})) = f_0(\vec{c}),$$

$$f(\vec{b}) = f_0(f_1(\vec{b}), \ldots, f_k(\vec{b})) = f_0(\vec{d}).$$

Так как $\vec{a} \preccurlyeq \vec{b}$ и функции $f_1, \ldots, f_k$ монотонны, то $\vec{c} \preccurlyeq \vec{d}$, а так как f_0 монотонна, то $f_0(\vec{c}) \leqslant f_0(\vec{d})$. Значит $f(\vec{a}) \leqslant f(\vec{b})$, то есть класс S_M замкнут.

5. В суперпозиции линейных функций

$$f_0(f_1(x_1, \ldots, x_n), \ldots, f_k(x_1, \ldots, x_n))$$

каждую функцию можно представить в виде

$$f_0(y_1, \ldots, y_k) = a_0 \oplus a_1 y_1 \oplus \cdots \oplus a_k y_k,$$

$$f_1(x_1,\dots,x_n)=b_0^1\oplus b_1^1x_1\oplus\dots\oplus b_n^1x_1,$$

$$-------------------,$$

$$f_k(x_1,\dots,x_n)=b_0^k\oplus b_1^kx_1\oplus\dots\oplus b_n^kx_n;$$

тогда

$$f(x_1,\dots,x_n)=f_0(f_1(x_1,\dots,x_n),\dots,f_k(x_1,\dots,x_n))=$$

$$a_0\oplus a_1f_1(x_1,\cdots,x_n)\oplus\dots\oplus a_kf_k(x_1,\cdots,x_n)=$$

$$a_0\oplus a_1(b_0^1\oplus b_1^1x_1\oplus\dots\oplus b_n^1x_n)\oplus\dots\oplus a_k(b_0^k\oplus b_1^kx_1\oplus\cdots b_n^kx_n)=$$

$$(a_0\oplus a_1b_0^1\oplus\dots\oplus a_kb_0^k)\oplus(a_1b_1^1\oplus\dots\oplus a_kb_1^k)x_1\oplus\dots\oplus(a_nb_n^1\oplus\dots\oplus a_kb_n^k)x_n.$$

Каждая скобка есть константа, значит функция $f(x_1,\dots,x_n)$ линейна и класс S_L замкнут. ⊠

Следствие. Классы функций S_0, S_1, S_*, S_M, S_L не являются полными.

Действительно, суперпозиции функций из каждого класса дают функции из того же класса.

Критерий Поста

Подход к построению функционально полной системы заключается в следующем. Используя функции, не сохраняющую константу 0, не сохраняющую константу 1 и несамодвойственную можно представить константы 0 и 1. С помощью констант и немонотонной функции можно получить отрицание. С помощью констант, отрицания и нелинейной функции можно получить конъюнкцию или дизъюнкцию. Таким образом, суперпозицией этих функций можно представить функционально полные системы функций $\{^-,\cdot\}$ или $\{^-,\vee\}$. По теореме 3.10 такое множество функций также функционально полно.

Теорема 3.12 (Теорема Поста) *Множество $\Phi\in P_n$ булевых функций функционально полно тогда и только тогда, когда оно **не** является подмножеством ни одного из классов S_0, S_1, S_*, S_M, S_L. Другими словами, функционально полная система* *функций должна содержать хотя бы одну функцию, не сохраняющую константу 0, хотя бы одну функцию, не сохраняющую константу 1, хотя бы одну нелинейную функцию, хотя бы одну несамодвойственную функцию и хотя бы одну немонотонную функцию.*

Доказательство.

Необходимость.

Предположим, что Φ содержит хотя бы одну функцию из любого из классов $S \in \{S_0,\ S_1,\ S_*,\ S_M,\ S_L\}$. Так как класс S замкнут, то функция $f \notin S$ не может быть представлена никакой суперпозицией функций из S. Значит Φ не является функционально полным множеством.

Достаточность.

1. *Из любой функции, не сохраняющей константу 0, можно получить либо константу 1, либо отрицание.*

Пусть $f(x_1, \ldots, x_n)$ не сохраняет 0. Тогда $f(0, 0, \ldots, 0) = 1$. Если $f(1, 1, \ldots, 1) = 1$, то функция одной переменной $g(x) = f(x, x, \ldots, x)$ дает константу 1. Если $f(1, 1, \ldots, 1) = 0$, то $g(x) = f(x, x, \ldots, x) = \bar{x}$.

Аналогично, *из любой функции, не сохраняющей константу 1, можно получить либо константу 0, либо отрицание.*

2. *Из любой несамодвойственной функции с помощью отрицания можно получить константы 0 и 1.*

Для несамодвойственной функции

$$f(x_1, \ldots, x_n) \neq f^*(x_1, \ldots, x_n)$$

существует такой набор $\vec{a} = (a_1, \ldots, a_n)$, что

$$f(a_1, \ldots, a_n) \neq \bar{f}(\bar{a}_1, \ldots, \bar{a}_n),$$

то есть

$$f(a_1, \ldots, a_n) = f(\bar{a}_1, \ldots, \bar{a}_n). \qquad (\star)$$

Заменим каждый аргумент x_i функции $f(x_1, \ldots, x_n)$ на x^{a_i}. Напомним, что $0^x = \bar{x}, 1^x = x$. Для полученной функции одной переменной $g(x) = f(x^{a_1}, \ldots, x^{a_n})$

$$g(0) = f(0^{a_1}, \ldots, 0^{a_n}) = f(\bar{a}_1, \ldots, \bar{a}_n),$$

$$g(1) = f(1^{a_1}, \ldots, 1^{a_n}) = f(a_1, \ldots, a_n).$$

Учитывая равенство ($\star$), получим, что $g(0) = g(1)$. Таким образом, $g(x)$ – одна константа, $\bar{g}(x)$ – другая константа.

3. *Из любой немонотонной функции и констант 0 и 1 можно получить отрицание.*

Для немонотонной функции $f(x_1, \ldots, x_n)$ существуют по крайней мере два набора $\vec{a} = (a_1, \ldots, a_n)$ и $\vec{b} = (b_1, \ldots, b_n)$ такие, что $\vec{a} \preccurlyeq \vec{b}$ и $f(\vec{a}) > f(\vec{b})$ $(f(\vec{a}) = 1, f(\vec{b}) = 0)$. Выберем в $\vec{a}$ такие компоненты, для которых $a_i < b_i$ $(a_i = 0, b_i = 1)$ и подставим переменную x вместо $a_i = 0$. Для полученной таким образом функции $g(x)$ одной переменной имеем

$$g(0) = f(a_1, \ldots, a_n) = 1,$$

$$g(1) = f(b_1, \ldots, b_n) = 0,$$

то есть $g(x) = \bar{x}$.

4. *Из любой нелинейной функции с помощью подстановки констант и отрицания можно получить конъюнкцию или дизъюнкцию.*

В представлении нелинейной функции полиномом Жегалкина сгруппируем конъюнкции таким образом, что первое слагаемое содержит конъюнкцию x_1x_2, второе содержит x_1 и не содержит x_2, третье слагаемое содержит x_2 и не содержит x_1; четвертое слагаемое содержит все остальные конъюнкции:

$$f(x_1, \ldots, x_n) = x_1x_2f_1(x_3, \ldots, x_n) \oplus x_1f_2(x_3, \ldots, x_n) \oplus$$

$$\oplus\ x_2f_3(x_3, \ldots, x_n) \oplus f_4(x_3, \ldots, x_n)$$

$x_1x_2f_1(x_3, \ldots, x_n) \neq 0$, так как найдется хотя бы одна конъюнкция, содержащая x_1x_2 (x_1 и x_2 выбраны произвольно без нарушения общности). Это значит, что найдется такой набор $a_3, \ldots, a_n$, что $f_1(a_3, \ldots, a_n) = 1$. Тогда

$$f(x_1, x_2, a_3, \ldots, a_n) = x_1x_2 \oplus x_1f_2(a_3, \ldots, a_n) \oplus$$

$$\oplus x_2f_3(a_3, \ldots, a_n) \oplus f_4(a_3, \ldots, a_n) =$$

$$g(x_1, x_2) = x_1x_2 \oplus c_1x_1 \oplus c_2x_2 \oplus c_3, \quad c_1, c_2, c_3 \in \{0, 1\}.$$

Если $c_3 = 0$, то
если $c_1 = 0, c_2 = 0$, то $g(x_1, x_2) = x_1x_2$;
если $c_1 = 0, c_2 = 1$, то $g(\bar{x}_1, x_2) = \bar{x}_1x_2 \oplus x_2 = x_2(1 \oplus \bar{x}_1) = x_1x_2$ (подставили отрицание в x_1;)
если $c_1 = 1, c_2 = 0$, то $g(x_1, \bar{x}_2) = x_1\bar{x}_2 \oplus x_1 = x_1(1 \oplus \bar{x}_2) = x_1x_2$; (подставили отрицание в x_2;)
если $c_1 = 1, c_2 = 1$, то $g(x_1, x_2) = x_1x_2 \oplus x_1 \oplus x_2 = x_1 \vee x_2$.

Если $c_3 = 1$, то все варианты получаются как отрицания вариантов при $c_3 = 0$ $(x \oplus 1 = \bar{x})$.

Часть 2

Математическая логика

Глава 4

Логика высказываний

4.1 Основные понятия

Высказывания и формулы

Высказывание *– это утвердительное (повествовательное) предложение, которое может быть либо истинным, либо ложным, но не тем и другим вместе.* "***Истина***"или "***ложь*** приписанная некоторому высказыванию, называется ***истинностным*** значением высказывания.

Истинностные значения "истина"и "ложь"обозначаются, соответственно, И и Л, T и F (True, False), или 1 и 0. Чаще других мы будем использовать обозначения 1 и 0.

Символы p, q, r и т.д., которые используются для обозначения высказываний называются высказывательными (пропозициональными) ***переменнымии***, *атомарными формулами* или ***атомами***.

Примеры 4.1.

1. Земля вертится (фактическая истина: высказывание выражает некоторый факт из физики и астрономии).

2. Если истинно утверждение: когда идет дождь, дорога мокрая, то истинно и следующее утверждение: если дорога сухая, то дождя нет. ("Идет дождь", "дорога сухая", "дорога мокрая" – атомарные формулы (атомы), "если – то" – логи-

ческая связка). Истинностное значение всего предложения – *логическая истина.* ◀

Из атомов можно строить составные высказывания – ***формулы***, используя ***логические связки*** или логические ***операции***:

Операция	Альтернативные обозначения
отрицание	$\neg \mid \sim \mid not \mid$не
конъюнкция	$\wedge \mid \& \mid \cdot \mid, \mid and \mid$и
дизъюнкция	$\vee \mid + \mid; \mid or \mid$или
импликация	$\to \mid \subset \mid \supset \mid if \cdots then \mid$если $\cdots$ то
эквивалентность	$\leftrightarrow \mid \equiv \mid iff \mid$тогда и только тогда, когда

Примеры 4.2.
1. Из атомарных высказываний
p : команда выиграла на своем поле
q : команда проиграла на чужом поле
r : команда выходит в следующий круг соревнований
можно построить составное высказывание

$(p \wedge (\neg q)) \to r$:

если команда выиграла на своем поле и не проиграла на чужом поле, то она выходит в следующий круг соревнований.
2. Если студент сдал сессию без троек, или если он получил не более одной тройки и за него ходатайствует группа, то он будет получать стипендию (из легенды). ◀

Определение. ***Правильно построенная формула*** *(ППФ) или просто* ***формула*** *в логике высказываний определяется следующим образом:*
1. Атом – формула (1 *и* 0 *– формулы).*
2. Если F *– формула, то и* $(\neg F)$ *– формула.*
3. Если F *и* G *– формулы, то*
$(F \vee G)$, $(F \wedge G)$, $(F \to G)$, $(F \leftrightarrow G)$ *– формулы.*
4. Формулы порождаются только конечным числом применений правил 1 – 3.

Логические (пропозициональные) связки упорядочиваются по приоритетам (в порядке убывания):

$$\neg, \wedge, \vee, \rightarrow, \leftrightarrow.$$

Поэтому некоторые скобки в формулах можно опускать, например:

$(p \vee q)$ можно заменить на $p \vee q$ (внешние скобки обычно опускаются), $(p \rightarrow (q \wedge r))$ – на $p \rightarrow q \wedge r$, $p \rightarrow (q \wedge (\neg r))$ – на $p \rightarrow q \wedge \neg r$.

Истинностные значения формул определяются с помощью ***таблиц истинности*** логических операций:

Таблица 4.1. Таблица итинности логических операций

p	$\neg p$
0	1
1	0

p	q	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	0
1	1	1	1	1	1

Истинностные значения формулы вычисляются по истинностным значениям атомов, входящих в эту формулу, и таблицам истинности логических связок.

Интерпретация формул

Пусть F – данная пропозициональная формула и $a_1, \ldots a_n$ – входящие в нее атомы. ***Интерпретацией*** формулы F является приписывание истинностных значений атомам $a_1, \ldots, a_n$, таких, что каждому a_i приписано либо 1, либо 0 (но не оба вместе). Формула F “истинна "в некоторой интерпретации тогда и только тогда, когда F получает значение 1 в (при) этой интерпретации. В противном случае формула F является ложной в (при) этой интерпретации.

Если формула содержит n атомов, то она имеет 2^n различных интерпретаций.

Пусть I – некоторая интерпретация формулы F.

Говорят, что I *удовлетворяет* формуле F, если $F = 1$ в I и I *опровергает* F, если $F = 0$ в I.

Пример 4.3.
Найдем все интерпретации формулы $(p \wedge q) \to r$:

Таблица 4.2. Интерпретации формулы $(p \wedge q) \to r$

p	q	r	$p \wedge q$	$p \wedge q \to r$
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	1	0
1	1	1	1	1

З а м е ч а н и е. Интерпретация, при которой истинностное значение формулы есть 1, называется ***моделью*** этой формулы. ▶

Общезначимость и противоречивость

Пример 4.4. Рассмотрим формулу $F : (p \to q) \wedge p \to q$ и найдем все ее интерпретации. Число атомов в формуле $n = 2$; формула имеет $2^2 = 4$ интерпретации:

Таблица 4.3. Интерпретации общезначимой формулы

p	q	$p \to q$	$(p \to q) \wedge p$	$(p \to q) \wedge p \to q$
0	0	1	0	1
0	1	1	0	1
1	0	0	0	1
1	1	1	1	1

Определение. *Формула F* ***общезначима*** *тогда и только тогда, когда она истинна во всех возможных интерпретациях.*

Таким образом, формула из примера 4.4 общезначима. Общезначимая формула называется также ***тавтологией*** (от греческого *tauto* – то же самое, *logos* – слово, мысль, речь, разум).

Формула *необщезначима* тогда и только тогда, когда она не является общезначимой.

Определение. *Формула F* ***противоречива (невыполнима)*** *то-*

гда и только тогда, когда она ложна при всех возможных интерпретациях.

Формула ***непротиворечива*** (или ***выполнима***) тогда и только тогда, когда она не является противоречивой.

Формулу, которая не является ни общезначимой, ни противоречивой (невыполнимой), называют также ***нейтральной***.

Из этих определений непосредственно следует:

- отрицание общезначимой формулы противоречиво;
- отрицание противоречивой формулы общезначимо.

Примеры 4.5.
1. $p \vee \neg p$ – формула общезначима, выполнима и непротиворечива.
2. $(p \to q) \wedge p \wedge \neg q$ – формула противоречива, невыполнима и необщезначима.
3. $p \wedge \neg p$ – формула противоречива, невыполнима и необщезначима.
4. $p \to \neg p$ – формула необщезначима и непротиворечива (нейтральна). ◄

Если формула F истинна в интерпретации I, то говорят, что I *удовлетворяет* F, или F выполнена в I.

Если формула F ложна в интерпретации I, то говорят, что I *опровергает* F, или F опровергается в интерпретации I (отсюда следует, что любая интерпретация опровергает противоречивую формулу).

Если интерпретация I удовлетворяет формуле F, то I называется также ***моделью*** F.

Ввиду конечности числа интерпретаций, путем полного перебора всех возможных интерпретаций всегда можно решить, общезначима формула или противоречива (или нейтральна).

Эквивалентность формул

Определение. *Две формулы F и G называются* ***эквивалентными*** *(или F эквивалентна G; обозначается $F = G$) тогда и только* *тогда, когда истинностные значения F и G совпадают при каждой интерпретации.*

Пример 4.6. Проверим эквивалентность формул $p \to q$ и $\neg p \vee q$:

Таблица 4.4. Эквивалентные формулы

p	q	$p \to q$	$\neg p \vee q$
0	0	1	1
0	1	1	1
1	0	0	0
1	1	1	1

Истинностные значения формул совпадают во всех интерпретациях, следовательно, $p \to q = \neg p \vee q$. ◄

Пусть F, G и H – формулы. С помощью таблиц истинности можно проверить эквивалентность формул:

1. $a) F \vee G = G \vee F;\ b) F \wedge G = G \wedge F;$

2. $a) (F \vee G) \vee H = F \vee (G \vee H);\ b) (F \wedge G) \wedge H = F \wedge (G \wedge H);$

3. $a) F \vee (G \wedge H) = (F \vee G) \wedge (F \vee H);$

$b) F \wedge (G \vee H) = (F \wedge G) \vee (F \wedge H);$

4. $a) F \vee 0 = F;\ b) F \wedge 1 = F;$

5. $a) F \vee \neg F = 1;\ b) F \wedge \neg F = 0.$

В логике эти пары эквивалентных формул часто называют законами:

1 – коммутативные законы для дизъюнкции и конъюнкции;
2 – ассоциативные законы для дизъюнкции и конъюнкции;
3 – дистрибутивные законы: дистрибутивность дизъюнкции относительно конъюнкции и дистрибутивность конъюнкции относительно дизъюнкции.
5 $a)$ – закон исключенного третьего (*tertium non datur* – третьего не дано).
5 $b)$ – закон противоречия.

Рассмотрим еще ряд эквивалентных формул, которые понадобятся нам при преобразованиях формул.

6. $a) F \vee 1 = 1$; $b) F \wedge 0 = 0$;

7. $\neg(\neg F) = F$ (закон отрицания);

8. $a) \neg(F \vee G) = \neg F \wedge \neg G$; $b) \neg(F \wedge G) = \neg F \vee \neg G$ (законы де Моргана);

9. $F \leftrightarrow G = (F \to G) \wedge (G \to F)$;

10. $F \to G = \neg F \vee G$;

11. $a) F \vee F = F$; $b) F \wedge F = F$ (идемпотентность);

12. $a) F \vee (F \wedge G) = F$; $b) F \wedge (F \vee G) = F$ (законы поглощения).

Пример 4.7. Доказать эквивалентность формул:

$(p \vee q) \wedge (p \vee r) \wedge (p \vee s) = p \vee (q \wedge r \wedge s)$.

Применяя к левой части последовательно дистрибутивный закон 3a), а затем ассоциативный закон 2 b), получим

$(p \vee (q \wedge r)) \wedge (p \vee s) = (p \vee (q \wedge r) \wedge s) = p \vee (q \wedge r \wedge s)$. ◄

Нормальные формы

Вследствие ассоциативных законов скобки в формулах $(F \vee G) \vee H$ или $(F \wedge G) \wedge H$ могут быть опущены, то есть можно писать $F \vee G \vee H$ и $F \wedge G \wedge H$.

В общем случае можно писать, не опасаясь двусмысленности

$F_1 \vee F_2 \vee \cdots \vee F_n$, где $F_1, \ldots, F_n$ – формулы.

Такая формула – *дизъюнкция формул* – истинна, когда хотя бы одна из формул F_i истинна.

Аналогично, можно написать

$F_1 \wedge F_2 \wedge \cdots \wedge F_n$.

Эта формула – *конъюнкция формул* – истинна, когда все формулы F_i истинны.

Порядок, в котором F_i встречается в конъюнкции или дизъюнкции, несуществен вследствие коммутативности операций $\wedge$ и $\vee$.

Определение. ***Литерал*** *– это атом или отрицание атома. Дизъюнкция литералов вида $p \vee \neg q \vee \cdots \vee t$ называется* ***клозом*** *или* ***дизъюнктом*** *(англ. clause – предложение). Конъюнкция литералов вида $\neg p \wedge q \cdots \wedge t$ называется* ***импликантой*** *(конъюнктой, конъюнктом, cube).*

Определение. *Формула F представлена в* ***конъюнктивной нормальной форме*** *(КНФ) тогда и только тогда, когда она имеет вид*

$$F = F_1 \wedge F_2 \wedge \cdots \wedge F_n,\ n \geqslant 1,$$

где каждая из формул F_i есть дизъюнкция литералов (клоз).

Определение. *Формула F представлена в* ***дизъюнктивной нормальной форме*** *(ДНФ) тогда и только тогда, когда она имеет вид*

$$F = F_1 \vee F_2 \vee \cdots \vee F_n,\ n \geqslant 1,$$

где каждая из формул F_i есть конъюнкция литералов (импликанта).

Приведение формул к нормальным формам

1. Используя законы

$F \leftrightarrow G = (F \to G) \wedge (G \to F)$ и $F \to G = \neg F \vee G$

устраняем логические связки $\leftrightarrow$ и $\to$.

2. Используя закон $\neg(\neg F) = F$ и законы де Моргана $\neg(F \vee G) = \neg F \wedge \neg G$, $\neg(F \wedge G) = \neg F \vee \neg G$, переносим знак отрицания непосредственно к атомам.

3. Используя нужное число раз дистрибутивные законы, приводим формулу к нужной форме.

Пример 4.8.

Привести формулу к ДНФ и КНФ: $(p \to q) \to r$.

$(p \to q) \to r = \neg(\neg p \vee q) \vee r = p \wedge \neg q \vee r$ (ДНФ) =

$= (p \vee r) \wedge (\neg q \vee r)$ (КНФ). ◀

Логическое следствие

Пусть даны формулы $F_1, F_2, \ldots, F_n$ и формула G.

Определение. *Формула G есть* ***логическое следствие*** *формул $F_1, F_2, \ldots, F_n$ (или логически следует из $F_1, F_2, \ldots, F_n$) тогда итолько тогда, когда для всякой интерпретации I, в которой $F_1 \wedge F_2 \wedge \cdots \wedge F_n$ истинна, G также истинна.*

Пример 4.9.
$F_1 : p \to q$;
$F_2 : p$;
$G : q$.

p	q	$p \to q$
0	0	1
0	1	1
1	0	0
1	1	1

В той интерпретации I, в которой $p \to q$ и p истинны, формула G тоже истинна; следовательно, G есть логическое следствие формул F_1 и F_2. ◄

Теоремы о логическом следствии

Пусть даны формулы $F_1, F_2, \ldots, F_n$ и формула G.

Теорема 4.1 *G есть логическое следствие формул $F_1, F_2, \ldots, F_n$ тогда и только тогда, когда формула*

$$F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G \ (*)$$

общезначима.

Доказательство. 1. ⇒ Пусть G – логическое следствие формул $F_1, F_2, \ldots, F_n$. Если $F_1, F_2, \ldots, F_n$ истинны в интерпретации I, то, по определению логического следствия, G истинна в I. Следовательно, формула

$$F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G \tag{4.1}$$

истинна в I.

Если же не все $F_1, F_2, \ldots, F_n$ истинны в I, т.е. хотя бы одна из них ложна в I, то $F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G$ все равно истинна в I.

Таким образом, формула $F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G$ истинна в любой интерпретации, т.е. это общезначимая формула.

2. $\Leftarrow$ Пусть $F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G$ общезначимая формула. Для всякой интерпретации I, в которой $F_1 \wedge F_2 \wedge \cdots \wedge F_n$ истинна, G также должна быть истинной. Но это и есть определение логического следствия. Теорема доказана. ☒

Теорема 4.2 *Формула G есть логическое следствие формул $F_1, F_2, \ldots, F_n$ тогда и только тогда, когда формула*

$$F_1 \wedge F_2 \wedge \cdots \wedge F_n \wedge \neg G \ (**)$$

противоречива.

Д о к а з а т е л ь с т в о . По теореме 4.1 G – логическое следствие формул $F_1, F_2, \ldots, F_n$ тогда итолько тогда, когда формула $F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G$ общезначима. Значит, G также является логическим следствием $F_1, F_2, \ldots, F_n$ тогда и только тогда, когда отрицание этой формулы противоречиво. Это, в свою очередь, значит, что формула

$$\neg(F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G) =$$
$$= \neg(\neg(F_1 \wedge F_2 \wedge \cdots \wedge F_n) \vee G) =$$
$$= \neg(\neg(F_1 \wedge F_2 \wedge \cdots \wedge F_n)) \wedge \neg G =$$
$$= F_1 \wedge F_2 \wedge \cdots \wedge F_n \wedge \neg G \tag{4.2}$$

должна быть противоречивой. ☒

Теоремы 4.1 и 4.2 являются фундаментальными в математической логике. Из них вытекает: доказательство того, что некоторая формула есть логическое следствие некоторого множества формул, эквивалентно доказательству того, что некоторая связанная с ними формула общезначима или противоречива.

Если G есть логическое следствие формул $F_1, F_2, \ldots, F_n$, то формула $(F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G)$ называется *теоремой*, $F_1, F_2, \ldots, F_n$ называются *аксиомами* (постулатами, посылками), а G – *следствием* (заключением). Тот факт, что формула G является логическим следствием формул $F_1, F_2, \ldots, F_n$ обозначается с помощью символов $\vdash$ или $\Rightarrow$:

$$F_1, F_2, \ldots, F_n \vdash G \text{ или } \vdash F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G,$$

либо

$$F_1, F_2, \ldots, F_n \ \Rightarrow \ G.$$

Существует и другое, традиционное, обозначение логического следствия:

$$\frac{F_1, F_2, \cdots, F_n}{G}.$$

В частности, если $F = G$ (формулы эквивалентны), то каждая из них является логическим следствием другой и доказательство эквивалентности является доказательством логического следствия.

Тавтология (общезначимая формула) обозначается символом $\models$. Таким образом, если G – логическое следствие формул $F_1, F_2, \ldots, F_n$, то обозначения

$$\vdash F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G \text{ и}$$

$$\models F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G$$

равносильны.

Методы доказательства теорем

Рассмотрим на примере, каким образом можно доказывать теоремы.

Пример 4.10. Пусть даны формулы $F_1 : p \to q; F_2 : \neg q; G : \neg p$.

Обозначим через F конъюнкцию $F = F_1 \wedge F_2 = (p \to q) \wedge \neg q$ и построим такие таблицы истинности :

p	q	F_1	F_2	G	$F \to G$	$F \wedge \neg G$
0	0	1	1	1	1	0
0	1	1	0	1	1	0
1	0	0	1	0	1	0
1	1	1	0	0	1	0

Метод 1. С помощью таблиц истинности показать, что формула G истинна в каждой модели формулы F (в нашем примере $(p \to q) \wedge \neg q$ и $\neg p$ истинны в интерпретации $\{\neg p, \neg q\}$), т.е. в каждой интерпретации, в которой истинна формула F.

Метод 2. С помощью таблиц истинности показать общезначимость формулы $F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G$ (в нашем примере: $(p \to q) \wedge \neg q \to \neg p$ общезначима).

Метод 3. С помощью таблиц истинности показать противоречивость формулы $F_1 \wedge F_2 \wedge \cdots \wedge F_n \wedge \neg G$ (в нашем примере: $(p \to q) \wedge \neg q \wedge p$ противоречива).

Метод 4. С помощью приведения к КНФ алгебраически доказать общезначимость формулы $F_1 \wedge F_2 \wedge \cdots \wedge F_n \to G$. Для нашего примера:

$$(p \to q) \wedge \neg q \to \neg p = \neg((\neg p \vee q) \wedge \neg q) \vee \neg p = (p \wedge \neg q) \vee q \vee \neg p =$$

$$= (p \vee q \vee \neg p) \wedge (\neg q \vee q \vee \neg p) = 1 \wedge 1 = 1.$$

Метод 5. С помощью приведения к ДНФ доказать алгебраически противоречивость формулы $F_1 \wedge F_2 \wedge \cdots \wedge F_n \wedge \neg G$. Для нашего примера:

$(p \to q) \wedge \neg q \wedge p = (\neg p \vee q) \wedge \neg q \wedge p = (\neg p \wedge \neg q \wedge p) \vee (q \wedge \neg q \wedge p) = 0 \vee 0 = 0.$

Метод резолюций

Преобразуя формулу $F_1 \wedge F_2 \wedge \cdots \wedge F_n \wedge \neg G$ из теоремы 4.2 в КНФ, получим формулу в виде конъюнкции клозов

$$C_1 \wedge C_2 \cdots \wedge C_n, \text{ где } C_i \text{ – клоз.}$$

Множество клозов определяется как

$$S = \{C_1, C_2, \ldots, C_n\}.$$

По существу это просто другая форма представления формулы – запятая между формулами заменяет конъюнкцию.

Определение. *Два литерала* L_1 *и* L_2 *называются* ***контрарными***, *если* $L_1 = \neg L_2$.

Пример 4.11.

Контрарные пары литералов: $L_1 = p$; $L_2 = \neg p$; $L_1 = \neg q$; $L_2 = q$. ◀

Пусть даны два клоза C_1 и C_2, такие, что C_1 содержит литерал L_1, а C_2 – контрарный ему литерал L_2 ($L_1 = \neg L_2$).

Определение. ***Резольвентой*** C *клозов* C_1 *и* C_2 *называется клоз* $C_1' \vee C_2'$, *где* C_1' *получен из* C_1 *отбрасыванием* L_1, *а* C_2' *получен из* C_2 *отбрасыванием* L_2.

Примеры 4.12.

1. $C_1 : p \vee r$, $C_2 : \neg p \vee q$; $C : r \vee q$.
2. $C_1 : r \vee q \vee \neg p$, $C_2 : p \vee s$; $C : r \vee q \vee s$.
3. $C_1 : \neg p \vee q$, $C_2 : \neg p \vee r$ – резольвенты не существует. ◀

Важным свойством резольвенты является то, что любая резольвента двух клозов C_1, C_2 есть логическое следствие C_1 и C_2.

Теорема 4.3 *Резольвента* C *клозов* C_1 *и* C_2 *является их логическим следствием.*

Доказательство. Пусть $C_1 = C_1' \vee L$, $C_2 = C_2' \vee \neg L$. $C = C_1' \vee C_2'$, где C_1', C_2' – клозы. Нужно доказать, что C есть логическое следствие C_1 и C_2, т.е. что

$(C_1' \vee L) \wedge (C_2' \vee \neg L) \to (C_1' \vee C_2')$ – тавтология.

Пусть C_1 и C_2 истинны в некоторой интерпретации I. Нужно доказать, что C также истинный в I. Отметим, что либо L, либо$\neg L$ ложно в I.

Если $L = 0$, то $C_1' = 1$ и $(C_1' \vee C_2') = 1$.

Если $\neg L = 0$, то $C_1' = 1$ и $(C_1' \vee C_2') = 1$.

Другое, более формальное доказательство: докажем противоречивость формулы

$(C_1' \vee L) \wedge (C_2' \vee \neg L) \wedge \neg(C_1' \vee C_2')$

$(C_1' \vee L) \wedge (C_2' \vee \neg L) \wedge \neg C_1' \wedge \neg C_2' =$
$= (C_1' \wedge C_2' \vee C_1' \wedge \neg L \vee C_2' \wedge L \vee L \wedge \neg L)\neg C_1' \wedge \neg C_2' = 0.$ ⊠

Определение. *Резольвента двух контрарных литералов $\{L, \neg L\}$, которая соответствует выводу 0 (формулы "ложь"), называется* ***пустым*** *клозом и обозначается* $\square$.

Если имеются два единичных клоза, то их резольвента, если она существует, есть пустой клоз. Более существенно то, что для невыполнимого множества клозов применение правил резолюции могут породить $\square$.

Пусть S – множество клозов.

Определение. ***Резолютивный вывод*** *C из S есть такая конечная последовательность $C_1, C_2, \ldots, C_N$ клозов, что каждый C_i или принадлежит S, или является резольвентой клозов, предшествующих C_i и $C_N = C$.*

Два клоза, из которых выведена резольвента C_i, называются *родительскими* клозами. Вывод $\square$ из S называется *опровержением* или *доказательством невыполнимости* S. Говорят, что клоз C может быть *выведен* из S, если существует вывод C из S.

Примеры 4.13.

1. $S = \{\neg p \vee q, \neg q, p\}$;

$C_1 : \neg p \vee q$,

$C_2 : \neg q$,

$C_3 : p$,

$C_4 : \neg p$(из C_1 и C_2),

$C_5 = \square$ (из C_3 и C_4).

2. $S = \{p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q\}$;

$C_1 : p \vee q$,

$C_2 : \neg p \vee q$,

$C_3 : p \vee \neg q$,

$C_4 : \neg p \vee \neg q$,

$C_5 : q$ (из C_1 и C_2),

$C_6 : \neg q$ (из C_3 и C_4),

$C_7 : \square$ (из C_5 и C_6). ◄

Теорема 4.4 (о полноте резолютивного вывода) *Множество клозов S противоречиво тогда и только тогда, когда существует резолютивный вывод пустого клоза $\Box$ из S.*

Доказательство. 1. $\Leftarrow$ Обозначим через S' множество клозов, состоящее из S и клозов, выведенных из S на основе метода резолюций. Покажем, что если S' противоречиво, то и S противоречиво.

Действительно, если S' противоречиво, то для всякой интерпретации I найдется такой клоз $C' \in S'$, что $C' = 0$.

Если $C' \in S$, то $S = C_1 \wedge C_2 \wedge \cdots \wedge C_k$ также принимает значение 0.

Если $C' \notin S$, то C' – резольвента, выведенная из S, например, из клозов $C_i, C_j \in S$. Тогда $C_i, C_j \vdash C'$, или формула $C_i \wedge C_j \rightarrow C'$ общезначима. Но так как C' ложно, то либо $C_i = 0$, либо $C_j = 0$. Таким образом, для любой интерпретации I найдется ложный клоз в S, следовательно, S противоречиво.

Теперь предположим, что из S выведен пустой клоз $\Box$. Докажем, что в этом случае множество S' противоречиво, а значит и S противоречиво (т.е. существует вывод $\Box$ из S).

Пусть $R_1, R_2, \ldots R_l$ – резольвенты в выводе. Предположим, что S' выполнимо в некоторой интерпретации I. Однако, если в этой интерпретации C_i и C_j выполнимы, то и любая их резольвента выполнима в этой интерпретации. Значит и резольвенты $R_1, R_2, \ldots R_l$ выполнимы в I.

Это, однако, невозможно, т.к. одна из резольвент есть $\Box$. Поэтому S' – невыполнимо (противоречиво), а значит, и S противоречиво.

2. $\Rightarrow$ Пусть теперь множество S противоречиво. Покажем, что существует резолютивный вывод $\Box$ из S.

Доказательство проводится индукцией по числу $k(S)$ "лишних"литералов. Пусть $S = \{C_1, \ldots, C_n\}$, а $|L_i|$ – число литералов в C_i. Тогда

$$k(S) = \sum_{i=1}^{n} |L_i| - n.$$

Базис индукции. Положим $k(S) = 0$. При этом возможны два случая.

1. Множество S состоит из единичных клозов, т.е. каждый C_i содержит по одному литералу. Так как S противоречиво, то оно должно содержать контрарную пару клозов (иначе возможна интерпретация, в которой S не ложно). Резольвента контрарной пары дает пустой клоз $\square$.

2. Множество S содержит хотя бы один клоз, состоящий из двух литералов. Но тогда S обязано содержать и пустой клоз (иначе оно не будет противоречивым).

Базис индукции доказан.

Шаг индукции. Предположим, что теорема справедлива для $k(S) < n$ и докажем ее для $k(S) = n$.

Пусть $\square \notin S$. Так как $k(S) > 0$, то в S найдется клоз $C = C_i \vee L$, где L – литерал.

Пусть $S = S' \cup \{C_i \vee L\}$, где $S' = S \setminus \{C\}$.

Обозначим

$$S_1 = S' \cup \{C_i\} \text{ и } S_2 = S' \cup \{L\}.$$

Покажем,что если S противоречиво, то S_1 и S_2 также противоречивы. Пусть I_1 – множество интерпретаций, в которых C истинно.

Так как S противоречиво, то S' содержит хотя бы один ложный клоз для каждой $I \in I_1$. Но тогда и S_1 и S_2 ложны для каждой $I \in I_1$.

Пусть I_0 – множество интерпретаций, в которых C ложен. Тогда для каждой $I \in I_0$ $C_i = 0$ и $L = 0$, следовательно, ложны и S_1 и S_2.

Таким образом, S_1 и S_2 ложны в любых интерпретациях, т.е. противоречивы.

Так как в S_1 и в S_2 одинаковое число клозов и столько же , сколько в S, S_1 содержит на один литерал меньше, а S_2 содержит по крайней мере на один литерал меньше, чем S, то

$$k(S_2) \leq k(S_1) < k(S) = n.$$

Следовательно, по предположению индукции, из S_1 и из S_2 может быть выведен пустой клоз $\Box$.

Обозначим через $R^l(S)$ множество клозов, полученных в результате применения к S l шагов резолютивного вывода. Мы показали, что существуют такие i и j, что

$$\Box \in R^i(S_1), \ \Box \in R^j(S_2).$$

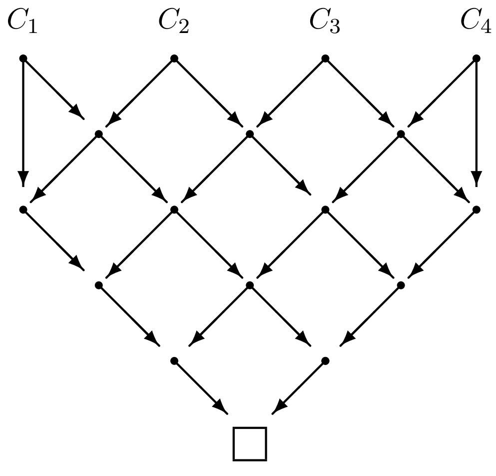
Пусть i шагов резолюции, которые привели к выводу $\Box$ из S_1 применены к $S = S' \cup \{C_i \vee L\}$. При этом будет получен либо пустой клоз $\Box$ (и тогда теорема доказана), либо L.

Если выведено L, то, так как $S' \subseteq S \subseteq R^i(S)$ и $L \in R^i(S)$, то и $S_2 = S' \cup \{L\} \subseteq R^i(S)$.

Теперь для вывода $\Box$ можно к S_2 применить, как к подмножеству $R^i(S)$, j шагов резолюции, которые, как мы показали, существуют. $\boxtimes$

Стратегии очищения Применение резолютивного вывода без ограничений приводит к большому числу клозов и существенному снижению эффективности доказательства теоремы.

Процесс резолютивного вывода пустого клоза можно представить в виде графа опровержения.



Каждая вершина графа (кроме исходных) представляет клоз, полученный в результате применения резолюции к двум *родительским* клозам (степени захода равны 2 или 0).

Для применения резолютивного вывода на практике используют различные стратегии отбора клозов для вывода – *стратегии очищения*.

Синтаксические стратегии производят отбор клозов для резолюции только на основе их структурных свойств, таких, например, как число содержащихся в них литералов. При этом в расчет не принимаются ни истинные, ни ложные интерпретации этих клозов.

Семантические стратегии, наоборот, учитывают в первую очередь истинность клозов в некоторых интерпретациях.

Стратегии, *учитывающие ход вывода* предполагают отбор клозов на основе хода уже осуществленного вывода. Например, в стратегии *исходных данных* в каждой паре клозов, из которых выводится резольвента, по крайней мере один клоз должен принадлежать исходному множеству S.

Примеры стратегий

Стратегия предпочтения одночленов.

Если клозы C_1 и C_2, содержащие, соответственно, m и n литералов, можно разрешить друг с другом, то их резольвента содержит не более $(m-1)+(n-1)$ литералов. Если в одном из них, например C_1, содержится только один литерал, то резольвента будет содержать $n-1$ литералов. Так как цель вывода – получить пустой клоз (не содержащий литералов), то это шаг в нужном направлении.

Стратегия предпочтения одночленов предполагает отбор в первую очередь одночленов (клозов с одним литералом). В более общем случае, сначала отбираются более короткие клозы, а затем уже более длинные. Из рассмотрения не исключается ни один клоз, а лишь устанавливается порядок применения резолюций.

Это чисто синтаксическая стратегия.

Исключение тавтологий и уникальных литералов.

Эта семантическая стратегия предлагает удалять некоторые клозы из S до их участия в выводе. При этом, конечно, оставшееся после удаления множество $S^\star$ должно оставаться противоречивым.

Удаление тавтологий не нарушает противоречивости S. Если, кроме того, некоторый клоз содержит уникальный литерал L, для которого нет контрарного, то его тоже можно удалить, т.к. никакая резолюция с ним не приведет к $\Box$.

Линейный вывод

Это пример стратегии, учитывающей ход вывода. Пусть C_0 – любой клоз противоречивого множества S, $C_i,\ i > 0$ – любой i-й клоз, выведенный последовательностью резолюций, начавшейся с C_0 и некоторого другого клоза из S.

Если i-й клоз всегда имеет в качестве одного из родительских клозов (левый) $(i-1)$-й клоз вывода, то вывод называется *линейным*.

Пример 4.14.

$$S = \{q, p \vee r, \neg p \vee s, \neg r \vee s, \neg p \vee \neg q \vee \neg s, \neg q \vee \neg r \vee \neg s\}.$$

Один из возможных выводов представлен на диаграмме. Горизонтальные стрелки на диаграмме показывают, из каких клозов выводится очередная резольвента; наклонные стрелки показывают выбор левого родительского клоза.

За начальный клоз выбран $C_0 :\ (\neg p \vee \neg q \vee \neg s)$; затем в качестве левого родительского клоза всегда выбирается резольвента, полученная на предыдущем шаге вывода; правым родительским клозом может быть любой клоз из множества S', состоящего из исходного множества S и резольвент, полученных на предыдущих шагах вывода. ◄

Открытым остается вопрос о том, как найти линейный вывод, который приведет к $\Box$ и, желательно, наиболее коротким путем.

Глава 5

Логика предикатов

Термы, предикаты, формулы

В логике высказываний атом представляет собой повествовательное (утвердительное) предложение, которое может быть либо истинным, либо ложным, но не то и другое вместе. Атом рассматривается как единое целое и его структура и состав не анализируются. Есть, однако, рассуждения которые не могут быть представлены столь простым способом. Классическое умозаключение (*силлогизм*), которое часто используется в качестве иллюстрации предикатов первого порядка

все люди смертны
Сократ – человек
Сократ смертен

не может быть доказано в логике высказываний, хотя это рассуждение интуитивно корректно. Распишем это рассуждение через высказывания

p: *все люди смертны*
q: *Сократ – человек*
r:*Сократ смертен.*

r не является логическим следствием в рамках логики высказываний, так как $p \wedge q \to r$ не является общезначимой формулой. Причина этого заключается в том, что в логике высказываний не учитывается структура атомов p, q, r.

В логике предикатов первого порядка понятие атома расширяется. Вводятся еще три логических понятия: *термы*, *предикаты* и *кванторы*.

Для построения атомов используются четыре типа символов:

1. *Индивидные* символы или *константы*.

2. *Символы предметных* переменных или *переменные*. Для обозначения переменных будем использовать строчные буквы латинского алфавита, возможно с индексами (t, u, v, x, y и другие).

3. *Функциональные символы* (*функции* или *функторы*). Для обозначения функторов будем использовать строчные буквы из ряда f, g, h или слова из строчных букв.

4. *Предикатные символы* (*предикаты*). Для обозначения предикатов будем использовать прописные буквы из ряда P, Q, R, или слова, начинающиеся с прописной буквы.

Предполагается, что задана область определения D функторов и предикатов (*домен, domain*), которая в логике называется *предметной* областью, или областью *интерпретации*.

Константы и переменные представляют элементы области D. Функция (функтор) – это отображение списка констант в константу из D.

1. Константы: $a, john$, "Волга впадает в Каспийское море 0.7.

2. Переменные: x, y.

3. Функции (функторы): $divide(1, x)$ – результат деления 1 на x; $plus(x, 3)$ – соответствует $x + 3$.

Если функциональный символ (функтор) имеет n аргументов, то он называется n-местным функциональным символом.

4. Предикаты: $Parent(x, y)$ – x является родителем y.

Предикаты (предикатные символы) отображают список констант в истинностные значения "истина"или "ложь". Это отображение осуществляется в процессе интерпретации, т.е.

установлении соответствия между элементами предметной области и значением предиката.

Весь спектр возможных аргументов предиката охватывается понятием *терм*. Интуитивно терм означает объект. Дадим точное определение терма.

Определение. *Терм определяется рекурсивно следующим образом:*

1. Константа есть терм.

2. Переменная есть терм.

3. Если f – n-местный функциональный символ и $t_1, \ldots, t_n$ – термы, то $f(t_1, \ldots, t_n)$ – терм.

4. Термы порождаются только конечным числом применений правил 1–3.

Замечания. 1. Индивидный символ или константу можно рассматривать как функциональный символ без аргументов.

2. Определение терма включает и рекурсивные функторы, например, $plus(plus(x, 1), x)$. ►

Теперь дадим точное определение атома.

Определение. *Если P – n-местный предикатный символ и $t_1, \ldots, t_n$ – термы, то $P(t_1, \ldots, t_n)$ – атом.*

Замечание. Можно считать, что предикат с одной переменной (одноместный предикат) выражает свойство объекта (терма). Например: x есть простое число, y есть студент и т.д.

Многоместный предикат выражает отношение между объектами, например,

Любит(Паниковский, жареных, гусей)

Любит(Бендер, холодное, пиво)

Заметим также, что имеется несколько вариантов представления некоторого факта или утверждения об отношении на языке логики предикатов первого порядка. Например, утверждение “солдатик оловянный"можно представить такими атомами:

Оловянный(солдатик);

Материал(солдатик,оловянный);

Свойство(материал,солдатик,оловянный). ▶

Для построения формул исчисления предикатов используются *логические связки* и *кванторы*. Логические связки $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ служат для образования выражений более сложных, чем атомы – формул.

Кванторы описывают количественные соотношения внутренней структуры логического выражения.

Кванторы естественного языка: для всех элементов, почти для всех, ровно для одного, ровно для трех, существует ровно один, существует хотя бы один и т.д.

В логике предикатов используются

- квантор *всеобщности* (универсальный квантор)

  $(\forall x)$ – для всех x, для каждого x;

- квантор *существования* (экзистенциальный квантор) $(\exists x)$ – существует x, по крайней мере для одного x, для некоторых x.

Примеры 5.1.

$(\forall x)[\textit{Молодой}(x) \wedge \textit{Учится}(x) \rightarrow \textit{Студент}(x)]$;

$(\exists y)\textit{Больше}(x, y)$;

$(\forall x)[(\textit{Человек}(x) \rightarrow \textit{Смертен}(x)) \wedge \textit{Человек(Сократ)}$

$\rightarrow \textit{Смертен(Сократ)}]$. ◀

Связанные и свободные переменные

Область действия квантора, входящего в формулу – это формула, к которой этот квантор применяется. Например, для формулы

$$(\forall x)(\exists y)\textit{Меньше}(x, y)$$

область действия квантора существования есть формула $\textit{Меньше}(x$
а область действия квантора всеобщности – $(\exists y)\textit{Меньше}(x, y)$.

Определение. *Переменная, входящая в формулу, к которой применяется квантор (в область действия квантора), называ-*

ется ***связанной****. В противном случае переменная называется* ***свободной****.*

Пример 5.2.

$(\forall x)P(x, y)$: переменная x связана (квантором), а переменная y свободна. ◀

Заметим, что переменная в формуле может быть свободной и связанной одновременно.

Пример 5.3.

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

Переменная y свободна в $(\forall x)P(x, y)$ и связана в $(\forall y)Q(y)$ ◀ В формулах $(\forall x)P(x)$ или $(\exists x)P(x)$ кванторы *связывают* переменную в том смысле, что хотя в записи формул встречается символ x, обозначающий переменную, обе эти формулы представляют именно *высказывания*, а не предикаты (пропозициональные функции). Значит, эти формулы от переменной больше не зависят. Присутствие переменной в формуле указывает лишь из какого предиката это высказывание образовано. Фактически, в этих формулах x не является переменной.

Рассмотрим подробнее смысл высказываний $(\forall x)P(x)$ и $(\exists x)P(x)$. Пусть предметная область есть $D = \{a, b, c\}$. Тогда высказывание $(\forall x)P(x)$ равносильно сложному высказыванию $P(a) \wedge P(b) \wedge P(c)$. Хотя квантор общности (универсальный квантор) можно использовать и относительно бесконечных множеств, в практических целях удобно считать, что квантор $\forall$ является *обобщением конъюнкции* конечного числа единичных высказываний.

Подобно квантору общности квантор существования (экзистенциальный квантор) $\exists$ тоже обобщает, но обобщает операцию *дизъюнкции*. Для той же предметной области D высказывание $(\exists x)P(x)$ равносильно высказыванию $P(a) \vee P(b) \vee P(c)$.

В высказываниях могут встретиться оба типа кванторов. Рассмотрим, например, высказывание типа "у каждого (человека) (есть) свои причуды". Пусть $D = A \times B$, где $A = \{a, b, c\}$ – множество людей, а $B = \{s, t\}$ – множество причуд; предикат

$R(x, y)$ обозначает, что у человека x есть (существует) причуда y. Тогда высказывание "у каждого свои причуды" можно записать в виде:

$$(\forall x \in A)(\exists y \in B)\ R(x, y) = (\forall x \in A)[(R(x, s) \vee R(x, t)] =$$
$$= [R(a, s) \vee R(a, t)] \wedge [R(b, s) \vee R(b, t)] \wedge [R(c, s) \vee R(c, t)].$$

Другой пример. Пусть $D = A \times B$, где $A = \{2, 5\}, B = \{3, 4, 5\}$. Высказывание $(\exists x \in A)(\forall y \in B)[y > x]$ равносильно высказыванию

$$(\forall y \in B)[y > 2] \ \vee \ (\forall y \in B)[y > 5] =$$
$$= [(3 > 2) \wedge (4 > 2) \wedge (5 > 2)] \vee [(3 > 5) \wedge (4 > 5) \wedge (5 > 5)].$$

Дадим теперь строгое определение формулы применением атомов, логических связок и кванторов.

Определение. *Правильно построенные формулы (**ППФ**) логики первого порядка рекурсивно определяются следующим образом.*

1. Атом есть формула.

2. Если F и G формулы, то $\neg F$, $(F \wedge G)$, $(F \vee G)$, $F \rightarrow G$, $F \leftrightarrow G$ – формулы.

3. Если F формула, а x – свободная переменная в F, то $(\forall x)F$ и $(\exists x)F$ – формулы.

4. Формулы порождаются только конечным числом применений правил 1, 2, 3.

З а м е ч а н и я. 1. Как и в логике высказываний, круглые скобки можно заменять квадратными и фигурными, а можно опускать совсем, когда это позволяет контекст. Скобки могут быть опущены в соответствии с приоритетом связок. Расширим эти приоритеты – будем считать, что кванторы имеют самый низший приоритет. Поэтому можно писать $\forall x P(x)$, $\exists x P(x)$ (полагая по умолчанию, что $x \in D$).

2. Термины "функция (функтор)" и "предикат" являются синонимами терминов "функциональная форма" и "предикатная форма", соответственно.

3. Термин "логика предикатов первого порядка" или "исчисление предикатов первого порядка" означает, что в формулах действие квантора распространяется только на термы.

В системах второго порядка действие кванторов может распространяться и на сами предикаты, например, $\exists P \forall x P(x)$ (существуют такие предикаты P, что для всех $x \in D$ выполняется $P(x)$). Здесь P выступает в роли переменной-предиката.

Если рассматривать термы как объекты, а предикаты как свойства или отношения, то исчисление предикатов второго и более высоких порядков описывает

- свойства свойств;
- свойства отношений;
- отношения между свойствами;
- отношения между отношениями;
- отношения между свойствами и отношениями. ►

5.1 Интерпретация формул в логике предикатов первого порядка

Формулы исчисления предикатов, как и формулы исчисления высказываний, получают истинностные значения в процессе *интерпретации*.

Интерпретация устанавливает соответствия между элементами предметной области (области интерпретации) и значениями констант, функциональных и предикатных символов.

Определение. *Интерпретация формулы F логики первого порядка состоит из непустой предметной области и указания значений всех констант, функциональных и предикатных символов, встречающихся в F.*

При этом:

1. Каждой константе ставится в соответствие некоторый элемент из D.

2. Каждому n-местному функциональному символу $f(x_1, \ldots, x_n)$ ставится в соответствие отображение $f : D^n \mapsto D$, где

$D^n = \{\langle x_1, \ldots, x_n \rangle\} | x_1 \in D, \ldots, x_n \in D\}$.
3. Каждому n-местному предикатному символу $P(x_1, \ldots, x_n)$ ставится в соответствие отображение $P : D^n \mapsto \{1, 0\}$.

Следует подчеркнуть, что интерпретация проводится именно на предметной области D. Когда определяется истинностное значение кванторов на D $\forall x$ следует интерпретировать как "для всех элементов x из D", а $\exists x$ – как "существует элемент x из D".

Для каждой интерпретации формулы на области D формула может получить истинностное значение "истина" или "ложь" по следующим правилам.
1. Если заданы истинностные значения формул F и G, то истинностные значения формул $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \to G)$, $(F \leftrightarrow G)$ интерпретируются с помощью таблиц истинности этих связок.
2. Формула $\forall x F(x)$ получает значение "истина", если $F(x)$ имеет значение "истина" для каждого x из D; в противном случае она получает значение "ложь".
3. Формула $\exists x F(x)$ получает значение "истина", если $F(x)$ имеет значение "истина" хотя бы для одного x из D; в противном случае она получает значение "ложь".

З а м е ч а н и е. Формула, содержащая свободные переменные не может быть интерпретирована (получить истинностное значение). Далее будем считать, что формула либо не содержит свободных переменных, либо свободные переменные рассматриваются как константы. ▶

Пример 5.4.
Определить истинностное значение формулы

$$\forall x[P(x) \to Q(f(x), a)]$$

для следующей интерпретации:

область $D = \{1, 2\}$;

значения для a: $a = 1$;

значения для f: $f(1) = 2, f(2) = 1$;

значения для P и Q: $P(1) = 0; P(2) = 1$.

$Q(1,1) = 1; Q(1,2) = 1; Q(2,1) = 0; Q(2,2) = 1.$

В этой интерпретации формулу можно записать в виде

$[P(1) \to Q(f(1),1)] \wedge [P(2) \to Q(f(2),1)] =$

$= [P(1) \to Q(2,1)] \wedge [P(2) \to Q(1,1)] = (0 \to 0) \wedge (1 \to 1) = 1 \wedge 1 = 1,$

т.е. формула истинна в заданной интерпретации. ◄

Как только определены интерпретации, все понятия логики высказываний можно определить и для логики первого порядка. Если формула в логике первого порядка не содержит переменных и кванторов, ее можно рассматривать как формулу в логике высказываний.

Формулу, истинную при всех возможных интерпретациях (как и в логике высказываний) будем называть *общезначимой* (*тавтологией*).

Формула, ложная при всех возможных интерпретациях, называется *противоречивой* (*невыполнимой*).

Формула *непротиворечива* (*выполнима*), если существует интерпретация I, в которой формула имеет значение "истина".

Определение. *Формула G есть **логическое следствие** формул $F_1, \ldots, F_n$ тогда и только тогда, когда для каждой интерпретации I, в которой $F_1 \wedge \cdots \wedge F_n$ истинна в I, G также истинна.*

Теоремы 4.1 и 4.2 о логическом следствии справедливы и в логике первого порядка.

Так как в логике первого порядка имеется бесконечное множество областей интерпретации (и сами области могут быть бесконечными), то, вообще говоря, имеется бесконечное число интерпретаций формулы.

5.2 Предваренная нормальная форма

Определение. *Говорят, что формула F представлена в **предваренной** нормальной форме (ПНФ), если F имеет вид*

$\rangle\!|_1 x_1 \rangle\!|_2 x_2 \ldots \rangle\!|_n x_n M[x_1, \ldots, x_n],$

где каждое $\rangle\!|x_i, i=\overline{1,n}$ есть либо $\forall x_i$, либо $\exists x_i$, а $M[x_1,\ldots,x_n]$ – формула, не содержащая кванторов (***матрица*** формулы F), $\rangle\!|_1 x_1 \rangle\!|_2 x_2 \ldots \rangle\!|_n x_n$ – *префикс* формулы F.

Примеры 5.5.
Формулы, представленные в ПНФ.

1) $\forall x \forall y [P(x,y) \wedge Q(y)]$;

2) $\forall x \forall y [\neg P(x,y) \rightarrow Q(y)]$;

3) $\forall x \forall y \exists z [Q(x,y) \rightarrow R(z)]$. ◀

Напомним, что две формулы F и G эквивалентны ($F = G$), когда истинностные значения F и G одинаковы в любой интерпретации. Основные пары эквивалентных формул (с логическими связками) логики высказываний справедливы и в логике первого порядка.

Рассмотрим эквивалентные формулы, содержащие кванторы. Это понадобится нам для преобразования формулы в предваренную нормальную форму.

Пусть $F[x]$ формула, содержащая свободную переменную x; G – формула, в которую не входит свободная переменная x; $\rangle\!|$ – квантор $\forall$ или $\exists$.

Каждую пару эквивалентных формул будем называть для краткости *законом*:

1. $a) \rangle\!|x F[x] \vee G = \rangle\!|x(F[x] \vee G)$;

   $b) \rangle\!|x F[x] \wedge G = \rangle\!|x(F[x] \wedge G)$;

2. $a) \neg(\exists x F[x]) = \forall x(\neg F[x])$;

   $b) \neg(\forall x F[x]) = \exists x(\neg F[x])$.

Законы $1a$) и $1b$) истинны, т.к. G не содержит переменной x и может быть внесена в область действия квантора.

Докажем $2a$). Пусть I некоторая интерпретация с областью D.

Если $\neg(\exists x F[x])$ истинна в I, то $(\exists x F[x])$ – ложна в I. Это означает, что в D нет такого элемента, для которого $F[x]$ истинна в I; следовательно, для любого элемента в D $F[x]$ ложна в I, или $\neg F[x]$ истинна в I, т.е. справедливо, что формула $(\forall x \neg F[x])$ истинна в I.

Пусть теперь $\neg(\exists x F[x])$ ложна в I. Тогда $\exists x F[x]$ истинна в I, т.е. $\exists e \in D$ (в D существует некоторый элемент e), такой, что $F[e]$ истинна в I, или $\neg F[x]$ – ложна в I. Но тогда формула $\forall x(\neg F[x])$ ложна в I. Таким образом, формулы принимают одни и те же истинностные значения в любой интерпретации, т.е. эквивалентны.

Докажем теперь $2b)$. Пусть I – произвольная интерпретация с областью D.

Если $\neg(\forall x F[x])$ истинна в I, то $\forall x F[x]$ ложна в I. Это значит, что существует элемент $e \in D$, что $F[e]$ ложна, т.е. $\neg F[e]$ истинна. Следовательно, $\exists x(\neg F[x])$ – истинна в I.

Пусть теперь $\neg(\forall x F[x])$ ложна в I. Тогда $\forall x F[x]$ истинна в I. Это значит, что $F[x]$ истинна для каждого $x \in D$, т.е. $\neg F[x]$ ложна для каждого $x \in D$, следовательно, $\exists x(\neg F[x])$ ложна в I. Таким образом, формулы принимают одинаковые истинностные значения в любой интерпретации, т.е. эквивалентны.

Для конечных областей $D = \{a_1, \dots, a_k\}$ законы $2a), 2b)$ можно легко доказать используя законы де Моргана.
Для закона $2a)$:

$$\neg(\exists x F[x]) = \neg(F[a_1] \vee \dots \vee F[a_k]) =$$
$$= \neg F[a_1] \wedge \dots \wedge \neg F[a_k] = \forall x(\neg F[x]).$$

Для закона $2b)$:

$$\neg(\forall x F[x]) = \neg(F[a_1] \wedge \dots \wedge F[a_k]) =$$

$$= \neg F[a_1] \vee \dots \vee \neg F[a_k]) = \exists x(\neg F[x]).$$

3. $a)\forall x F[x] \wedge \forall x H[x] = \forall x(F[x] \wedge H[x])$,

$b)\exists x F[x] \vee \exists x H[x] = \exists x(F[x] \vee H[x])$,

т.е. кванторы $\forall$ и $\exists$ можно распределять по $\wedge$ и $\vee$, соответственно. Однако $\forall$ и $\exists$ *нельзя* распределять по $\vee$ и $\wedge$, соответственно, т.е.

1)$\forall x F[x] \vee \forall x H[x] \neq \forall x(F[x] \vee H[x])$,

2)$\exists x F[x] \wedge \exists x H[x] \neq \exists x(F[x] \wedge H[x])$,

Докажем, например, 1).

Пусть I –произвольная интерпретация с областью D и формула $\forall x(F[x] \vee \forall x H[x])$ истинна в I. Выберем конкретный элемент $e_1 \in D$ и предположим, что $F[e_1]$ истинна, а $H[e_1]$ – ложна. Далее, выберем элемент $e_2 \in D$ и предположим, что $F[e_2]$ – ложна, а $H[e_2]$ – истинна в I. Оба этих варианта не нарушают предположения об истинности $\forall x(F[x] \vee H[x])$ в I.

Но тогда $\forall x H[x]$ ложна (т.к. $H[e_1]$ ложна) в I, и $\forall x F[x]$ ложна (т.к. $F[e_2]$ ложна) в I.

Таким образом, мы нашли интерпретацию, в которой $\forall x(F[x] \vee H[x])$ истинна, а $\forall x F[x] \vee \forall x H[x]$ ложна, т.е. формулы неэквивлентны.

Возникает вопрос, нельзя ли все-таки вынести в префикс кванторы и в этих случаях? Оказывается – можно.

Каждую связанную переменную в формуле можно рассматривать лишь как место для подстановки какой угодно переменной (связанная переменная – “немая"). Это значит, что каждую связанную переменную можно переименовать. Заменим, например, x на z и формула $\forall x H[x]$ преобразуется в $\forall z H[z]$. Предположим, что мы выбрали z, которая не встречается в $F[x]$. Тогда

$$\forall x F[x] \vee \forall x H[x] = \forall x F[x] \vee \forall z H[z]$$

А теперь применим к каждой переменной правило 1. вынесения квантора за скобки. Получим

4. $\forall x F[x] \vee \forall x H[x] = \forall x \forall z(F[x] \vee H[z])$.

Аналогично,

5. $\exists x F[x] \wedge \exists x H[x] = \exists x F[x] \wedge \exists z H[z] =$

$= \exists x \exists z(F[x] \wedge H[z])$.

В общем случае

6. $\rangle\!|_1 x F[x] \vee \rangle\!|_2 x H[x] = \rangle\!|_1 x \rangle\!|_2 z(F[x] \vee H[z])$.

7. $\rangle\!|_3 x F[x] \wedge \rangle\!|_4 x H[x] = \rangle\!|_3 x \rangle\!|_4 z(F[x] \wedge H[z])$.

5.2.1 Приведение формул к предваренной нормальной форме.

Любую формулу можно преобразовать в ПНФ, если использовать следующую последовательность эквивалентных преобразований.

1. Для исключения связок $\leftrightarrow$ и $\rightarrow$ применяем законы

$$F \leftrightarrow G = (F \rightarrow G) \wedge (G \rightarrow F),$$

$$F \rightarrow G = \neg F \vee G.$$

2. Для того, чтобы отрицание относилось непосредственно к атому, применяем законы

$\neg(\neg F) = F$ (двойное отрицание);

$$\left.\begin{array}{l} \neg(F \vee G) = \neg F \wedge \neg G \\ \neg(F \wedge G) = \neg F \vee \neg G \end{array}\right\} \text{(законы де Моргана);}$$

$$\left.\begin{array}{l} \neg(\forall x F[x]) = \exists x(\neg F[x]) \\ \neg(\exists x F[x]) = \forall x(\neg F[x]) \end{array}\right\} [2a), 2b)].$$

3. Переименовываем связанные переменные, если это необходимо.

4. Используем законы вынесения кванторов в префикс.

$$\rangle\!| x F[x] \vee G = \rangle\!| x(F[x] \vee G),$$

$$\rangle\!| x F[x] \wedge G = \rangle\!| x(F[x] \wedge G),$$

$$\forall x F[x] \wedge \forall x H[x] = \forall x(F[x] \wedge H[x]),$$

$$\exists x F[x] \vee \exists x H[x] = \exists x(F[x] \vee H[x]),$$

$$\rangle\!|_1 x F[x] \vee \rangle\!|_2 x H[x] = \rangle\!|_1 x \rangle\!|_2 z(F[x] \vee H[z]),$$

$$\rangle\!|_3 x F[x] \wedge \rangle\!|_4 x H[x] = \rangle\!|_3 x \rangle\!|_4 z(F[x] \wedge H[z]).$$

З а м е ч а н и я. 1. Описанный процесс приведения к ПНФ одновременно служит конструктивным способом доказательства существования ПНФ.

2. Напомним, что в процессе приведения к ПНФ мы выполняли эквивалентные преобразования, поэтому формула, представленная в ПНФ, эквивалентна исходной формуле.

3. Предваренная нормальная форма не единственна для формулы. Результат зависит от порядка применения эквивалентных преобразований, а также от произвола при переименовании. Эквивалентны, например, следующие ПНФ:

$\forall x(P(x) \wedge Q(x))$ и $\forall x \forall y(P(x) \wedge Q(y))$. ►

Пример 5.6.
Преобразовать в ПНФ формулу $\forall x P(x) \to \exists x Q(x)$.

$$\forall x P(x) \to \exists x Q(x) = \neg(\forall x P(x)) \vee \exists x Q(x) =$$

$$= \exists x(\neg P(x)) \vee \exists x Q(x) \;=\; \exists x(\neg P(x) \vee Q(x)) =$$

$$= \exists x(P(x) \to Q(x)).$$ ◄

Очевидны следующие соотношения для кванторов (почему?):

$$\forall x \forall y P(x,y) = \forall y \forall x P(x,y);$$

$$\exists x \exists y P(x,y) = \exists y \exists x P(x,y);$$

$$\exists x \forall y P(x,y) \neq \forall y \exists x P(x,y).$$

5.3 Стандартная нормальная форма

Для исчисления высказываний мы приводили формулы к КНФ и представляли КНФ в виде множества клозов. Затем задача логического вывода сводилась к выводу пустого клоза $\Box$ из множества клозов. Нечто подобное мы хотим сделать и с формулами исчисления первого порядка – привести к некоторой *стандартной нормальной форме* (СНФ), а затем представить в виде множества клозов.

Процесс приведения произвольной формулы к СНФ сводится к следующим действиям.

1. Приведение формулы к ПНФ.

2. Приведение матрицы M формулы (в ПНФ) к конъюнктивной нормальной форме (КНФ). Матрица M представлена в КНФ, если она записана в виде $M = C_1 \wedge C_2 \wedge \cdots \wedge C_n$, где C_i

– клоз. Приведение матрицы к КНФ достигается применением нужного числа раз дистрибутивного закона $F \vee (G \wedge H) = (F \vee G) \wedge (F \vee H)$.

3. *Сколемизация* – исключение из формулы кванторов существования $\exists$.

Пусть формула F представлена в ПНФ:

$\rangle\!|_1 x_1 \rangle\!|_2 x_2 \ldots \rangle\!|_n x_n M[x_1, \ldots, x_n]$,

где M представлена в КНФ.

Пусть, далее, $\rangle\!|_k x_k,\ 1 \leq k \leq n$, есть квантор $\exists x_k$. Если никакой квантор всеобщности $\forall x_j$ не находится в префиксе левее $\exists x_k = \rangle\!|_k x_k$, то выберем новую константу c, отличную от констант входящих в M, и заменим все вхождения x_k в M на c, а $\rangle\!|_k x_k$ удалим из префикса.

Если $\rangle\!|_{s_1} x_{s_1} \cdots \rangle\!|_{s_m} x_{s_m}$ – список всех кванторов всеобщности, встретившихся левее $\exists x_k = \rangle\!|_k x_k$, $1 \leq s_1 < s_2 < \cdots < k$, то выберем новый m-местный функциональный символ f, отличный от других функциональных символов, заменим в M все x_k на $f(x_{s_1}, \ldots, x_{s_m})$ и удалим $\rangle\!|_k x_k$ из префикса.

Этот процесс применим ко всем кванторам существования в префиксе. В результате полученная формула будет представлена в *стандартной нормальной форме* формулы F (сколемовской, клаузальной, стандартной форме). Константы и функции, используемые для замены переменных, связанных квантором $\exists$, называются *сколемовскими* функциями. Выбор сколемовских функций не произволен: для каждой конкретной формулы выбирается конкретная функция.

Примеры 5.7. .

1. Пусть дана формула $\forall y \exists x[x + y = 0], D = \mathcal{R}$ (множество вещественных чисел). Ясно, что этой формуле удовлетворяет функция $x = f(y) = (-y)$. Заменив вхождение переменной x в формулу на $f(y)$ и удалив квантор $\exists$ из префикса, получим стандартную форму исходной формулы:

$\forall y[(-y) + y = 0]$.

2.Пусть $G = (X, U)$ – обыкновенный граф, где X – множество вершин, U – множество ребер, а предикат $P(x, u, y)$ читается следующим образом: ребро $u \in U$ соединяет вершины $x, y \in X$. Тогда определение полного обыкновенного графа (все вершины которого попарно смежны) можно записать с помощью формулы:

$$\forall x \in X \forall y \in X \exists u \in U \ [P(x, u, y)].$$

Для представления этой формулы в стандартной форме заменим вхождение переменной u в формулу на $f(x, y)$ и удалим квантор $\exists$ из префикса. Теперь формула представляется в виде

$$\forall x \forall y \ [P(x, f(x, y), y],$$

где $f(x, y)$ определяет ребро, соединяющее вершины x, y. ◄

Следующий шаг, скорее традиционный чем необходимый, заключается в представлении формулы в виде множества клозов. В этом представлении предполагается, что некоторое множество клозов S есть конъюнкция всех клозов из F, а каждая переменная в атомах считается связанной квантором всеобщности.

Пример 5.8.
Представить в виде множества клозов формулу

$$\forall x \exists y \exists z[(\neg P(x, y) \wedge Q(x, z))) \vee R(x, y, z)].$$

Эта формула представлена в ПНФ. Приведем матрицу M к КНФ:

$$[\neg P(x, y) \vee R(x, y, z)] \wedge [Q(x, z) \vee R(x, y, z)].$$

Проведем сколемизацию. Перед кванторами $\exists y$ и $\exists z$ есть квантор всеобщности $\forall x$. Поэтому заменим переменные y, z соответственно одноместными функциями $f(x)$ и $g(x)$. Получим

$$\forall x[(\neg P(x, f(x)) \vee R(x, f(x), g(x)) \wedge (Q(x, g(x)) \vee R(x, f(x), g(x))].$$

Образуем множество клозов:

$$S = \{\neg P(x, f(x)) \vee R(x, f(x), g(x)), Q(x, g(x)) \vee R(x, f(x), g(x))\}. ◄$$

Приводя формулу к стандартной (клаузальной) форме, мы хотели иметь удобное представление для доказательства противоречивости формулы. Удаляя кванторы существования мы предполагаем, что противоречивость формулы сохраняется. Это действительно так.

Пусть S – множество клозов, которые представляют стандартную форму формулы F.

Теорема 5.1 *Формула F противоречива тогда и только тогда, когда множество S противоречиво.*

З а м е ч а н и е. Подчеркнем, что при сколемизации сохраняется только противоречивость, а не эквивалентность $F = S$, которая может не сохраняться. ▶

Д о к а з а т е л ь с т в о . Без потери общности можно положить, что F представлена в ПНФ, т.е.

$$F = \rangle\!\rangle|_1 x_1 \cdots \rangle\!\rangle|_n x_n M[x_1, \ldots, x_n].$$

Пусть $\rangle\!\rangle|_k$ первый слева в формуле квантор $\exists$ и пусть

$$F_1 = \rangle\!\rangle|_1 x_1 \cdots \rangle\!\rangle|_{k-1} x_{k-1} \rangle\!\rangle|_{k+1} x_{k+1} \cdots \rangle\!\rangle|_n x_n$$
$$M[x_1, \ldots, x_{k-1}, f(x_1, \ldots, x_{k-1}), \ldots, x_n],$$

где $f(x_1, \ldots, x_{k-1})$ – сколемовская функция, соответствующая $x_k,\ 1 \le k \le n$.

Покажем, что F противоречива тогда и только тогда, когда F_1 противоречива.

$\Rightarrow$ Пусть F противоречива. Если F_1 непротиворечива, то существует такая интерпретация I, что F_1 истинна в I, т.е. для всех $x_1, \ldots, x_{k-1}$ существует элемент $f(x_1, \ldots, x_{k-1})$, для которого формула

$$\rangle\!\rangle|_{k+1} x_{k+1} \cdots \rangle\!\rangle|_n x_n M[x_1, \ldots, x_{k-1}, f(x_1, \ldots, x_{k-1}), x_{k+1}, \ldots, x_n],$$

истинна в I. Но тогда и F истинна в I, что противоречит предположению. Следовательно, F_1 должна быть противоречивой.

$\Leftarrow$ Пусть теперь F_1 противоречива. Если F непротиворечива, то в D существует такая интерпретация I, что F истинна в I, т.е. для всех $x_1, \ldots, x_{k-1}$ существует такой элемент x_k, что

$$\exists_{k+1} x_{k+1} \cdots \exists_n x_n M[x_1, \ldots, x_{k-1}, x_k, x_{k+1}, \ldots, x_n],$$

истинна в I.

Расширим интерпретацию I до I', включив функцию $f(x_1, \ldots, x_{$ x_k (функция, которая отображает $x_1, \ldots, x_{k-1}$ в x_k). Тогда

$$\forall x_1 \cdots \forall x_{k-1} \exists_{k+1} x_{k+1} \cdots \exists_n x_n$$

$$M[x_1, \ldots, x_{k-1}, f(x_1, \ldots, x_{k-1}), x_{k+1}, \ldots, x_n],$$

истинна вI', т.е. F_1 истинна в I', что противоречит предположению. Следовательно, F должна быть противоречивой.

Следующий шаг доказательства.

Пусть в F имеется m кванторов существования. Положим $F_0 = F$ (F_1 была определена ранее). Определим, что $F_l,\ l = \overline{1, m}$, получается из F_{k-1} заменой переменной, связанной первым слева квантором существования, сколемовской функцией. Ясно, что $F_m = S$.

Используя те же рассуждения, можно показать, что F_{l-1} противоречива тогда и только тогда, когда F_l противоречива для всех $l = \overline{1, m}$. В конечном счете мы заключим, что F противоречива тогда и только тогда, когда S противоречива. Теорема доказана. ⊠

5.4 Подстановки и унификация

Наша цель состоит в применении метода резолюций для логики предикатов первого порядка. Существенным в применении метода резолюций является нахождение контрарных пар литералов. Если клозы не содержат переменных, то резолюция выполняется так же, как и для высказываний. Если клозы содержат переменные, то задача усложняется.

Пример 5.9.

$C_1 : P(x) \vee Q(x)$,

$C_2 : \neg P(f(y)) \vee R(x)$.

Литералы $P(x)$ и $P(f(y))$ в исходном виде не образуют контрарную пару. Однако если подставить $a \in D$ вместо y в C_2 и $f(a)$ вместо x в C_1 и C_2, то получим

$C_1' : P(f(a)) \vee Q(f(a))$,

$C_2' : \neg P(f(a)) \vee R(f(a))$.

Теперь $P(f(a))$ и $\neg P(f(a))$ контрараны и из C_1' и C_2 можно получить резольвенту $C' : Q(f(a)) \vee R(f(a))$.

В общем случае, подставив $f(y)$ вместо x в C_1 и C_2, получим:

$C_1^\star : P(f(y)) \vee Q(f(y))$,

$C_2^\star : \neg P(f(y)) \vee R(f(y))$.

Литералы $P(f(y))$ и $\neg P(f(y))$ теперь контрарны и можно получить резольвенту

$C^\star : Q(f(y)) \vee R(f(y))$. ◄

Определение. ***Подстановка*** *– это конечное множество вида*

$\{t_1/v_1, \ldots, t_n/v_n\}$,

где v_i *– переменная,* t_i *– терм, отличный от* v_i*, и все* v_i *различны.*

Выражением называют терм, множество термов, множество атомов, множество клозов, формулу и т.д. Любое выражение, не содержащее переменных, называют *основным*, например, основной терм. Если термы основные, то подстановка называется основной. Подстановка, которая не содержит элементов называется *пустой* и обозначается ε.

Пример 5.10.

$\{f(z)/x, y/z\}, \{a/x, g(y)/y, f(g(b))/z\}$ – примеры подстановок; $\{x/a, z/y\}$ – не подстановка, так как a не переменная. ◄

Пусть $\theta = \{t_1/v_1, \ldots, t_n/v_n\}$ – подстановка, E – выражение. Тогда $E\theta$ – выражение, полученное из E заменой одновременно всех вхождений переменных v_i, $i = l1, m$ в E на термы t_i.

Определение. *$E\theta$ называется* ***примером*** *E.*

Пример 5.11.

$\theta = \{a/x, f(b)/y, c/z\}, E = P(x, y, z)$ – выражение,

$E\theta = P(a, f(b), c)$ – основной пример.

Пусть заданы две подстановки:

$\theta = \{t_1/x_1, \ldots, t_n/x_n\}$ и $\lambda = \{u_1/y_1, \ldots, u_m/y_m\}$.

Определение. ***Композиция*** *подстановок θ и λ (обозначается $\theta \circ \lambda$) есть подстановка, которая получается из множества*

$$\{t_1\lambda/x_1, \ldots, t_n\lambda/x_n, u_1/y_1, \ldots, u_m/y_m\}$$

удалением всех элементов $t_j\lambda/x_j$, для которых $t_j\lambda = x_j$ и всех элементов u_i/y_i, таких, что $y_i \in \{x_1, \ldots, x_n\}$.

Пример 5.12.

Пусть $\theta = \{t_1/x_1, t_2/x_2\} = \{f(y)/x, z/y\}$,
$\lambda = \{u_1/y_1, u_2/y_2, u_3/y_3\} = \{a/x, b/y, y/z\}$.
Тогда $\theta \circ \lambda = \{t_1\lambda/x_1, t_2\lambda/x_2, u_1/y_1, u_2/y_2, u_3/y_3\} =$
$= \{f(b)/x, y/y, a/x, b/y, y/z\}$
Так как $t_2\lambda = x_2$, то $t_2\lambda/x_2$ (т.е. y/y) удаляется. Далее, y_1 и $y_2 \in \{x_1, x_2\}$, значит u_1/y_1 и u_2/y_2 (т.е. a/x и b/y) тоже удаляются. В результате получим:

$$\theta \circ \lambda = \{f(b)/x, y/z\}.$$ ◄

Нетрудно заметить, что композиция подстановок ассоциативна и дает тот же результат, что и последовательное их применение, т.е. выражение $E(\theta \circ \lambda)$ эквивалентно $(E\theta)\lambda$.

В процедуре доказательства методом резолюций для отождествления контрарных пар литералов придется *унифицировать* два или более выражений, т.е. нужно найти подстановку, которая может сделать несколько выражений тождественными.

Определение. *Подстановка θ называется* ***унификатором*** *для множества выражений $\{E_1, \ldots, E_k\}$ тогда и только тогда, когда $E_1\theta = E_2\theta = \cdots E_k\theta$.*

Множество $\{E_1, \ldots, E_k\}$ ***унифицируемо****, если для него существует унификатор.*

Пример 5.13. Множество выражений $E = \{P(a, y), P(x, f(z))\}$ унифицируемо, т.к. подстановка $\theta_1 = \{a/x, f(z)/y\}$ является его унификатором. Другой унификатор: $\theta_2 = \{a/x, a/z, f(a)/y\}$. ◄

Таким образом, для множества выражений может существовать несколько унификаторов. Нас будет интересовать в некотором смысле простейший из унификаторов.

Определение. *Унификатор σ для множества выражений $\{E_1, \ldots, E_k\}$ является* ***наиболее общим унификатором*** *(НОУ) тогда и только тогда, когда для каждого унификатора θ этого множества существует такая подстановка λ, что $\theta = \sigma \circ \lambda$.*

В предыдущем примере θ_1 есть НОУ, а $\theta_2 = \theta_1 \circ \{a/z\}$.

5.4.0.1 Алгоритм унификации

Этот алгоритм позволяет найти НОУ для конечного множества выражений, если оно унифицируемо. Если множество неунифицируемо, алгоритм выявит этот факт.

Рассмотрим выражение $P(a)$ и $P(x)$. Они не тождественны. Чтобы отождествить выражения, нужно сначала найти *рассогласование*, а затем попытаться его исключить. Для $P(a)$ и $P(x)$ рассогласование будет $\{a, x\}$. Так как x – переменная, то x можно заменить на a и, таким образом, устранить рассогласование (рассогласование начинается с 3-й знаковой позиции в выражениях). В этом заключается идея алгоритма унификации.

Множество рассогласований $\mathcal{D}$ непустого множества выражений E получается выявлением первой слева позиции, в которой не для всех выражений из E находится один и тот же символ, и затем выписыванием из каждого выражения в E подвыражения, которое начинается с символа, занимающего эту позицию.

Пример 5.14. Пусть $E = \{P(x, f(y, z)), P(x, a), P(x, g(h(k(x))))\}$. Первая позиция, в которой не все выражения содержат один и тот же символ – пятая (первые 4 символа, включая скобки,одинаковы во всех выражениях $P(x, \ldots)$). Таким образом, множество рассогласований есть

$$\{f(y, z), a, g(h(k(x)))\}.$$

5.4.0.2 Описание алгоритма

Шаг 1. Положить $k = 1; E_k = E; \sigma_k = \varepsilon$.

Шаг 2. Если E_k состоит из одного выражения (т.е. все выражения равны), то **останов**: σ_k есть НОУ для E. В противном случае найти множество рассогласований $\mathcal{D}_k$ для E_k.

Шаг 3. Если в $\mathcal{D}_k$ существуют такие элементы v_k и t_k, что v_k – переменная, не входящая в t_k, то на **Шаг 4**; иначе **стоп**: множество не может быть унифицировано.

Шаг 4. Положить $\sigma_{k+1} = \sigma_k \circ \{t_k/v_k\}$ и $E_{k+1} = E_k\{t_k/v_k\}$. (Заметим, что $E_{k+1} = E\sigma_{k+1}$).

Шаг 5. Положить $k = k + 1$; на **Шаг 2**.

Пример 5.15.
Найти НОУ для множества выражений:

$$E = \{P(a, x, f(g(y))),\ P(z, f(z), f(u))\}.$$

Шаг 1. $k = 0; E_0 = E; \sigma_0 = \varepsilon$.
Шаг 2. Так как в E выражения различны, то σ_0 не является НОУ для E. Множество рассогласований $\mathcal{D}_0 = \{a, z\}$.
Шаг 3. В $\mathcal{D}_0$ есть переменная z, не входящая в a.
Шаг 4. Положим $\sigma_1 = \sigma_0 \circ \{t_0/v_0\} = \varepsilon \circ \{a/z\} = \{a/z\}$.
$E_1 = E\{t_0/v_0\} = E\{a/z\} = \{P(a, x, f(g(y))),\ P(a, f(a), f(u))\}$
Шаг 5. $k := k + 1 = 1$; на Шаг 2.
Шаг 2. Множество E_1 содержит более одного выражения. Множество рассогласований для E_1: $\mathcal{D}_1 = \{x, f(a)\}$.
Шаг 3. Из $\mathcal{D}_1$ находим: $v_1 = x; t_1 = f(a)$.
Шаг 4. $\sigma_2 = \sigma_1 \circ \{t_1/v_1\} = \{a/z\} \circ \{f(a)/x\} = \{a/z, f(a)/x\}$.
$E_2 = E_1\{t_1/v_1\} = E_1\{f(a)/x\} = \{P(a, f(a), f(g(y))), P(a, f(a), f(u))\}$.
Шаг 5. $k = 2$; на Шаг 2.
Шаг 2. Множество E_2 содержит более одного выражения. Множество рассогласований для E_2: $\mathcal{D}_2 = \{g(y), u\}$.
Шаг 3. $v_2 = u, t_2 = g(y); u$ не содержится в $g(y)$.
Шаг 4. $\sigma_3 = \sigma_2 \circ \{g(y)/u\} = \{a/z, f(a)/x, g(y)/u\}$.
$E_3 = E_2\{g(y)/u\} = \{P(a, f(a), f(g(y))),\ P(a, f(a), f(g(y)))\} =$
$= \{P(a, f(a), f(g(y)))\}$.
Шаг 5. $k = 3$; на Шаг 2.
Шаг 2. Так как E_3 содержит одно выражение, то

$\sigma_3 = \{a/z, f(a)/x, g(y)/u\}$ –НОУ для E. ◄**Пример 5.16.** Множество выражений: $E = \{Q(f(a), g(x)), Q(y, y)\}$ Рассмотрим вкратце попытку унифицировать эти выражения.

$\sigma_1 = \{f(a)/y\}$; $E_1 = \{Q(f(a), g(x)), Q(f(a), f(a))\}$;

E_1 содержит более одного выражения; множество рассогласований $\mathcal{D}_1 = \{g(x), f(a)\}$. В $\mathcal{D}_1$ нет переменной, нет НОУ. Отсюда заключаем, что E не может быть унифицировано. ◄

Теорема 5.2 (теорема унификации) *Если E – конечное непустое унифицироемое множество выражений, то алгоритм унификации всегда закончит работу на шаге 2 и последняя σ_k будет наиболее общим унификатором для E.*

(Без доказательства).

5.5 Метод резолюций для логики первого порядка

В логике высказываний резольвента определяется как дизъюнкция двух клозов, из которых удалена пара контрарных литералов. Обобщим теперь понятие резольвенты на логику первого порядка.

Определение. *Если два или более литералов (с одинаковым знаком) клоза C имеют НОУ σ, то $C\sigma$ называется* ***склейкой*** *клоза C.*

Пример склейки 5.17.

Клоз $C : P(x) \vee P(f(y)) \vee \neg Q(x)$; НОУ: $\sigma = \{f(y)/x\}$;

$$C\sigma = P(f(y)) \vee P(f(y)) \vee \neg Q(f(y)) = \overbrace{P(f(y)) \vee \neg Q(f(y))}^{\text{склейка}}. \blacktriangleleft$$

Пример 5.18. Если $C\sigma$ – единичный клоз, то склейка называется *единичной склейкой*.

З а м е ч а н и е. Клоз, содержащий m литералов называется *m-литерным* клозом. Однолитерный клоз называется *единичным*, 0-литерный – это пустой клоз $\square$. ►

Для удобства будем рассматривать множество литералов как синоним клоза. Например, клоз $P \vee Q \vee \neg R$ представляется в виде множества литералов $\{P, Q, \neg R\}$.

Пусть C_1, C_2 – клозы (клозы-посылки), которые не имеют общих переменных. Пусть L_1, L_2 – литералы в C_1 и C_2 соответственно.

Определение. *Если L_1 и $\neg L_2$ имеют НОУ σ, то клоз*

$$C = (C_1\sigma \setminus L_1\sigma) \cup (C_2\sigma \setminus L_2\sigma)$$

*называется **бинарной резольвентой** клозов C_1 и C_2.*

Литералы L_1, L_2 называются *отрезаемыми* литералами.

Пример 5.19.

$C_1 : P(x) \vee Q(x)$; $C_2 : \neg P(a) \vee R(x)$.

Так как x входит в оба клоза (они имеют общую переменную), заменим переменную в $C_2 : \neg P(a) \vee R(y)$.

$L_1 = P(x)$; $L_2 = \neg P(a)$.

L_1 и L_2 имеют НОУ: $\sigma = \{a/x\}$. Тогда

$$(C_1\sigma \setminus L_1\sigma) \cup (C_2\sigma \setminus L_2\sigma) =$$

$$= \{\{P(a), Q(a)\} \setminus P(a)\} \cup \{\{\neg P(a), R(y)\} \setminus \neg P(a)\} =$$

$$= \{Q(a), R(y)\} = Q(a) \vee R(y).$$

Таким образом, $Q(a) \vee R(y)$ – бинарная резольвента клозов C_1 и C_2; $P(x)$ и $\neg P(a)$ – отрезаемые литералы. ◄

Определение. *Резольвентой клозов- посылок C_1 и C_2 является одна из следующих резольвент:*

1. *Бинарная резольвента C_1 и C_2.*
2. *Бинарная резольвента C_1 и склейки C_2.*
3. *Бинарная резольвента C_2 и склейки C_1.*
4. *Бинарная резольвента склейки C_1 и склейки C_2.*

Пример 5.20.

$C_1 : P(x) \vee P(f(y)) \vee R(g(y))$;

$C_2 : \neg P(f(g(a))) \vee Q(b)$.

$\sigma_1 = \{f(y)/x\}$, склейка C_1 есть

$C_1' : P(f(y)) \vee P(f(y)) \vee R(g(y)) = P(f(y)) \vee R(g(y))$.

$\sigma_2 = \{g(a)/y\}$;

$C_1'' : P(f(g(a))) \vee R(g(g(a)))$;

$C_2 : \neg P(f(g(a))) \vee Q(b)$;

Бинарная резольвента (она же и резольвента) C_1' и C_2 есть

$C : \; R(g(g(a))) \vee Q(b)$.

5.6 Исчисление высказываний

Формальные теории

Формальная теория (исчисление) $\boldsymbol{T}$ строится следующим образом.

1. Задается конечное или счетное множество символов $\boldsymbol{A}$ – ***алфавит*** теории.

2. Определяется множество $\boldsymbol{F}$ слов в алфавите $\boldsymbol{A}$, $\boldsymbol{F} \subset \bigcup\limits_{i=1}^{\infty} \boldsymbol{A}^i$ – ***формулы*** (правильно построенные выражения). Это множество задается конструктивно (как правило, индуктивным определением). Множества $\boldsymbol{A}$ и $\boldsymbol{F}$ определяют язык (***сигнатуру***) теории.

3. Из $\boldsymbol{F}$ выделяется подмножество формул – ***аксиомы*** теории.

4. Задаются ***правила вывода*** теории. Правило вывода $R(F_1, \ldots, F_n, G)$ – это вычислимое отношение на множестве формул. Если формулы $F_1, \ldots, F_n$ и G находятся в отношении R, то формула G называется ***непосредственно выводимой*** из $F_1, \ldots, F_n$ по правилу R.

Часто правило $R(F_1, \ldots, F_n, G)$ записывают в виде $\dfrac{F_1, \ldots, F_n}{G}R$.

Формулы $F_1, \ldots, F_n$ называются ***посылками*** правила R, а G – его ***следствием*** или ***заключением***.

5. ***Выводом*** формулы G из формул $F_1, \ldots, F_n$ в теории $\boldsymbol{T}$ называется такая последовательность формул $\Phi_1, \ldots, \Phi_n, \Phi_{n+1}, \ldots, \Phi_N$, что $\Phi_N = G$, а любая Φ_i,$(i = 1 \ldots, N)$ есть либо аксиома, либо

одна из исходных формул $F_1, \ldots, F_n$, либо непосредственно выводима из формул $\Phi_1, \ldots, \Phi_{i-1}$ (или какого-то их подмножества) по одному из правил вывода.

Если существует вывод G из $F_1, \ldots, F_n$, то говорят,что G ***выводима*** из $F_1, \ldots, F_n$ в теории T. Этот факт обозначают

$$F_1, \ldots, F_n \vdash_T G$$

Если теория T фиксирована, то индекс опускается:

$$F_1, \ldots, F_n \vdash G$$

Формулы $F_1, \ldots, F_n$ называются ***гипотезами*** или ***посылками*** вывода. Множество гипотез обозначается $\Gamma = \{F_1, \ldots, F_n\}$. Переход в выводе от Φ_{i-1} к Φ_i называется i-м шагом вывода.

6. ***Доказательством*** формулы G в теории T называется вывод G из пустого множества формул, то есть вывод, в котором в качестве исходных формул используются только ***аксиомы***

$$\vdash_T G$$

Формула G, для которой существует доказательство называется формулой, ***доказуемой*** в теории T, или ***теоремой*** теории T.

Добавление формул к гипотезам не нарушает выводимости. Поэтому, если $\vdash G$, то $F \vdash G$, и если $F_1, \ldots, F_n \vdash G$, то и $F_1, \ldots, F_n, F_{n+1} \vdash G$ при любых F и F_{n+1}. Порядок гипотез несуществен.

Замечания. При изучении формальных теорий рассматриваются два типа высказываний:

1. Высказывания самой теории (***теоремы***) – формальные объекты определенные ранее.

2. Высказывания о теории (о свойствах теорем, доказательств и т.д.), которые формулируются на языке, внешнем по отношению к теории – ***метаязыке*** и называемые ***метатеоремами***.

Интерпретацией формальной теории T в область интерпретации называется функция $I : f \mapsto M$, которая каждой формуле формальной теории T однозначно сопоставляет некоторое содержательное высказывание относительно объектов множества (алгебраической системы) . Это высказывание может быть истинным или ложным (или не иметь истинностного значения). Если соответствующее высказывание является истинным, то говорят, что формула ***выполняется*** в данной интерпретации.

Интерпретация I называется ***моделью*** множества формул Г, если все формулы этого множества выполняются в интерпретации I. Интерпретация I называется ***моделью*** формальной теории , если все теоремы этой теории выполняются в интерпретации I (то есть все выводимые формулы оказываются истинными в данной интерпретации).

Формальная теория должна опираться на общие логические законы (общезначимые высказывания типа $A \vee \neg A$ или $\forall x P(x) \to P(y)$).

Формальная теория называется ***полной***, если для всякого высказывания A в этой теории

$$\vdash A \text{ или } \vdash \neg A.$$

Формальная теория называется ***непротиворечивой***, если в ней не является доказуемой формула

$$A \wedge \neg A.$$

Система аксиом непротиворечивой теории называется ***независимой***, если никакая из аксиом не выводима из остальных по правилам вывода теории.

Формальная теория называется ***разрешимой***, если существует алгоритм, который определяет доказуемость для любой формулы теории.

Классическое исчисление высказываний (КИВ)

определяется следующим образом:

1. Алфавит КИВ состоит из переменных высказываний (пропозициональных переменных, букв, литер, атомов)

$$a, b, c, \ldots, p, q, r, \ldots, A, B, C, \ldots, F, G, H, \ldots,$$

знаков логических связок

$\neg, \wedge, \vee, \rightarrow, \leftrightarrow,$

скобок, запятых и точек.

2. Формулы (синтаксические правила):

a) атом есть формула;

b) если A, B – формулы, то

$\neg A, (A \wedge B), (A \vee B), (A \rightarrow B), (A \leftrightarrow B)$

формулы. Внешние скобки обычно опускают.

c) других формул нет (пропозициональными формулами являются лишь символы, построенные в соответствии с правилами a), b))

3. Аксиомы (общезначимые формулы).

4. Правила вывода.

1. Правило подстановки.

Если $A(a)$ – выводимая формула, содержащая букву a, то выводима и формула $A(B)$, полученная из A заменой всех вхождений a на произвольную формулу B.

2. Правило заключения

Логическим выводом формулы B из формул $A_1, \ldots, A_n$ называется такая последовательность формул

$$C_1 \ldots C_k, B,$$

что каждая формула C_i $(i \leqslant k)$ является либо аксиомой, либо исходной формулой A_j $(C_i = A_j)$, либо получена из предыдущих формул последовательности по правилам вывода.

Тот факт, что формула B выводима из формул $A_1, \ldots, A_n$, выражают записью

$$A_1, \ldots, A_n \vdash B. \ (\star)$$

Здесь $\vdash$ – ***оператор дедуктивной выводимости***; формулы $A_1, \ldots, A_n$ называются ***гипотезами*** вывода и обозначаются $\Gamma = \{A_1, \ldots, A_n\}$. Тогда $(\star)$ есть $\Gamma \vdash B$.

Дедуктикой D называется конечный набор систем аксиом и правилам вывода. Рассмотрим некоторые из них.

Система аксиом 1.

Эта система аксиом использует только две связки: $\neg, \to$; при этом сокращается алфавит исчисления.

Другие связки вводятся определениями (как сокращения некоторых формул), а не аксиомами: $A \vee B$ заменяет $\neg A \to B$, $A \wedge B$ – $\neg(A \to \neg B)$, $A \leftrightarrow B$ – $(A \to B) \wedge (B \to A)$.

Аксиомы:

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

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

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

Правило вывода одно – Modus Ponens (MP, правило отделения посылки)

$$\frac{A, A \to B}{B}$$

Пример 5.21. Показать, что формула $a \to a$ выводима из аксиом.

$$\vdash a \to a$$

#	Формулы	Comment
1	$a \to ((a \to a) \to a))$	A1: a/A, $(a \to a)/B$
2	$(a \to ((a \to a) \to a)) \to$ $\to ((a \to (a \to a)) \to (a \to a))$	A2:подстановки $(a \to a)/B$, a/C
3	$(a \to (a \to a)) \to (a \to a)$	MP из 1,2
4	$a \to (a \to a)$	A1:a/B
5	$a \to a$	MP из 3,4

Пример 5.22. Вывести $a \vdash b \to a$.

Из аксиомы 1 $(A \to (B \to A))$ по правилу MP

$$\frac{a, a \to (b \to a)}{b \to a}$$

Всякую доказанную в исчислении выводимость вида $\Gamma \vdash A$, где Γ – множество формул (гипотез), можно рассматривать как правило вывода $\dfrac{\Gamma}{A}$, которое можно добавить к уже имеющимся.

Полученную выводимость $A \vdash B \to A$ вместе с правилом подстановки можно рассматривать как правило $\dfrac{A}{B \to A}$: если формула A выводима, то выводима и формула $B \to A$, где B – любая формула.

Пример 5.23. Доказать выводимость: $a \to b, b \to c \vdash a \to a \to c$.

#	Формулы	Comment
1.	$b \to c \vdash a \to (b \to c)$	Правило $\dfrac{A}{B \to A}$
2.	$(a \to b) \to (a \to c)$	Из $a \to (b \to c)$ и A2 по MP $b \to c \vdash (a \to b) \to (a \to c)$
3.	$a \to c$	Из $a \to b$ и $(a \to b) \to (a \to c)$ по MP
4.	$a \to a \to c$	Правило $\dfrac{A}{B \to A}$: $\dfrac{a \to c}{a \to a \to c}$

Вывод $\dfrac{A \to B, B \to C}{A \to C}$ дает новое правило вывода, которое называется ***правилом транзитивности импликации*** (правило ***силлогизма***).

Система аксиом 2. (Клини, 1952)

1. $A \to (B \to A)$;
2. $(A \to B) \to ((A \to (B \to C)) \to (A \to C))$;
3. $A \to (B \to (A \wedge B))$ — введение $\wedge$;

$$\left.\begin{array}{l} 4. (A \wedge B) \to A \\ 5. (A \wedge B) \to B \end{array}\right\} \text{— удаление } \wedge;$$

$$\left.\begin{array}{l} 6. A \to (A \vee B) \\ 7. B \to (A \vee B) \end{array}\right\} \text{— введение } \vee;$$

8. $(A \to C) \to ((B \to C) \to ((A \vee B) \to C))$ — удаление $\vee$;
9. $(A \to B)((A \to \neg B) \to \neg A)$ — введение $\neg$;
10. $\neg\neg A \to A$ — удаление $\neg$;
11. $(A \to B) \to ((B \to A) \to (A \leftrightarrow B))$ — введение $\leftrightarrow$;

$$\left.\begin{array}{l} 12.(A \leftrightarrow B) \to (A \to B) \\ 13.(A \leftrightarrow B) \to (B \to A) \end{array}\right\} — \text{удаление} \leftrightarrow .$$

Эти системы аксиом равносильны в том смысле, что порождают одно и то же множество формул (эти системы аксиом выводятся друг из друга с учетом обозначения связок $\wedge, \vee$).

Пример 5.24. Доказать выводимость: $a \to (b \to c), a \wedge b \vdash c$

#	Вывод	Comment
1.	$a \to (b \to c)$	Посылка
2.	$a \wedge b$	Посылка
3.	$a \wedge b \to a$	A4
4.	a	MP (2, 3)
5.	$a \wedge b \to b$	A5
6.	b	MP (2,5)
7.	$b \to c$	MP (4,1)
8.	c	MP (6,7)

Система аксиом 3 (Гильберт, Аккерман, 1938)

Связки: $\neg, \vee, \quad (A \to B = \neg A \vee B)$

Аксиомы:

1. $A \vee A \to A$;
2. $A \to A \vee B$;
3. $A \vee B \to B \vee A$;
4. $(B \to C) \to (A \vee B \to A \vee C)$.

Правило: $\dfrac{A, A \to B}{B}$ (Modus Ponens)

Система аксиом 4 (Россер, 1953)

Связки: $\neg, \wedge, \quad (A \to B = \neg(A \wedge \neg B))$.

Аксиомы:

1. $A \to A \wedge A$;
2. $A \wedge B \to A$;
3. $(A \to B) \to (\neg(B \wedge C) \to \neg(C \wedge A))$.

Правило: Modus Ponens.

Система аксиом 5 (Никод, 1917)

Связка: $|$ $(A|B = \neg A \vee \neg B)$.

Аксиома:

$(A|(B|C))|((D|(D|(D|D))| \ ((E|B)|((A|E)|(A|E)))))$.

Правило: $\dfrac{A, A|(B|C)}{C}$

Система натурального вывода

1. $A \to B, A \vdash B$ удаление импликации, Modus Ponens (правило отделения)

2. $A \to B, \neg B \vdash \neg A$ Modus Tollens

(правило отрицания)

3. $A, B \vdash A \wedge B$ введение конъюнкции

4. $A \wedge B \vdash A$ удаление конъюнкции

$A \wedge B \vdash A$

5. $A \vdash A \vee B$ введение дизъюнкции

6. $A \vee B, \neg B \vdash A$ удаление дизъюнкции

7. $\neg\neg A \vdash A$ удаление отрицания

8. $A \to B, B \to C \vdash A \to C$ транзитивность импликации (правило силлогизма, цепное правило)

9. $A \to B, A \to \neg B \vdash \neg A$ сведение к абсурду (reductio ad absurdum)

10. $A \to B \vdash \neg B \to \neg B$ правило контрпозиции

11. $A \vee B, \neg A \vdash B$ $A \vee B, \neg B \vdash A$

$\neg A \vee B, A \vdash B$

$A \vee \neg B, B \vdash A$

$A \vee B \vee C, \neg A \wedge \neg B \vdash C$

дизъюнктивные силлогизмы

(разновидности Modus Tollendo Ponens)

12. $A \to B, C \to B, A \vee C \vdash B$ конструктивная дилемма

$A \to B, A \to C, \neg B \vee \neg C \vdash \neg A$ деструктивная дилемма

Свойства отношения выводимости

Теорема 1. $a)$ $A_1, \ldots, A_n \vdash A_i$ $(i = 1, \ldots, n)$;

Доказательство. Последовательность формул $A_1, \ldots, A_n$ запишем в произвольном порядке, поместив последней формулу A_i.

$b)$ Если

$\ldots\ldots,$

$$\frac{\begin{array}{ll} A_1, \ldots, A_n \vdash B_k, & \text{и} \\ B_1, \ldots, B_k \vdash C, & \text{то} \end{array}}{A_1, \ldots, A_n \vdash C}$$

Доказательство. В выводе C из $B_1, \ldots, B_k$ подставим вместо B_i их выводы из $A_1, \ldots, A_n$ и получим вывод $A_1, \ldots, A_n \vdash C$.

Часть 3

Графы

Глава 6

Определения и примеры

6.1 Определения графа

Граф как бинарное отношение

Пусть даны два множества X, Y, и на них задано отношение $R \subseteq X \times Y$. Граф G обозначается как тройка $G = \langle X, Y, R \rangle$, или $G = (X, Y, R)$. Чаще, однако, полагают $X = Y$ и $R \subseteq X \times X$; тогда $G = (X, R)$.

Определение. *Граф $G = (X, R)$ задан, если заданы множество X и бинарное отношение R на X.*

Элементы множества X называются ***вершинами***. Элементы множества R – упорядоченные пары вершин $\langle x, y \rangle$, или (x, y), или $\overrightarrow{xy}$ называются ***дугами***. Пара (x, x) называется ***петлей***.

Говорят, что дуга (x, y) ***исходит*** из вершины x и ***заходит*** в вершину y, или дуга ***соединяет*** вершины x и y.

Пример 6.1.
$X = \{a, b, c, d, e\}$; $R = \{(a, e), (a, d), (b, a), (b, c), (c, c), (d, a), (d, b), (d, c)\}$ ◀

Определение Бержа

Определение. *Граф $G = (X, \Gamma)$ задан, если заданы множество X и отображение $\Gamma : X \mapsto X$.*

Здесь X – множество *вершин*; множество пар (x, y), для которых $x \in X, y \in \Gamma x$ – это множество *дуг*; пары (x, x), $x \in \Gamma x$ – *петли*.

Пример 6.2.
$X = \{a, b, c, d, e\}$; $\Gamma a = \{c, d\}$, $\Gamma b = \{b, c\}$, $\Gamma c = \varnothing$, $\Gamma d = \{a\}$, $\Gamma e = \{e\}$.
Дуги: $(a, c), (a, d), (b, c), (d, a)$; петли: $(b, b), (e, e)$. ◀

Для иллюстрации вершины графа изображаются в виде точек или кружков, а дуги (x, y) в виде стрелок, соединяющих пару вершин.

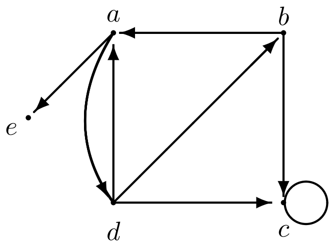


Рис. 6.1. Граф как бинарное отношение

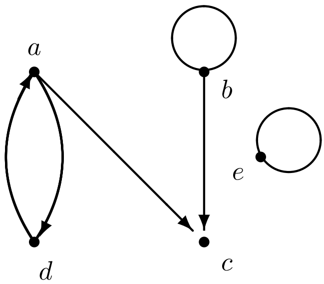


Рис. 6.2. Граф как отображение

Графы из примеров 1 и 2 представлены на рисунках 6.1 и 6.2.

Определение Зыкова

Определение. *Граф* $G = (X, U; P)$ *задан, если заданы два множества* $X \neq \varnothing, U (X \cap U = \varnothing)$ *и трехместный предикат* P*, такой, что*

1. P *определен на всех упорядоченных тройках элементов, для которых* $x, y \in X, u \in U$.

2. $\forall u \exists x, y \{P(x, u, y) \ \wedge \ \forall x', y' [P(x', u, y')$

$$\Rightarrow (x = x') \ \wedge \ (y = y') \vee \ (x = y') \ \wedge \ (y = x')]\}.$$

Здесь X – множество *вершин*, U – множество *ребер*, $P(x, u, y)$ – ***инцидентор*** графа G. Высказывание $P(x, u, y)$ читается так: ребро u *соединяет* вершину x с вершиной y, или ребро u соединяет упорядоченную пару вершин $\langle x, y \rangle$.

Условие 2 говорит о том, что каждое ребро графа соединяет какую-либо пару вершин (x, y), но кроме этой пары может (хотя и необязательно) соединять еще только обратную пару (y, x). Для любого ребра $u \in U$ истинно одно и только одно из трех следующих высказываний:

$\forall u \in U$

$$\exists x, y \in X[x \neq y \ \wedge \ P(x, u, y) \ \wedge \ \neg P(y, u, x)]$$

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

$$\exists x, y[x \neq y \ \wedge \ P(x, u, y) \ \wedge \ P(y, u, x)].$$

В соответствии с этим множество ребер U разбивается на три попарно непересекающихся подмножества $\overrightarrow{U}$, $\mathring{U}$, $\widetilde{U}$, которые называются, соответственно, множествами ***дуг***, ***пéтель*** и ***звеньев***. Если для некоторой тройки элементов x, u, y истинно высказывание $P(x, u, y)$, т.е. ребро u соединяет вершину x с вершиной y, то

• если $u \in \overrightarrow{U}$S, то говорят: дуга u исходит из вершины x и заходит в вершину y, или дуга u идет из вершины x в вершину y ;

• если $u \in \mathring{U}$, т.е. $x = y$, то говорят: u есть петля при вершине x;

• если $u \in \widetilde{U}$, то говорят: вершины x, y соединены звеном u.

Пример 6.3.

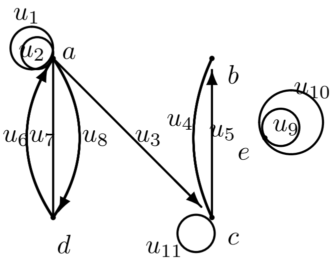


Рис. 6.3. Граф по Зыкову

$X = \{a, b, c, d, e\}$, Истинны следующие высказывания:
$P(a, u_1, a), P(a, u_2, a), P(a, u_3, c), P(b, u_4, c)$, $P(c, u_4, b)$,

$P(c, u_5, b)$, $P(c, u_{11}, c), P(a, u_7, d), P(d, u_7, a)$, $P(a, u_8, d)$,

$P(d, u_6, a)$, $P(e, u_9, e), P(e, u_{10}, e)$. ◄

Две вершины x, y называются ***смежными***, если существует по крайней мере одно соединяющее их ребро, т.е. если истинно высказывание

$J(x, y) \Leftrightarrow \exists u[P(x, u, y) \vee P(y, u, x)]$,

в частности, вершина смежна сама с собой тогда и только тогда, когда при ней имеется хотя бы одна петля.

С помощью инцидентора P определяются еще три двуместных предиката:

$I^+(x, u) \Leftrightarrow \exists z P(x, u, z)$,

$I^-(x, u) \Leftrightarrow \exists z P(z, u, x)$,

$I^\circ(x, u) \Leftrightarrow P(x, u, x)$,

а также двуместный предикат

$I(x, u) \Leftrightarrow I^+(x, u) \vee I^-(x, u)$

Если истинно указанное высказывание, то говорят

$I^+(x, u)$ – дуга u исходит из вершины x;

$I^-(x, u)$ – дуга u заходит в вершину x;

$I^\circ(x, u)$ – u есть петля при вершине x.

При произвольном $u \in U$ говорят, что ребро u и вершина x ***инцидентны***, если $I(x, u)$ истинно, т.е. ребро соединяет вершину x хотя бы с одной вершиной y (может быть $x = y$).

Два ребра u, v называются ***смежными*** , если существует хотя бы одна инцидентная им обоим вершина, т.е. если истинно высказывание

$\exists x[I(x, u) \ \wedge \ I(x, v)]$.

З а м е ч а н и е. Определение 2 (стр. 193)

6.1.4 Части графа

Пусть $X' \subseteq X$ и $U' \subseteq U$ – подмножества вершин и ребер графа $G = (X, U; P)$, а P' – предикат, *индуцированный инцидентором* P на этих подмножествах. Если X' и U' выбраны так, что они вместе с P' удовлетворяют всем условиям определения графа, то граф $G' = (X', U'; P')$ называется *частью* графа G, порожденной подмножествами X' и U'. Очевидно, что включив в U' некоторое ребро графа, необходимо также включить в X' те вершины, которые соединяются этим ребром. К числу частей графа G относится и сам G. Остальные части – собственные или правильные. Предикат P', индуцированный предикатом P, будем далее обозначать P.

Наиболее часто рассматриваемые части имеют собственные названия.

Определение. *Часть $G' = (X', U'; P)$ называется подграфом графа G, порожденным подмножеством вершин $X' \subseteq X$, если при образовании части G' сохранены все те ребра, которые соединяют между собой сохраненные вершины.*

Определение. *Часть $G' = (X', U'; P)$ называется суграфом, порожденным подмножеством ребер $U' \subseteq U$, если $X' = X$.*

Части графа можно определить и так: введем следующие операции над элементами графа.

1. *Удаление вершины* с инцидентными ей ребрами:

$X' = X \setminus \{x\}, U' = U \setminus \{u | I(x, u)\}$.

2. *Удаление ребра*: $U' = U \setminus \{u\}, X' = X$.

Тогда подграф $G' = (X', U'; P)$ образуется из G удалением некоторых вершин;

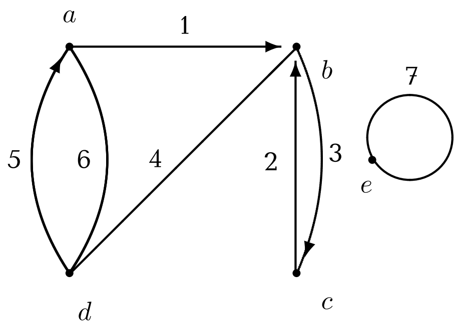
суграф $G' = (X, U'; P)$ образуется из G удалением некоторых ребер;

часть $G' = (X', U'; P)$ образуется из G удалением некоторых вершин и/или ребер (подграф, суграф, подграф суграфа, суграф подграфа).

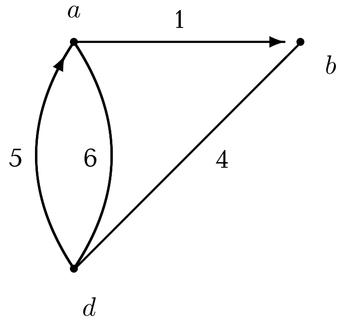
Сам граф по отношению к подграфу называется *надграфом*, по отношению к суграфу – *сверхграфом*, по отношению к части – *объемлющим графом* .

Пример 6.4.

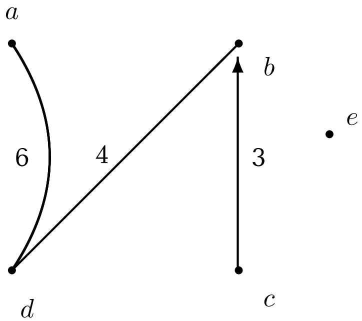
Граф $G = (X, U; P)$; $X = \{a, b, c, d, e\}$; $U = \{1, 2, \ldots, 7\}$;
$P = \{P(a, 1, b)$, $P(a, 6, d)$, $P(d, 6, a)$, $P(b, 3, c)$, $P(c, 2, b)$, $P(b, 4, d)$, $P(d, 4, b)$, $P(d, 5, a)$, $P(e, 7, e)\}$.



Подграф $G' = (X', U'; P')$; $X' = \{a, b, d\}$; $U' = \{1, 4, 5, 6\}$;
$P' = \{P(a, 1, b)$, $P(a, 6, d)$, $P(d, 6, a)$, $P(b, 4, d)$, $P(d, 4, b)$, $P(d, 5, a)\}$.



Суграф $G'' = (X'', U''; P)$; $X'' = \{a, b, c, d, e\}$; $U'' = \{3, 4, 6\}$;
$P'' = \{P(a, 6, d)$, $P(d, 6, a)$, $P(b, 3, c)$, $P(b, 4, d)$, $P(d, 4, b)\}$.



6.1.5 Изоморфизм графов

Определение. *Два графа $G = (X, U; P)$ и $G' = (X', U'; P')$ называются изоморфными , если между их вершинами, а также между*

их ребрами можно установить взаимно однозначное соответствие $X \leftrightarrow X', U \leftrightarrow U'$*, сохраняющее инцидентор* P*,*
то есть

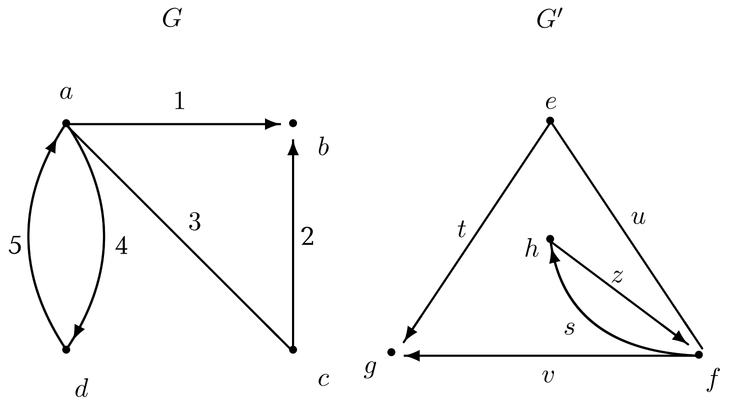
$$\begin{gathered}\forall x, y \in X \ \forall x', y' \in X' \ \forall u \in U \ \forall u' \in U' \\ \{x \leftrightarrow x' \ \& \ y \leftrightarrow y' \ \& \ u \leftrightarrow u' \ \Rightarrow \\ \Rightarrow \ [P(x,u,y) \ \Leftrightarrow \ P'(x',u',y')]\}.\end{gathered}$$

Здесь $\leftrightarrow$ обозначает взаимно-однозначное соответствие, $\Rightarrow$ – логическое следствие, $\Leftrightarrow$ – логическую эквивалентность.

Отношение изоморфизма рефлексивно, симметрично и транзитивно, то есть является отношением эквивалентности.

В дальнейшем нас будут интересовать в основном такие свойства графов, которые сохраняются при замене графа изоморфным ему графом. Тем самым оказываются несущественными как сама природа элементов, составляющих множества X и U, так и конкретный смысл предиката P. Но даже при таком абстрактном изучении графов требуется индивидуализация элементов. Поэтому всегда предполагается, что вершины и ребра графа либо сами являются определенными объектами (буквами, числами и т.п.), либо помечены различными индексами из каких-либо индексных множеств (обозначены буквами или пронумерованы). При этом два графа, отличающиеся лишь индексацией элементов, рассматриваются как изоморфные, но не как тождественные.

Пример 6.5.



Для графов, приведенных на рисунке:

$X = \{a, b, c, d\}; U = \{1, 2, 3, 4, 5\};$

$P = \{P(a,1,b), P(a,3,c), P(c,3,a), P(a,4,d), P(c,2,b),$
$P(d,5,a)\}$.
$G' = (X', U'; P'); X' = \{e,f,g,h\}; U' = \{s,t,u,v,z\}$;

$P' = \{P'(e,t,g), P'(e,u,f), P'(f,u,e), P'(f,v,g),$

$P'(f,s,h), P'(h,z,f)\}$.

Взаимно-однозначное соответствие устанавливается следующим образом:
$a \leftrightarrow f, b \leftrightarrow g, c \leftrightarrow e, d \leftrightarrow h$;
$1 \leftrightarrow v, 2 \leftrightarrow t, 3 \leftrightarrow u, 4 \leftrightarrow s, 5 \leftrightarrow z$;

$P(a,1,b) \Leftrightarrow P'(f,v,g), P(a,3,c) \& P(c,3,a) \Leftrightarrow$

$\Leftrightarrow P'(f,u,e) \& P'(e,u,f)$,

$P(a,4,d) \Leftrightarrow P'(f,s,h), P(c,2,b) \Leftrightarrow P'(e,t,g)$,

$P(d,5,a) \Leftrightarrow P'(h,z,f)$. ◄

6.2 Задание графов с помощью матриц

Чтобы задать конкретный граф, нужно задать пару множеств X, U и инцидентор P. Так как P – трехместный предикат (тернарное отношение), то в общем случае для задания графа нужна трехмерная матрица. Использование многозначной логики позволяет представить граф двумерной матрицей. Другой способ, равносильный применению пятизначной логики состоит в следующем.

6.2.1 Матрица инциденций

Пусть **K** – свободное полукольцо с нулем и образующими элементами

$$\textcircled{$>$},\ \textcircled{$<$},\ \textcircled{$\cdot$},\ \textcircled{$\sim$},\ 0.$$

Пусть задан граф $G = (X, U; P)$, $X = \{x_1, \ldots, x_n\}$, $U = \{u_1, \ldots, u_m\}$, $U \neq \varnothing$.
Определение. ***Матрицей инциденций*** *над полукольцом* ***K*** *графа* G *называется прямоугольная таблица*

$A = A(G) = \|a_{ij}\|, i = 1, \ldots, n,\ j = 1, \ldots, m,$

элементы которой a_{ij} принадлежат полукольцу $\boldsymbol{K}$ и определяются по графу G следующим образом:

если u_j – дуга, исходящая из вершины x_i, то $a_{ij} = \textcircled{$>$}$;

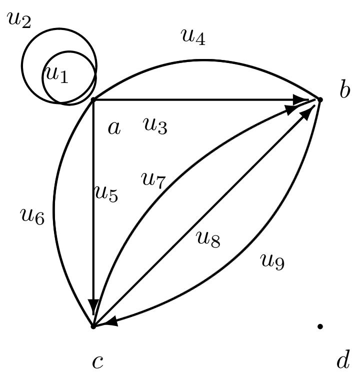
если u_j – дуга, заходящая в вершину x_i, то $a_{ij} = \textcircled{$<$}$;

если u_j – звено, инцидентное вершине x_i, то $a_{ij} = \textcircled{$\sim$}$;

если u_j – петля при вершине x_i, то $a_{ij} = \circledcirc$;

если ребро u_j и вершина x_i не инцидентны, то $a_{ij} = 0$.

Пример 6.6.



Для графа, изображенного на рисунке, матрица инциденций имеет следующий вид:

$A(G) =$

A	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
a	$\circledcirc$	$\circledcirc$	$\textcircled{$>$}$	$\textcircled{$\sim$}$	$\textcircled{$>$}$	$\textcircled{$\sim$}$	0	0	0
b	0	0	$\textcircled{$<$}$	$\textcircled{$\sim$}$	0	0	$\textcircled{$<$}$	$\textcircled{$<$}$	$\textcircled{$>$}$
c	0	0	0	0	$\textcircled{$<$}$	$\textcircled{$\sim$}$	$\textcircled{$>$}$	$\textcircled{$>$}$	$\textcircled{$<$}$
d	0	0	0	0	0	0	0	0	0

◄

6.2.2 Матрица соседства вершин

Определение. *Квадратная матрица $B(G) = B = AA^{\mathsf{T}}$ порядка $n = |X|$ называется матрицей соседства вершин.*

Пример 6.7.

Для графа из примера 6.6 получим:

$B(G) =$

B	a	b	c	d
a	$2\odot^2 + 2\textcircled{>}^2 + 2\textcircled{\sim}^2$	$\textcircled{>}\textcircled{<} + \textcircled{\sim}^2$	$\textcircled{>}\textcircled{<} + \textcircled{\sim}^2$	0
b	$\textcircled{>}\textcircled{<} + \textcircled{\sim}^2$	$3\textcircled{<}^2 + 2\textcircled{>}^2 + \textcircled{\sim}^2$	$2\textcircled{<}\textcircled{>} + \textcircled{>}\textcircled{<}$	0
c	$\textcircled{<}\textcircled{>} + \textcircled{\sim}^2$	$2\textcircled{<}\textcircled{>} + \textcircled{>}\textcircled{<}$	$2\textcircled{<}^2 + 2\textcircled{>}^2 + \textcircled{\sim}^2$	0
d	0	0	0	0

Здесь $2\odot^2$ означает $\odot \cdot \odot + \odot \cdot \odot$. (Знак умножения $\cdot$ мы договорились опускать ранее), $\textcircled{>}^2$ означает $\textcircled{>} \cdot \textcircled{>}$ и т.д. ◀

Из определения матрицы соседства вершин B следует, что ее элементы имеют вид

$b_{ii} = s^+(x_i)\,\textcircled{>}^2 + s^-(x_i)\,\textcircled{<}^2 + s^\circ(x_i)\,\odot^2 + s^\sim(x_i)\,\textcircled{\sim}^2$,

$i \neq j, b_{ij} = s^+(x_i, x_j)\,\textcircled{>}\textcircled{<} + s^-(x_i, x_j)\,\textcircled{<}\textcircled{>} + s^\sim(x_i, x_j)\,\textcircled{\sim}^2$,

где

$s^+(x_i)$ – число дуг, исходящих из вершины x_i;

$s^-(x_i)$ – число дуг, заходящих в вершину x_i;

$s^\circ(x_i)$ – число петель при вершине x_i;

$s^\sim(x_i)$ –число звеньев, инцидентных вершине x_i;

$s^+(x_i, x_j)$ – число дуг, идущих из x_i в x_j;

$s^-(x_i, x_j)$ – число дуг, идущих из x_j в x_i.

Число ребер, соединяющих вершины x, y графа:

$s(x, y) = s^+(x, y) + s^-(x, y) + s^\sim(x, y)$, если $x \neq y$;

$s(x, y) = s^\circ(x, y)$, если $x = y$. *Степень* вершины x обозначается $s(x)$ и определяется как

$s(x) = s^+(x) + s^-(x) + s^\circ(x) + s^\sim(x)$

(число ребер, инцидентных вершине x). *Валентность* $v(x)$ вершины x есть

$v(x) = s^+(x) + s^-(x) + 2s^\circ(x) + s^\sim(x)$

(число “усиков” при вершине).

Очевидно,

$$\sum_{x \in X} v(x) = 2|U| = 2m(G).$$

Число ребер, соединяющих вершину x с другими вершинами графа (т.е. число непетель, инцидентных x) есть

$$2s(x) - v(x) = s^+(x) + s^-(x) + s^\sim(x).$$

- Если $2s(x)-v(x)=0$, то вершина x называется *изолированной* (при вершине могут быть петли);
- Если $s(x)=0$ или $v(x)=0$, вершина x называется *голой* (нет никаких ребер, инцидентных x).
- Если $2s(x)-v(x)=1$, то вершина называется *висячей*.

При переходе от матрицы инциденций к матрице соседства вершин теряется индивидуализиция ребер графа, т.е. матрица B определяет граф G с точностью до нумерации ребер.

6.2.3 Матрица смежности

Определение. *Матрица смежности $R(G)=R=\|r_{ij}\|$ графа получается из матрицы соседства вершин $B(G)$ следующим образом:*

$$r_{ii} \;=\; s^{\circ}(x_i)\odot^2,$$

$$i \neq j \; r_{ij} \;=\; b_{ij}.$$

Переход к матрице смежности не приводит к дальнейшей потере информации о графе, так как

$$s^{+}(x) \;=\; \sum_{i=1}^{n} \; s^{+}(x, x_i),$$

$$s^{-}(x) \;=\; \sum_{i=1}^{n} \; s^{-}(x, x_i),$$

$$s^{\sim}(x) \;=\; \sum_{i=1}^{n} \; s^{\sim}(x, x_i).$$

Пример 6.8.
Из матрицы соседства вершин в предыдущем примере получим матрицу смежности вершин:

$R(G) =$

R	a	b	c	d
a	$2\odot^2$	$\textcircled{>}\textcircled{<}+\textcircled{\sim}^2$	$\textcircled{>}\textcircled{<}+\textcircled{\sim}^2$	0
b	$\textcircled{>}\textcircled{<}+\textcircled{\sim}^2$	0	$2\textcircled{<}\textcircled{>}+\textcircled{>}\textcircled{<}$	0
c	$\textcircled{<}\textcircled{>}+\textcircled{\sim}^2$	$2\textcircled{<}\textcircled{>}+\textcircled{>}\textcircled{<}$	0	0
d	0	0	0	0

◀

Очень редко используется *матрица соседства ребер*

$H(G)=H=A^TA$ порядка $m=|U|$.

Заметим лишь, что при переходе от матрицы инциденций к матрице соседства ребер теряется индивидуализация вершин, т.е. матрица H определяет граф с точностью до нумерации вершин. При этом может оказаться, что у неизоморфных графов матрицы H будут одинаковыми.

6.3 Типы графов

Классификация графов по типам проводится по различным его характеристикам.

Классификация по типу ребер.

- *Ориентированный* граф (*орграф*, диграф – *directed graph*) – это граф без звеньев: $\widetilde{U} = \varnothing$.
- *Неориентированный* граф: $\vec{U} = \varnothing$.
- Граф *без петель*: $\mathring{U} = \varnothing$. Возможны варианты: ориентированный граф без петель, неориентированный граф без петель.
- *Вырожденный* граф: $\widetilde{U} = \varnothing$, $\vec{U} = \varnothing$, $\mathring{U} \neq \varnothing$.
- *Пустой* граф: $U = \varnothing$.

Классификация по числу ребер, соединяющих пару вершин.

p-граф – это такой граф, в котором каждая пара вершин соединена не более чем p ребрами, т.е.

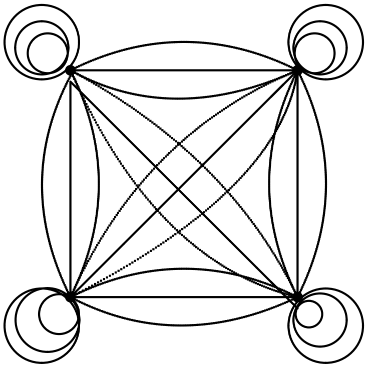
$\forall x, y \in X[s(x, y) \leqslant p]$.

При $p = 1$ p-граф называется *униграфом*, а при $p > 1$ – *мультиграфом*; при $p = 0$ граф *вырожденный* или *пустой*.

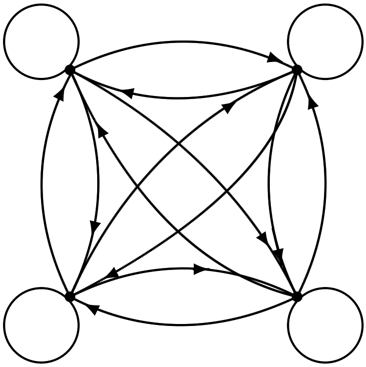
Граф общего вида, в котором все вершины попарно смежны, называется *плотным*.

Граф является *полным* в своем классе, если он содержит все ребра, возможные при принадлежности графа к данному классу. Например, p-граф будет полным, если при каждой вершине будет ровно p петель, а каждая пара различных вершин соединена ровно p ребрами (причем среди ребер могут быть как звенья, так и дуги любых направлений). В полном ориентированном униграфе без петель из каждой вершины в другую идет ровно одна дуга. Для графов общего вида понятие полноты не имеет смысла.

Примеры 6.9.
1. Полный 3-граф (на 4-х вершинах).



2. Полный ориентированный униграф.



З а м е ч а н и е. Если граф задан как бинарное отношение, то графы можно классифицировать по **типам** (свойствам) **бинарных отношений**. Так граф может быть рефлексивным, симметричным, транзитивным, графом порядка и т.д. Например, полный симметричный и рефлексивный граф есть полный ориентированный униграф.

Примером классификации по **топологии** может служить планарный (плоский) граф. Граф называется *планарным* (*плоским*), если его можно разместить на плоскости (на сфере) таким образом, чтобы никакие два ребра не пересекались.

Наконец, графы можно классифицировать по **мощности** множеств его элементов. Если мощности вершин и ребер графа $|X|, |U|$ конечны, то граф – *конечный*, в противном случае – *бесконечный*. Мы будем рассматривать только конечные графы. ▶

6.3.1 Обыкновенные графы

Очень важную роль в теории графов и ее приложениях играют неориентированные униграфы без петель – *обыкновенные* графы. Элементы матрицы соседства вершин обыкновенного графа $G = (X, U; P)$ имеют вид:

$b_{ii} = s^{\sim}(x_i)\circledS^2$, где $s^{\sim}(x_i) = \sum\limits_{j=1}^{n} s^{\sim}(x_i, x_j)$,

$i \neq j,\ b_{ij} = s^{\sim}(x_i, x_j)\circledS^2$,

$s^{\sim}(x_i, x_j) = s^{\sim}(x_j, x_i) = 1$, если вершины x_i, x_j смежны,

$s^{\sim}(x_i, x_j) = s^{\sim}(x_j, x_i) = 0$, в противном случае.

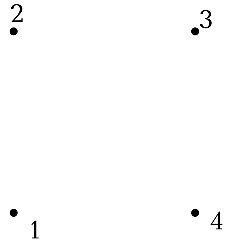
Элементы матрицы смежности R будут, соответственно, такими:

$r_{ii} = 0$,

$i \neq j, r_{ij} = b_{ij}$.

Мы не потеряем никакой информации о графе, если наложим на полукольцо $\boldsymbol{K}$ определяющее соотношение $\circledS^2 = 1$ (не обязательно это делается всегда). Таким образом, обыкновенный граф можно определить как множество с заданным на нем бинарным симметричным антирефлексивным отношением смежности. При этом можно установить взаимно-однозначное соответствие между множеством всех ребер и множеством всех неупорядоченных пар смежных вершин, т.е. каждое ребро может быть представлено парой смежных вершин, как это и делается на практике. Мы будем обозначать ребра обыкновенного графа в виде пары (x, y) , либо $\widetilde{xy}$. Сам граф G будем обозначать $G = (X, U)$, указывая этим, что инцидентор P полностью определяется заданием множеств X и U . При этом U можно рассматривать как бинарное отношение смежности $U \subseteq X \times X$.

Пример 6.10.



$X = \{1, 2, 3, 4\}$, $U = \{\widetilde{12}, \widetilde{13}, \widetilde{23}, \widetilde{34}\}$, или

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

Матрицы соседства вершин и смежности для нашего примера

$B =$

B	1	2	3	4
1	2	1	1	0
2	1	2	1	0
3	1	1	3	1
4	0	0	1	1

$R =$

R	1	2	3	4
1	0	1	1	0
2	1	0	1	0
3	1	1	0	1
4	0	0	1	0

◄

Обыкновенный граф будет *полным* (или *плотным*, что в данном случае одно и то же), если все его вершины попарно смежны. Полный и пустой обыкновенные n-вершинные графы обозначим соответственно F_n и E_n.

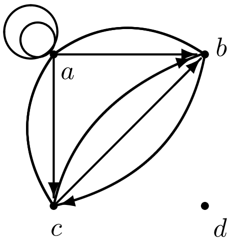
Если при исследовании графа G общего вида нужна не вся информация о нем, а лишь о смежности вершин, то от G можно перейти к его *скелету*, задав определяющие соотношения

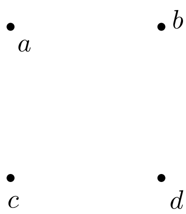
$\textcircled{>}\textcircled{<} = \textcircled{<}\textcircled{>} = \textcircled{\sim}^2$; $2\,\textcircled{\sim}^2 = \textcircled{\sim}^2$, т.е. $\textcircled{\sim}^2 + \textcircled{\sim}^2 = \textcircled{\sim}^2$, $\textcircled{\circ}^2 = 0$.

Кроме того, можно положить $\textcircled{\sim}^2 = 1$.

Граф G плотный, если его скелет – полный.

Пример 6.11.





Исходный граф $G = (X, U; P)$, его скелет $\widetilde{G}= (X, U)$. ◄

6.3.2 Графы Бержа

Определение. *Ориентированный униграф (1-орграф) называется графом Бержа.*

В нем нет звеньев и

$s^+(x_i, x_j) \leqslant 1, s^-(x_i, x_j) \leqslant 1, s^\circ(x_i) \leqslant 1.$

Налагая на полукольцо $\boldsymbol{K}$ определяющие соотношения

$\textcircled{<}\textcircled{>}= 0$, $\textcircled{>}\textcircled{<}=\textcircled{\circ}= 1.$

получим такую матрицу смежности R для графа Бержа

$$R = \begin{Vmatrix} s^\circ(x_1) & s^+(x_1, s_2) & \cdots & s^+(x_1, x_n) \\ s^+(x_2, x_1) & s^\circ(x_2) & \cdots & s^+(x_2, x_n) \\ \cdots & \cdots & \cdots & \cdots \\ s^+(x_n, x_1) & s^+(x_n, x_2) & \cdots & s^\circ(x_n) \end{Vmatrix}$$

$s^\circ(x_i) = 1$, если при вершине x_i есть петля,

$s^\circ(x_i) = 0$, если петли нет,

$s^+(x_i, x_j) = 1$, если есть дуга, идущая из x_i в x_j, $i \neq j$;

$s^+(x_i, x_j) = s^-(x_j, x_i) = 0$, если такой дуги нет.

В этом случае (при таких определяющих соотношениях, налагаемых на полукольцо $\boldsymbol{K}$) граф Бержа определяется однозначно с точностью до индивидуализации ребер заданием на X бинарного отношения

$\overrightarrow{J}\ (x, y) \ \Leftrightarrow\ \exists u\ P(x, u, y)$, или

$x\ \overrightarrow{J}\ y \ \Leftrightarrow\ \exists u\ P(x, u, y).$

Если задано отношение $\overrightarrow{J}$, то можно определить множество ребер U:

$U = \{\overrightarrow{xy}\ \mid x, y \in X\ \&\ x\ \overrightarrow{J}\ y\}$

(здесь мы обозначили $\overrightarrow{xy} =\ \langle x, \rangle$ – упорядоченную пару) ;

$P(x, u, y) \ \Leftrightarrow\ u =\overrightarrow{xy}\ \ \&\ u \in U.$

Граф Бержа мы также (как и обыкновенный) будем обозначать

$G = (X, U)$, или $G = (X, \vec{U})$,

(чтобы подчеркнуть, что граф – ориентированный). При этом U (или $\vec{U}$) можно рассматривать как бинарное отношение

$\vec{J} = U = \vec{U} \subseteq X \times X$.

Сам К.Берж вместо отношения $\vec{J}$ задает отображение ΓX в X, которое ставит в соответствие каждой вершине $x \in X$ подмножество (возможно, пустое)

$\Gamma x = \{y | y \in X \ \& \ \vec{J}\ (x, y)\}$.

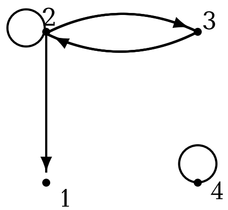
При этом способе задания графа Бержа $G = (X, U)$

$U = \{\vec{xy}\ | x, y \ \in \ X, y \ \in \ \Gamma x\}$;

$P(x, u, y) \ \Leftrightarrow \ u = \vec{xy} \ \ \& \ y \ \in \ \Gamma \ x$.

Граф Бержа можно обозначать и через $G = (X, \Gamma)$.

Пример 6.12.



$X = \{1, 2, 3, 4\}, U = \{\vec{21}, \vec{22}, \vec{23}, \vec{32}, \vec{44}\}$,

$\Gamma 1 = \varnothing, \Gamma 2 = \{1, 2, 3\}, \Gamma 3 = \{2\}, \Gamma = \{4\}$. ◀

В дальнейшем для удобства мы будем обозначать *окружение* x (множество вершин, смежных с x) через Γx для графов любых типов.

Так как граф Бержа задается как бинарное отношение, то он и обладает свойствами бинарных отношений, т.е. может быть рефлексивным, симметричным, асимметричным и т.д.

Асимметричный граф Бержа ($\forall x, y[y \in \Gamma x \ \ \Rightarrow \ \ x \notin \Gamma \ y \ ]$) можно назвать также *обыкновенным орграфом*, так как в результате *дезориентации* всех дуг он превращается в обыкновенный граф.

Дезориентация дуг орграфа заключается в наложении на образующие полукольца $\boldsymbol{K}$ следующих определяющих соотношений:

$$> \, < = 1, < \, > = 0, \sim^2 = 1, 2 = 1, \text{ (т.е.} 1 + 1 = 1),$$

то есть в переходе к булевой алгебре $\mathbf{B}\{0, 1\}$).

Матрица смежности $R(\widetilde{G})$ над $\mathbf{B}$ неориентированного графа $G = (X, U)$ получается из матрицы смежности $R(\overrightarrow{G})$ орграфа $\overrightarrow{G} = (X, \overrightarrow{U})$ следующим образом:

$$R(\widetilde{G}) = R(\overrightarrow{G}) \cdot R^{\mathsf{T}}(\overrightarrow{G}) \text{ (}^{\mathsf{T}} \text{ – знак транспонирования).}$$

6.3.3 Двудольные (бихроматические) графы

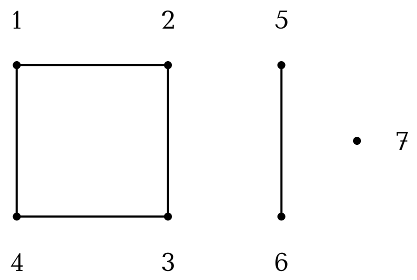
Обыкновенный граф $G = (X, U)$ называется *двудольным* (бихроматическим, графом Кенига, простым), если множество его вершин X можно представить в виде объединения двух непересекающихся подмножеств X_1 и X_2, таких, что никакие вершины одного и того же множества не смежны, т.е.

$X = X_1 \cup X_2, \; X_1 \cap \; X_2 = \varnothing,$

$\forall x, y \in X[\widetilde{xy} \in U \Rightarrow (x \in X_1 \& y \in X_2) \vee (x \in X_2 \& y \in X_1)].$

Двудольный граф записывают в виде $G = (X_1, X_2, U)$. Тем самым задается и конкретный граф.

Пример 6.13.



Граф, представленный на рисунке, может быть задан как двудольный четырьмя различными способами:

$G_1 = (\{1, 3, 5, 7\}, \{2, 4, 6\}, U); G_2 = (\{1, 3, 6, 7\}, \{2, 4, 5\}, U);$

$G_3 = (\{1, 3, 5\}, \{2, 4, 6, 7\}, U); G_4 = (\{1, 3, 6\}, \{2, 4, 5, 7\}, U);$

Матрица смежности двудольного графа полностью определяется своей прямоугольной подматрицей, строки которой соответствуют вершинам X_1, а столбцы – вершинам X_2.

Для нашего примера G_1 :

$$R = \begin{array}{|c|ccc|} \hline R & 2 & 4 & 6 \\ \hline 1 & 1 & 1 & 0 \\ 3 & 1 & 1 & 0 \\ 5 & 0 & 0 & 1 \\ 7 & 0 & 0 & 0 \\ \hline \end{array} \blacktriangleleft$$

Задание двудольного графа $G = (X_1, X_2, U)$ равносильно заданию двух множеств X_1 и X_2 и многозначного отображения Г, которое ставит в соответствие каждой вершине $x \in X$ подмножество (возможно, пустое) $\Gamma\, x \subseteq X_2$ всех смежных с x вершин $y \in X_2$.

Двудольный граф называется *полным*, если каждая вершина множества X_1 смежна с каждой вершиной множества X_2.

Такой граф определяется с точностью до изоморфизма, сохраняющего каждое из подмножеств X_1, X_2 заданием упорядоченной пары чисел $\mid X_! \mid$, $\mid X_2 \mid$. Полный двудольный граф обозначается $K_{|X_1|,|X_2|}$.

Например, $K_{3,4}$ – полный двудольный граф с $|X_1| = 3, |X_2| = 4$.

6.3.4 Помеченные и взвешенные графы

Задавая на вершинах и на ребрах графа $G = (X, U; P)$ функции

$f : X \mapsto L$,

$g : U \mapsto C$,

где L и C – произвольные множества, получим *помеченный* граф

$G[f, g] = (X, U; P; f, g)$.

На множествах X и U можно задать и более, чем по одной функции, или, наоборот, можно задать функцию, например, только на ребрах. Значения функций f и g будем называть *метками*. В том случае, если значения функций есть численные значения, будем называть граф *взвешенным*, а сами метки – *весами*.

Примеры 6.14.
1. Карта дорог : L – названия населенных пунктов, C – расстояния между ними.

2. Семантическая сеть : L – имена объектов, C – имена отношений между ними.

3. Потоки в сетях: каждой дуге сети ставится в соответствие ее пропускная способность и величина потока по дуге.

6.4 Другие способы задания графа

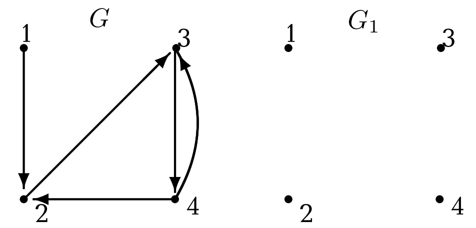
Представления графов с помощью матриц инциденций и матриц смежности, при всех их достоинствах, не лишены и ряда недостатков. Рассмотрим матрицы инциденций для графа Бержа (1-орграфа) и обыкновенного графа. Для 1-орграфа без петель можно положить

$\textcircled{>} = 1,\ \textcircled{<} = -1,\ \textcircled{\cdot} = 0,\ \textcircled{\sim} = 0,$

а для обыкновенного графа

$\textcircled{>} = 1,\ \textcircled{<} = 1,\ \textcircled{\sim} = 1,\ \textcircled{\cdot} = 0.$

Пример 6.15.



Для графа Бержа G и обыкновенного графа G_1 матрицы инциденций имеют следующий вид:

$A =$

A	$\overrightarrow{12}$	$\overrightarrow{13}$	$\overrightarrow{34}$	$\overrightarrow{41}$	$\overrightarrow{43}$
1	1	1	0	-1	0
2	-1	0	0	0	0
3	0	-1	1	0	-1
4	0	0	-1	1	1

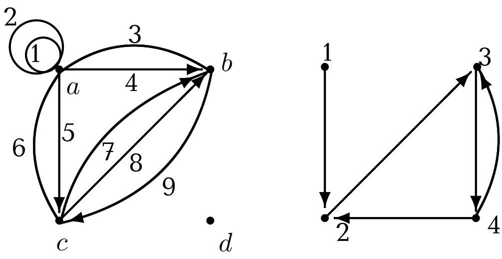
$$A_1 = \begin{array}{|c|cccc|}
\hline
A_1 & \widetilde{12} & \widetilde{13} & \widetilde{14} & \widetilde{12} \\
\hline
1 & 1 & 1 & 1 & 0 \\
2 & 1 & 0 & 0 & 0 \\
3 & 0 & 1 & 0 & 1 \\
4 & 0 & 0 & 1 & 1 \\
\hline
\end{array} \blacktriangleleft$$

Размерность матрицы инциденций равна $n \times m$, а число ненулевых элементов составляет всего $2 \times m$. Таким образом, $(n-2) \times m$ элементов матрицы – нулевые, и, следовательно, такое представление очень неэкономно. Кроме того, поиск смежных вершин или дуги, соединяющей пару вершин, приводит к перебору в худшем случае m столбцов матрицы.

Представление графа матрицей смежности также неэкономно в том случае, когда число ребер значительно меньше n^2 (размерность матрицы смежности). Однако пользоваться матрицей смежности во многих случаях очень удобно, так как за одну проверку можно определить смежность пары вершин.

Более экономным, особенно для неплотных графов, когда m значительно меньше n^2 , является способ представления графа с помощью списка или массива пар вершин, соответствующих его ребрам (если граф можно представить в виде бинарного отношения), или с помощью списка или массива троек вида $[x \ u \ y]$, где $x, y \ \in \ X, u \in U$, для графов общего вида. Размерность такого представления – до $2 \times m$ (если все ребра – звенья). Неудобством является большое число шагов поиска – до m в худшем случае – вершин смежных данной вершине. Число шагов поиска смежных вершин или ребер можно значительно уменьшить, упорядочив множество пар или троек в лексикографическом порядке и применяя двоичный поиск.

Пример 6.16.



G_1 G_2

Граф Бержа G_2 представлен в виде массива пар, упорядоченного в лексикографическом порядке:

$[(1,2),(2,3),(3,4),(4,2),(4,3)].$

Граф общего вида G_1 представлен в виде массива троек, также упорядоченного в лексикографическом порядке:

$[(a,1,a),(a,2,a),(a,3,b),(a,4,b),(a,5,c),(a,6,c),$

$(b,4,a),(b,9,c),(c,6,a),(c,7,b),(c,8,b)].$

Заметим, что звенья (вместе с парой инцидентных им вершин) в массиве представлены дважды.

В некоторых случаях граф удобно представлять в виде списков смежности. Каждый список содержит саму вершину x и подсписок, представляющий ее окружение Γx. Размерность такого представления равна $m+n$.

Граф G_2, например, можно представить в виде такого списка:

$[1,[2]],[2,[3]],[3,[4]],[4,[2,3]].$ ◄

З а м е ч а н и е. Напомним, что последним элементом списка и, следовательно, любого подсписка является (и подразумевается по умолчанию) пустой список **nil** или $[\ ]$.

Описанные выше способы представления графа аналогичны способам представления множества: представление графа с помощью матриц есть задание характеристических функций (функций принадлежности), а представление в виде массивов и списков – это перечисление элементов графа.

Возможен и дескриптивный (предикатный, высказывательный) способ представления графа.

Выбор способа представления является существенной частью разработки алгоритма решения конкретной задачи и всегда является компромиссом между размерностью представления и эффективностью алгоритма.

Глава 7

Связность графов

7.1 Маршруты, цепи, циклы

Предварительные замечания. В этом разделе рассматриваются такие свойства графов, которые не зависят от ориентации, то есть такие, которые полностью определяются в терминах полуинцидентора

$$\widetilde{P}\,(x,u,y) = P(x,u,y)\ \vee\ P(y,u,x).$$

Будем обозначать граф общего вида как $G = (X, U)$, предполагая заданным полуинцидентор $\widetilde{P}$.

Об индексах. Обозначение вершины x_i не обязывает нас считать индекс i номером вершины в множестве X; вершины с индексами i и j не обязательно различны при $i \neq j$. Индексы обозначают порядковый номер соответствующего элемента в маршруте.

Определение. *Конечная последовательность*

$$x_0u_1x_1u_2x_2\dots x_{l-1}u_lx_l\ (l \geq 0)$$

элементов графа G, для которой истинно высказывание

$$\widetilde{P}\,(x_0,u_1,x_1)\ \&\ \ \widetilde{P}\,(x_1,u_2,x_2)\ \&\ \dots\ \&\ \ \widetilde{P}\,(x_{l-1},u_l,x_l)$$

называется маршрутом из вершины x_0 в вершину x_l, или маршрутом, соединяющим вершину x_0 с вершиной x_l.

В случае $x_0 = x_l$ маршрут называется *циклическим*. Число l называется *длиной* маршрута.

Маршрут не является частью графа, так как порядок его обхода имеет существенное значение. Так, маршрут

$$x_lu_lx_{l-1}\ \dots\ x_2u_2x_1u_1x_0$$

при $l \neq 0$ не совпадает с приведенным в определении, хотя состоит из тех же элементов и с той же инцидентностью.

7.1.1 Число маршрутов

Рассмотрим матрицу смежности R над полукольцом $\boldsymbol{K}$со следующими определяющими соотношениями

$$\textcircled{>}\textcircled{<} = \textcircled{<}\textcircled{>} = \circledcirc^2 = \textcircled{\sim}^2 = 1, 1+1=2$$

(обычное арифметическое сложение). Таким образом, $\boldsymbol{K}$содержит подполукольцо $\mathbf{K}'$ целых неотрицательных чисел с обычными сложением и умножением.

Элемент r_{ij} матрицы смежности в этом случае определяет число ребер $s(x_i, x_j)$, соединяющих вершины x_i, x_j.

Рассмотрим далее l-ю степень

$$R^l = \|r_{ij}^{(l)}\|, l = 1, 2, \ldots.$$

матрицы смежности R графа G.

Лемма 1 *Элемент $r_{ij}^{(l)}$ равен числу различных маршрутов длины l из вершины x_i в вершину x_j.*

Д о к а з а т е л ь с т в о . При $l = 1$ это утверждение очевидно. В общем случае утверждение доказывается индукцией по l. Если известно, что $r_{ik}^{(l-1)}$ – число различных маршрутов длины $l-1$ из вершины x_i в вершину x_k, то для числа маршрутов длины l из вершины x_i в вершину x_j имеем (для фиксированной вершины x_k)

$$r_{ik}^{(l-1)} \cdot r_{kj},$$

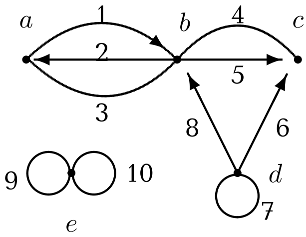
а для всех k число маршрутов равно

$$\sum_{k=1}^{n} r_{ik}^{(l-1)} \cdot r_{kj},$$

а это ни что иное, как $r_{ij}^{(l)}$ – элемент матрицы R^l. $\boxtimes$

Пример 7.1.

Рассмотрим степени матрицы смежности графа, представленного на рисунке:



$R =$

R	a	b	c	d	e
a	0	3	0	0	0
b	3	0	2	1	0
c	0	2	0	1	0
d	0	1	1	1	0
e	0	0	0	0	2

$R^2 =$

R^2	a	b	c	d	e
a	9	0	6	3	0
b	0	14	1	3	0
c	6	1	5	3	0
d	3	3	3	3	0
e	0	0	0	0	4

$R^3 =$

R^3	a	b	c	d	e
a	0	42	3	9	0
b	42	5	31	18	0
c	3	31	5	9	0
d	9	18	9	9	0
e	0	0	0	0	8

Например, из вершины c в c идут 5 маршрутов длины 2:

$c\ 4\ b\ 4\ c$, $c\ 5\ b\ 5\ c$, $c\ 4\ b\ 5\ c$, $c\ 5\ b\ 4\ c$, $c\ 6\ d\ 6\ c$;

из c в d идут 3 маршрута длины 2:

$c\ 4\ b\ 8\ d$, $c\ 5\ b\ 8\ d$, $c\ 6\ d\ 7\ d$;

из d в c идут 9 маршрутов длины 3:

$d\ 6\ c\ 6\ d\ 6\ c$, $d\ 6\ c\ 4\ b\ 4\ c$, $d\ 6\ c\ 4\ b\ 5\ c$, $d\ 6\ c\ 5\ b\ 4\ c$, $d\ 6\ c\ 5\ b\ 5\ c$,

$d\ 7\ d\ 7\ d\ 6\ c$, $d\ 7\ d\ 8\ b\ 4\ c$, $d\ 7\ d\ 8\ b\ 5\ c$, $d\ 8\ b\ 8\ d\ 6\ c$. ◀

Если нас интересует только достижимость j-й вершины из i-й за число шагов l (то есть только наличие маршрута длины l из вершины x_i в вершину x_j), добавим к определяющим соотношениям в полукольце $\boldsymbol{K}$ еще булево $2 = 1$ (то есть $1 + 1 = 1$). Тогда подполукольцо $\mathbf{K}' \subset \boldsymbol{K}$ станет булевой алгеброй $\mathbf{B}\{0, 1\}$. При этом элемент $r_{ij}^{(l)}$ матрицы R^l будет равен 1, если существует хотя бы один маршрут длины l из вершины x_i в вершину x_j, и 0, если не существует ни одного такого маршрута.

Пример 7.2.

Для графа из предыдущего примера имеем:

$R =$

R	a	b	c	d	e
a	0	1	0	0	0
b	1	0	1	1	0
c	0	1	0	1	0
d	0	1	1	1	0
e	0	0	0	0	1

$R^2 =$

R^2	a	b	c	d	e
a	1	0	1	1	0
b	0	1	1	1	0
c	1	1	1	1	0
d	1	1	1	1	0
e	0	0	0	0	1

$R^3 =$

R^3	a	b	c	d	e
a	0	1	1	1	0
b	1	1	1	1	0
c	1	1	1	1	0
d	1	1	1	1	0
e	0	0	0	0	1

$R^4 = R^5 =$

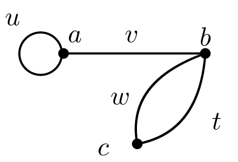
R^4	a	b	c	d	e
a	1	1	1	1	0
b	1	1	1	1	0
c	1	1	1	1	0
d	1	1	1	1	0
e	0	0	0	0	1

◄

Для нахождения всех маршрутов данной длины, идущих из одной заданной вершины в другую, можно использовать *модифицированную* матрицу смежности. Продемонстрируем это на примере.

Пример 7.3.

Для графа, изображенного на рисунке, построим такую матрицу смежности:



$R_u =$

R	a	b	c
a	u	v	0
b	v	0	$w+t$
c	0	$w+t$	0

Элемент r_{ij} матрицы R_u заменен на ребро, соединяющее вершины x_i с x_j; если таких ребер несколько, то r_{ij} будет суммой ребер.

Таким образом, символы u, v, w, t из примера можно рассматривать как образующие элементы нового некоммутативного кольца $\mathbf{K}_u$, которое будем считать ассоциативным и дистрибутивным. Теперь последовательно образуем степени матрицы R_u:

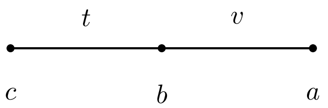
$R_u^2 =$

R	a	b	c
a	u^2+v^2	uv	$vw+vt$
b	vu	$v^2+w^2+wt+tw$	0
c	$wv+tv$	0	$w^2+t^2+wt+tw$

Поясним смысл, например, элемента $r^2_{ac} = vw + vt$: из вершины a в вершину c имеется два маршрута длины 2 – $[avbwc]$ и $[avbtc]$. ◄

Этот способ нахождения всех маршрутов более нагляден и прост в реализации, если вместо матричного представления графа использовать представление в виде массива
$\{a\ u\ a,\ \ a\ v\ b,\ \ b\ v\ a,\ \ b\ t\ c,\ \ b\ w\ c,\ \ c\ t\ b,\ \ c\ w\ b\}$.
Тогда “2-ю степень" аналога матрицы смежности можно представить как массив
$\{a\ u\ a\ u\ a,\ a\ u\ a\ v\ b,\ a\ v\ b\ v\ a,\ a\ b\ t\ c,\ a\ v\ b\ v\ c,\ b\ v\ a\ u\ a,\ b\ v\ a\ v\ b,$ $b\ t\ c\ t\ b,\ b\ t\ c\ w\ b,\ b\ w\ c\ t\ b,\ b\ w\ c\ w\ b,\ c\ t\ b\ v\ a,\ c\ t\ b\ t\ c,\ c\ t\ b\ w\ c,$ $c\ w\ b\ v\ a,\ c\ w\ b\ t\ c,\ c\ w\ b\ w\ c\}$.

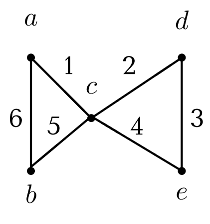
Как видно из примера, маршрут длины l образуется в том случае, если начало l-ой тройки маршрута (например, $b\ v\ a$), совпадает с концом маршрута длины $l-1$ (например, $c\ t\ b$).



Следующую степень получим, умножая аналог 2-й степени матрицы смежности на аналог первой степени матрицы смежности.

Определение. *Маршрут* $x_0u_1x_1u_2x_2\ldots x_{l-1}u_lx_l$ *называется цепью, если все ребра в нем различны. Циклическая цепь называется циклом.*

Пример 7.4.



Примеры циклов: $[a1c4e3d2c5b6c]$, $[b6a1c5b]$, $[c2d3e4c]$. ◄

Определение. *Цепь называется простой, если все ее вершины различны.*

Цикл называется простым, если все его вершины различны, кроме $x_0 = x_l$.

Если нужно выявить только простые цепи (с помощью R), то после каждого умножения нужно вычеркивать те слагаемые, в которых сомножители встречаются более одного раза (т.е. элементы в квадрате).

Пример 7.5. Для нашего графа

$$R_u^2 = \begin{array}{|c|ccc|} \hline R^2 & a & b & c \\ \hline a & 0 & uv & vw+vt \\ b & vu & wt+tw & 0 \\ c & wv+tv & 0 & wt+tw \\ \hline \end{array} \quad ◄$$

Лемма 2 *Всякий маршрут (в частности, всякая цепь) графа содержит хотя бы одну простую цепь, соединяющую ту же пару вершин.*

Всякий циклический маршрут нечетной длины содержит простой цикл нечетной длины.

Всякий цикл содержит простой цикл.

Доказательство. 1. Если в данном маршруте

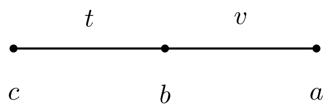
$$x_0 u_1 x_1 \ldots x_{l-1} u_l x_l$$

все вершины различны, то он сам и есть искомая простая цепь. Если x_i – первая из тех вершин маршрута, которая входит в него более одного раза, а x_j – последняя из совпадающих с x_i вершин этого маршрута, то его можно заменить более коротким

$$x_0 u_1 x_1 \ldots x_i u_{j+1} x_{j+1} u_{j+2} \ldots x_{l-1} u_l x_l.$$

Если в полученном маршруте есть еще повторяющиеся вершины, то снова заменяем его более коротким и так далее, пока не выделим маршрут без повторяющихся вершин, соединяющий x_0 с x_l, то есть искомую простую цепь (если исходный маршрут циклический, то эта цепь будет состоять из единственной вершины $x_0 = x_l$).

2. Рассматривается циклический маршрут нечетной длины потому, что циклический маршрут четной длины может не содержать простого цикла; например,



циклический маршрут четной длины $[c\,t\,b\,v\,a\,v\,b\,t\,c]$ не содержит простого цикла.

Итак пусть

$x_0 u_1 x_1 \ldots x_{2k} u_{2k+1} x_0$

циклический маршрут нечетной длины. Тогда этот маршрут либо сам является простым циклом, либо содержит простой цикл нечетной длины. Действительно, пусть x_i, x_j такие, что $0 \le i < j \le 2k$ и $x_i = x_j$, тогда маршруты

$x_0 u_1 x_1 \ldots x_{i-1} u_i x_i u_{j+1} x_{j+1} \ldots x_{2k} u_{2k+1} x_0$

и

$x_i u_{i+1} x_{i+1} \ldots x_{j-1} u_j x_j$

оба циклические и один из них имеет нечетную длину. С этим маршрутом поступаем точно так же (если это не простой цикл) и так до тех пор, пока не выделим простой цикл нечетной длины.

3. Доказательство третьего утверждения аналогично доказательству 2-го; заметим только, что заменить цикл в условии произвольным циклическим маршрутом нельзя. ⊠

Следствие. *Всякий кратчайший маршрут между двумя заданными вершинами графа есть простая цепь.*
Всякий цикл наименьшей длины при заданной вершине является простым.

7.2 Теорема Кёнига

Теорема 7.1 *(Кёнига) Обыкновенный граф $G = (X, U)$ является двудольным (бихроматическим) тогда и только тогда, когда он не содержит циклических маршрутов нечетной длины.*

Доказательство. 1. $\Rightarrow$ *Необходимость.* Пусть G – бихроматический, т.е. $X = X_1 \cup X_2,\ X_1 \cap X_2 = \emptyset$ и вершины каждого из них попарно несмежны. У любого маршрута

$$x_0 u_1 x_1 \dots x_{l-1} u_l x_l$$

вершины должны попеременно входить то в X_1, то в X_2; совпадение $x_l = x_0$ возможно лишь при четном l.

2. $\Leftarrow$ *Достаточность.* Пусть теперь $G = (X, U)$ – граф без циклических маршрутов нечетной длины.Будем рассматривать только связный G. Для несвязного G доказательство проводится по каждой компоненте.

Используем для пометки вершин знаки $+$ и $-$. Выберем любую вершину и пометим ее знаком $+$. Затем выполним следующую итеративную процедуру. Выберем одну из помеченных вершин x и пометим все вершины множества Γx (т.е. все смежные с x вершины) знаком противоположным тому, которым помечена вершина x. Продолжаем разметку до тех пор, когда:

a) все вершины помечены, причем любые две смежные вершины помечены разными знаками;

b) некоторая вершина x , которая уже была помечена каким-то знаком ($+$ или $-$) может быть помечена со стороны другой вершины другим знаком.

В первом случае все вершины, помеченные знаком $+$ отнесем к множеству X_1, а помеченные знаком $-$ – к множеству X_2. Так как все ребра соединяют вершины из разных множеств, то граф двудольный.

Во втором случае вершина x должна быть помечена знаком $+$ на некотором маршруте $P_1 = x_1 x_2 \dots x$, причем знак $+$ и $-$ должны образовывать на P_1 чередующиеся последовательности вида "$+, -, + \dots$"или "$-, +, - \dots$". Знаком $-$ вершина x помечена вдоль некоторого маршрута P_2 . Пусть y – предпоследняя (последняя– это x) общая вершина маршрутов P_1 и P_2. Если вершина y помечена знаком $+$, то число ребер от y до x в маршруте P_1 должно быть четным, а в маршруте P_2 – нечетным. Следовательно, циклический маршрут от y до x по маршруту P_1 и от x до y по маршруту P_2 имеет нечетную длину. Это противоречит предположению о том, что G не содержит циклических маршрутов нечетной длины, т.е. второй случай невозможен. Теорема доказана. ⊠

Подмножества X_1 и X_2 образуют классы разбиения; никакие две вершины одного класса не смежны – в противном случае соединяющее их ребро вместе с соединяющим их маршрутом четной длины образовало бы циклический маршрут нечетной длины.

Следствие. *Граф $G = (X, U)$ является бихроматическим тогда и только тогда, когда он не содержит простых циклов нечетной длины.*

7.3 Компоненты связности

Если маршрут рассматривать с учетом ориентации ребер (может быть и по звеньям), т.е. в определении маршрута

$x_0 u_1 x_1 \ldots x_{l-1} u_l x_l$
считаем

$P(x_0, u_1, x_1)\ \&\ \ldots\ \&\ P(x_{l-1}, u_l, x_l)$,
получим соответствующие определения:

- *частично ориентированный* маршрут;
- частично ориентированная цепь;
- частично ориентированный цикл.

Если же все u_i – дуги, то все маршруты, соответственно, *ориентированные*. Обычно при ссылке на ориентированные маршруты для краткости используют префикс "ор" – ормаршрут, орцепь, орцикл, в том числе и простые: простая орцепь, простой орцикл и т.д. Часто встречается термин *путь* – орцепь (простой путь – простая орцепь).

Вопросы, которые мы сейчас рассмотрим применимы, по отдельности, к неориентированным и ориентированным графам.

Для неориентированных графов

Определение. *Вершины x, y графа G называются отделенными , если в G не существует никакой соединяющей их цепи, и неотделенными, если хотя бы одна такая цепь существует.*

Отношение неотделенности вершин

- рефлексивно (каждая вершина соединена с собой цепью нулевой длины);
- симметрично (цепь, записанная в обратном порядке – тоже цепь);

- транзитивно (если существует цепь из x в y, и цепь из y в z, то существует и цепь из x в z).

Таким образом, отношение неотделенности есть отношение эквивалентности на множестве вершин X; при этом X разбивается на классы попарно неотделенных вершин $X_1, X_2, \ldots X_\kappa$.

Определение. *Подграфы $G_i = (X_i, U_i; P)$, порожденные подмножествами X_i, не имеют общих вершин и ребер и называются компонентами связности (или просто компонентами).*

Число их обозначается через $\kappa(G)$ (каппа). Если $\kappa(G) = 1$, то граф называется *связным*. Другая крайность – *вырожденный* граф: число его компонент равно числу вершин.

Отношение "быть в одной компоненте" для ребер – эквивалентность, как и для вершин.

Для ориентированных графов.

Вершина y *достижима* из вершины x, если существует путь из x в y. Высказывание "y достижима из x" обозначим через $D(x, y)$.

Вершины x и y *взаимодостижимы* , если истинно высказывание

$D(x, y)$ & $D(y, x)$.

Отношение взаимодостижимости рефлексивно, симметрично и транзитивно, т.е. является отношением эквивалентности. Множество X вершин графа разбивается на классы взаимодостижимых вершин $X_1, X_2, \ldots X_{\vec{\kappa}}$.

Определение. *Подграфы, порожденные этими подмножествами называются компонентами бисвязности, или бикомпонентами графа G.*

Число бикомпонент обозначается $\vec{\kappa}(G)$.

Множество вершин, достижимых из x, обозначим $D(x)$.

Теорема 7.2 *Вершины x и y взаимодостижимы тогда и только тогда, когда $D(x) = D(y)$.*

Доказательство. 1. $\Leftarrow$ *Достаточность.* Пусть $D(x) = D(y)$; требуется доказать взаимодостижимость x и y. Так как $x \in D(x)$ и $y \in D(y)$, то из равенства $D(x) = D(y)$ следует, что $y \in D(x)$ и $x \in D(y)$, т.е. вершины x и y взаимодостижимы.

2. ⇒ *Необходимость.* Если x и y взаимодостижимы, то для любой вершины $z \in X$, ввиду транзитивности отношения взаимодостижимости, из $z \in D(x)$ следует $z \in D(y)$, а из $z \in D(y)$ следует $z \in D(x)$, поэтому $D(x) = D(y)$. ⊠

7.3.1 Нахождение компонент и бикомпонент

Рассмотрим матрицу смежности R графа G над булевой алгеброй $\mathbf{B}\{0, 1\}$; обозначим через E единичную матрицу порядка $n(G)$ и образуем последовательность матриц

$$E + R, (E + R)^2, \ldots, (E + R)^l,$$

элементы которых отражают наличие маршрутов (ормаршрутов) длины не более l. Для некоторого l_0 (а именно для цепи (пути) наибольшей длины) выполняется условие

$$(E + R)^{l_0} = (E + R)^{l+1} = (E + R)^{l_0+2} = \ldots .$$

Каждой системе одинаковых строк (или столбцов) "установившейся" матрицы соответствует компонента (бикомпонента) связности.

Введем еще несколько терминов, которые понадобятся нам в дальнейшем.

> *Перешеек* (*точка сочленения* или *шарнир*) – это ребро (вершина), при удалении которого (которой) число компонент связности увеличивается на 1.
>
> Максимальный связный подграф, не имеющий своих точек сочленения, называется *блоком* графа.

Соответственно могут быть поставлены задачи нахождения перешейков, точек сочленения (шарниров), блоков.

7.4 Кратчайшие цепи

В этом разделе рассматривается несколько алгоритмов нахождения кратчайших цепей между заданной парой вершин, а также между всеми парами вершин графа.

7.4.1 Алгоритм нахождения кратчайших цепей между заданными вершинами

Пусть задан граф $G = (X, U; P)$. Требуется найти кратчайшие цепи, соединяющие вершины s и t.

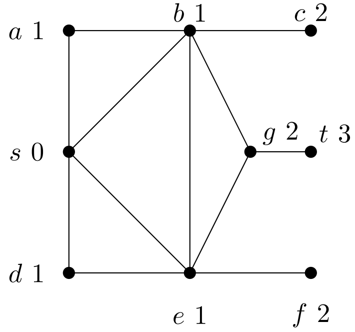
Описание алгоритма.

Шаг 1. Помечаем вершину s меткой 0.
Шаг 2. Меткой $k+1$ помечаем каждую вершину, которая еще не помечена и смежна хотя бы с одной вершиной, помеченной меткой k. Разметка прекращается, как только вершина t окажется помеченной некоторой меткой l.
Шаг 3. Сама кратчайшая цепь длины l отыскивается следующим образом. Начинаем с вершины t. За x_{l-1} берем любую вершину с меткой $l-1$ и смежную с t, а за u_l – любое ребро, соединяющее x_{l-1} с t. За x_{l-2} выбираем любую вершину, помеченную меткой $l-2$ и смежную с x_{l-1}, а за u_{l-1} – любое ребро, соединяющее x_{l-2} с x_{l-1} и так далее, пока не дойдем до вершины s. ⊠

Доказательство того, что цепь действительно кратчайшая, аналогично доказательству оптимальности алгоритма Дейкстры (смотри ниже).

Задача о нахождении кратчайших циклов, содержащих данное ребро u, сводится к предыдущей: достаточно удалить ребро u из G и в оставшемся суграфе найти кратчайшие цепи между вершинами x и y, которые в G были соединены ребром u.

Пример 7.6.
Найти кратчайшую цепь между вершинами s и t.



На рисунке буквами помечены вершины, цифрами – соответствующие метки вершин. Начиная с вершины t отыскиваем кратчайшую цепь (по вершинам) $[t\ g\ b\ s]$ или $[t\ g\ e\ s]$. В прямом направлении им соответствуют цепи $[s\ b\ g\ t]$ и $[s\ e\ g\ t]$; обе эти цепи – кратчайшие; длина кратчайшей цепи равна 3. ◀

7.4.2 Кратчайшая цепь между заданными вершинами (взвешенный граф)

Пусть задан граф $G = (X, U)$ и отображение $f : U \mapsto C$, ставящее в соответствие каждому ребру u, соединяющему пару вершин x_i, x_j, число c_{ij} – вес или длину ребра. Таким образом, задается *матрица весов* ребер $C = \|c_{ij}\|$, размерности $n \times n$.

Если последовательность вершин и ребер

$$su_1x_1u_2 \dots x_{l-1}u_lt$$

есть маршрут, соединяющий вершины s и t, то длина маршрута определяется как

$$c(s,t) = \sum_{k=1}^{l} c_{u_k},$$

где $c_{u_k} = c(x_i, x_j) = c_{ij}$; c_{ij} – вес ребра, соединяющего вершины x_i, x_j; рассматривается случай $c_{ij} \geq 0$.

З а м е ч а н и В. данной задаче имеет смысл рассматривать униграф, так как если пару вершин x_i, x_j соединяет более чем одно ребро, можно заранее выбрать из них ребро с наименьшим весом. ▶

Требуется найти кратчайший по длине маршрут (а мы показали, что это цепь), соединяющий вершины s и t.

Алгоритмы, приводимые ниже, используют разметку вершин. Составная метка при вершине x имеет вид $L(x) = [p, l(x)]$, где p – метка предшествования, а $l(x)$ – метка расстояния вершины x от s.

7.4.3 Алгоритм Форда

Шаг 0. Инициализация.
Назначим вершине s метку $L(s) = [s, 0]$. Для всех вершин x, отличных от s, $x \neq s$ назначим метки $L(x) = [x, \infty]$.
Шаг 1. Разметка вершин. Основной цикл.
Находим такое ребро (x, y), для которого

$$l(y) - l(x) > c(x, y)$$

и заменяем метку вершины y на $L'(y) = [x, l(x) + c(x, y)]$. При этом метка $l'(y) = l(x) + c(x, y)$ может только уменьшиться:

$$l'(y) < l(y) \text{ и } l'(y) > 0 \text{ при } y \neq s.$$

Этот процесс продолжаем до тех пор, пока имеется хотя бы одно ребро, для которого можно уменьшить метку.
Шаг 2. Нахождение кратчайшей цепи.
После того как метки установятся, то есть не найдется больше ни одной вершины y, для которой можно уменьшить метку $l(y)$, приступаем к построению кратчайшей цепи.

Начинаем с вершины t. Если ее метка есть $[x_k, l(t)]$, то x_k есть вершина в искомой цепи, предшествующая t; при этом $l(t) = l(x_k) + c(x_k, t)$.

Далее итеративно: если метка вершины x_k есть $[x_{k-1}, l(x_k)]$, то x_{k-1} есть вершина в искомой цепи, предшествующая x_k и

$$l(x_k) = l(x_{k-1}) + c(x_k, t)$$

В самом деле, метка $l(t)$ в процессе разметки уменьшалась, а x_k – последняя вершина, от которой уменьшалась метка $l(t)$. Точно так же, найдется вершина x_{k-1}, для которой

$l(x_k) = l(x_{k-1}) + c(x_{k-1}, x_k)$ и так далее.

Последовательность $l(t), l(x_k), l(x_{k-1}), \ldots$ строго убывающая, поэтому в некоторый момент будет $x_0 = s$.

Теорема 7.3 *$l(t)$ – длина кратчайшего маршрута из s в t и*

$$M = [s = x_0, \ldots, x_{k-1}, x_k, \ldots, t]$$

этот кратчайший маршрут.

Доказательство. Пусть в размеченном с помощью алгоритма графе имеется произвольный маршрут

$$s = x_0, x_{k_1}, x_{k_2}, \ldots, x_{k_n} = t$$

длины $c(s, t)$. Тогда

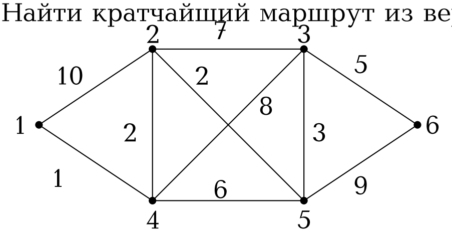
$$\begin{array}{lcll} l(x_{k_1}) & - & 0 & \leq & c(x_0, x_{k_1}), \\ l(x_{k_2}) & - & l(x_{k_1}) & \leq & c(x_{k_1}, x_{k_2}), \\ & \ldots & & \ldots & \ldots \\ l(x_{k_n}) & - & l(x_{k_{n-1}}) & \leq & c(x_{k_{n-1}}, x_{k_n}) \end{array}$$

Просуммировав левые и правые части неравенств, получим:

$$\begin{array}{lcl} l(x_{k_n}) - 0 & \leq & c(x_0, x_{k_n}) = c(s, t), \\ l(t) & \leq & c(s, t). \end{array}$$

Таким образом, длина произвольного маршрута не меньше длины маршрута, полученного с помощью алгоритма; следовательно, этот маршрут – кратчайший. ☒

Пример 7.7.
Найти кратчайший маршрут из вершины 1 в вершину 6.



Матрица весов (расстояний)

C =

C	1	2	3	4	5	6
1	0	10	∞	1	∞	∞
2	10	0	7	2	2	∞
3	∞	7	0	8	3	5
4	1	2	8	0	3	∞
5	∞	2	3	6	0	9
6	∞	∞	5	∞	9	0

Массив меток $L = [p, l(x)]$

	1	2	3	4	5	6
	1, 0	2, ∞	3, ∞	4, ∞	5, ∞	6, ∞
1	1, 0	1, 10	3, ∞	1, 1	5, ∞	6, ∞
2	1, 0	1, 10	2, 17	1, 1	2, 12	6, ∞
4	1, 0	4, 3	4, 9	1, 1	4, 7	6, ∞
5	1, 0	4, 3	4, 9	1, 1	4, 7	5, 16
2	1, 0	4, 3	4, 9	1, 1	2, 5	5, 16
5	1, 0	4, 3	5, 8	1, 1	2, 5	5, 16
3	1, 0	4, 3	5, 8	1, 1	2, 5	3, 13

Цифры в столбце слева от массива меток показывают начальную вершину x, цифра в строке на верху таблицы – конечную вершину y ребра (x, y), выбранного для разметки. Таким образом, длина кратчайшего маршрута равна 13, а сам маршрут, начиная с вершины 6, проходит по вершинам $[6, 3, 5, 2, 4, 1]$; или, начиная с вершины 1 – $[1, 4, 2, 5, 3, 6]$.

7.4.4 Алгоритм Дейкстры

Наиболее эффективный алгоритм решения задачи о кратчайшей $s-t$ цепи предложил Дейкстра. Рассматривается случай, когда все $c_{ij} \geq 0$. Алгоритм Дейкстры заключается в такой разметке вершин, что на каждой итерации временная метка в данной вершине дает верхнюю границу длины цепи от s до этой вершины. Эти метки постепенно уменьшаются с помощью итерационной процедуры и на каждом шаге итерации одна из временных меток становится постоянной. Тогда метка уже не является верхней границей, а дает точную длину кратчайшей цепи от s к вершине с постоянной меткой. Метка вершины x имеет вид $L(x) = [p, l(x)]$, где p – метка предшествования, $l(x)$ – метка расстояния вершины x от s.

Описание алгоритма.

Шаг 0. Инициализация.
Назначить вершине s метку $L(s) = [s, 0]$ и считать ее постоянной. Для всех $x \neq y$ назначить метки

$L(x) = [x, \infty]$ и считать эти метки временными.

$p := s$.

Шаг 1. Обновление меток. Начало итерации.
Для всех $x \in \Gamma p$ (смежных с р), метки у которых временные, изменить метки следующим образом: если $l(x) - l(p) > c(p, x)$, то назначить метку $[p, l(p) + c(p, x)]$, т.е.

$$l(x) = \min[l(x), l(p) + c(p, x)].$$

Шаг 2. Превращение метки в постоянную.
Среди всех вершин с временными метками найти такую $x^\dagger$, для которой

$$l(x^\dagger) = \min_x [l(x)]$$

Считать метку вершины $x^\dagger$ постоянной.

$p := x^\dagger$.

Шаг 3. Окончание разметки. Поиск цепи.
Рассматриваются два варианта.

1. Требуется найти только цепь от s к t.

Если $p = t$, то $l(t)$ – длина кратчайшей цепи от s к t; разметка закончена.

Если $p \neq t$, перейти на **шаг 1**.

2. Требуется найти кратчайшие цепи от s ко всем остальным вершинам.

Если все вершины помечены как постоянные, то эти метки дают длины кратчайших цепей; разметка закончена.

Если есть еще временные метки, перейти на **шаг 1**.

Шаг 4. Сама кратчайшая цепь строится точно так же, как и в алгоритме Форда. Начинаем с вершины t. Если ее метка есть $L(t) = [x, l(t)]$, то x – вершина искомой цепи, предшествующая t, причем $l(t) = l(x) + c(x,t)$. Далее, если метка вершины x есть $L(x) = [y, l(x)]$, то y есть вершина, предшествующая x, причем $l(x) = l(y) + c(y,x)$. Полагаем $x = y$ и продолжаем итерации до тех пор, пока не станет $x = s$. Найденная таким образом цепь и есть искомая.

Теорема 7.4 *Постоянная метка вершины x дает длину кратчайшей цепи из s в x.*

Доказательство. 1. Если все метки в графе постоянные, то доказательство то же, что и для алгоритма Форда.

2. Пусть S – множество вершин с постоянными метками, T – множество вершин с переменными метками.

Предположим, что на некоторой итерации разметки вершин постоянные метки дают длины кратчайших цепей (на первой итерации это очевидно). Нам нужно доказать, что кратчайшая цепь проходит только по вершинам из множества S.

Заметим, что на шаге 2 вершина $x^\dagger$ с минимальной для временных меткой включается в S. Предположим теперь, что на некоторой следующей итерации кратчайшая цепь из s в $x^\dagger$ не проходит полностью по вершинам из S и содержит по крайней мере одну вершину из T, и $z \in T$ – первая вершина в этой цепи, а вершина $y \in S$ – предшествует z в цепи. Тогда

$$c(s,z) = l(y) + c(y,z) \geq l(z).$$

Так как $l(z)$ – временная метка, а постоянная метка $l(x^\dagger)$ выбрана на этой итерации как наименьшая из временных, то $l(z) \geq l(x^\dagger)$; поскольку $c_{ij} \geq 0$, длина цепи между z и $x^\dagger$ должна быть неотрицательной, что противоречит этому неравенству. Таким образом мы доказали: кратчайшая цепь из вершины s в $x^\dagger$ проходит по вершинам с постоянными метками. По индукции доказательство продолжается до вершины t. ⊠

Пример 7.8.

Рассмотрим граф из предыдущего примера: найти кратчайшую цепь из вершины 1 в вершину 6. Процесс разметки будем рассматривать на массивах меток

$L(x) = [p(x), l(x)]$.

p	1	2	3	4	5	6	Примечания
$p=1$	$1,0$	$1,10$	$3,\infty$	$1,1$	$5,\infty$	$6,\infty$	$\min l(x) = 1; x^\dagger = 4$
$p=4$	$1,0$	$4,3$	$4,9$	$1,1$	$4,7$	$6,\infty$	$\min l(x) = 3; x^\dagger = 2$
$p=2$	$1,0$	$4,3$	$4,9$	$1,1$	$2,5$	$6,\infty$	$\min l(x) = 5; x^\dagger = 5$
$p=5$	$1,0$	$4,3$	$5,8$	$1,1$	$2,5$	$5,14$	$\min l(x) = 8; x^\dagger = 3$
$p=3$	$1,0$	$4,3$	$5,8$	$1,1$	$2,5$	$3,13$	$\min l(x) = 13; x^\dagger = 6$
$p=6$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	

Получили $p = x^\dagger = t$. Сама цепь находится точно так же, как и в алгоритме Форда.

7.4.5 Кратчайшие маршруты между всеми парами вершин. Алгоритм Флойда

Пусть задан взвешенный униграф (неориентированный, ориентированный, смешанный) $G = (X, U)$ с матрицей весов $C = \|c_{ij}\| = \|c(x_i, x_j)\|$, где $c_{ii} = 0, i = \overline{1, n}; c_{ij} = \infty$, если нет ребра (x_i, x_j). Требуется найти кратчайшие маршруты между всеми парами вершин.

Введем следующую операцию композиции матрицы весов $C \otimes C = \|c_{ij}^{(2)}\|$, где

$$c_{ij}^{(2)} = \min[c_{ij}, c_{ik} + c_{kj}]$$

при некотором фиксированном k. Алгоритм Флойда состоит в последовательном преобразовании матрицы весов. По завершении преобразований матрица весов представляет длины кратчайших цепей между всеми парами вершин. Для нахождения самих цепей определим матрицу цепей (или путей) $P = \|p_{ij}\|$. Под C^k и P^k будем понимать соответствующие матрицы, полученные на k-ой итерации.

Описание алгоритма.

Шаг 1. Инициализация.

$C^0 := C; P^0 := (p_{ij}^0)$, где $p_{ij}^0 = i; k := 1;$

Шаг 2. Для всех $i, j = 1, \ldots, n$ выполнить операцию

$c_{ij}^k := \min[c_{ij}^{k-1}, c_{ik}^{k-1} + c_{kj}^{k-1}]$;
Это значит:
$s := c_{ik}^{k-1} + c_{kj}^{k-1}$;
if $c_{ij}^{k-1} > s$ **then**
  begin
   $c_{ij}^k := s; p_{ij}^k := p_{kj}^{k-1}$;
  end
else
  begin
   $c_{ij}^k := c_{ij}^{k-1}; p_{ij}^k := p_{ij}^{k-1}$;
  end

Шаг 3. Проверка на наличие циклов отрицательной длины.

Если $c_{ll}^k < 0,\ 1 \le l \le n$, то алгоритм закончен: в графе имеется цикл отрицательной длины – решение невозможно.

Шаг 4. $k := k + 1$; Если $k < n$, то на **Шаг 2**.

Шаг 5. Получена матрица весов кратчайших цепей C^n и матрица самих цепей P^n.

Кратчайшая цепь между вершинами x_i, x_j определяется следующим образом:

$M(x_i, x_j) = [x_i, \ldots, x_{j_3}, x_{j_2}, x_{j_1}, x_j]$,
где $j_1 = P_{ij}^n; j_2 = P_{ij_1}^n; j_3 = P_{ij_2}^n; \ldots$, пока не получим $P_{ij_m}^n = i$.

Теорема 7.5 *Матрица C^n, полученная в результате работы алгоритма Флойда, является матрицей кратчайших расстояний между вершинами.*

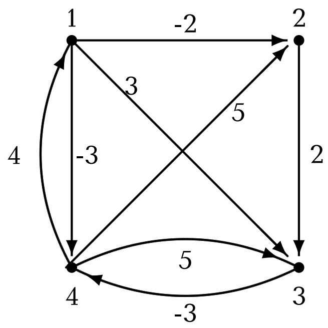
Доказательство. Пусть кратчайшая цепь из вершины x_i в вершину x_j проходит последовательно через вершины $x_1, x_2, \ldots, x_q$. Пусть k – первый шаг алгоритма, соответствующий какой-либо из этих вершин (пусть это будет вершина x_l; пусть также $x_i = x_0, x_j = x_{q+1}$). До начала итераций непосредственный переход от x_{l-1} к x_{l+1} не может быть короче, чем переход от x_{l-1} к x_{l+1} через x_l, так как в противном случае цепь

$x_0, \ldots, x_{l-1}, x_{l+1}, \ldots, x_{q+1}$
была бы короче кратчайшей. После изменения длины перехода от x_{l-1} до x_l (на шаге k) цепь

$x_0, \ldots, x_{l-1}, x_{l+1}, \ldots, x_{q+1}$

также становится кратчайшей. Повторяя этот процесс (переходя к новому k) установим, что в конце концов цепь из x_i в x_j становится кратчайшей. ☒

Пример 7.9.



$C^0 =$

C^0	1	2	3	4
1	0	-2	3	-3
2	∞	0	2	∞
3	∞	∞	0	-3
4	4	5	5	0

$P^0 =$

P^0	1	2	3	4
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4

$C^1 =$

C^1	1	2	3	4
1	0	-2	3	-3
2	∞	0	2	∞
3	∞	∞	0	-3
4	4	2	5	0

$P^1 =$

P^1	1	2	3	4
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	1	4	4

$C^2 =$

C^2	1	2	3	4
1	0	-2	0	-3
2	∞	0	2	∞
3	∞	∞	0	-3
4	4	2	4	0

$P^2 =$

P^2	1	2	3	4
1	1	1	2	1
2	2	2	2	2
3	3	3	3	3
4	4	1	2	4

$C^3 =$

C^3	1	2	3	4
1	0	-2	0	-3
2	∞	0	2	-1
3	∞	∞	0	-3
4	4	2	4	0

$P^3 =$

P^3	1	2	3	4
1	1	1	2	1
2	2	2	2	3
3	3	3	3	3
4	4	1	2	4

$C^4 =$

C^4	1	2	3	4
1	0	-2	0	-3
2	3	0	2	-1
3	1	-1	0	-3
4	4	2	4	0

$P^4 =$

P^4	1	2	3	4
1	1	1	2	1
2	4	2	2	3
3	4	1	3	3
4	4	1	2	4

Найдем, например, кратчайшую цепь из вершины x_2 в вершину x_1:

$$j_1 = P_{21} = 4; j_2 = P_{24} = 3; j_3 = P_{23} = 2; M = [2, 3, 4, 1].$$

7.5 Обходы графа

Существует ряд задач на графах, в которых требуется найти маршрут, который содержит все вершины или ребра графа – *обход*. Часто требуется, чтобы длина этого маршрута была минимальной (для взвешенных графов), или ограничивается число проходов по одному и тому же элементу графа. Одна из задач заключается в том, чтобы обойти все вершины графа и в каждой из них только один раз выполнить какое-либо действие: в простейшем случае пронумеровать или пометить вершину; в более сложных случаях решить какую-либо задачу, связанную с этой вершиной. Эту задачу часто называют *поиском* на графе.

7.5.1 Поиск в глубину на графе

В процессе поиска будем различать вершины

- непросмотренные;
- просмотренные (пронумерованные), но не помеченные;
- помеченные.

Перед началом просмотра все вершины считаются непросмотренными и непомеченными.

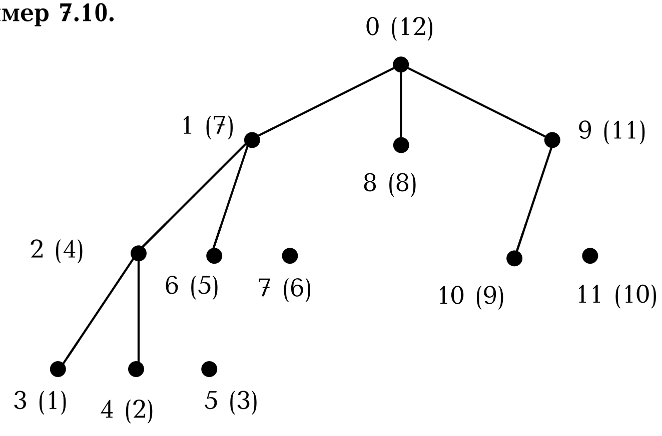
Общая идея метода состоит в следующем.

1. Поиск начинается с некоторой фиксированной вершины x_0, которой назначаем номер 0. Вершина x_0 теперь просмотрена, но не помечена. Затем выбираем любую вершину x_1, смежную с x_0, нумеруем ее единицей 1, и процесс поиска продолжается от нее. Считаем теперь вершину x_1, которая просмотрена, но не помечена, текущей.

2. Пусть x_i – текущая просмотренная вершина. Если существует еще непросмотренная вершина x_{i+1}, смежная с x_i, то поиск продолжается с вершины x_{i+1}. Если нет непросмотренной вершины, смежной с x_i, то вершина x_i считается помеченной. Теперь делается возврат – (*бэктрекинг, backtracking*) в вершину x_{i-1}, ту, из которой в процессе поиска мы попали в x_i. Вершина x_{i-1} становится текущей и поиск продолжается из нее.

Поиск продолжается до тех пор, пока снова не вернемся в вершину x_0 и x_0 окажется помеченной.

Пример 7.10.

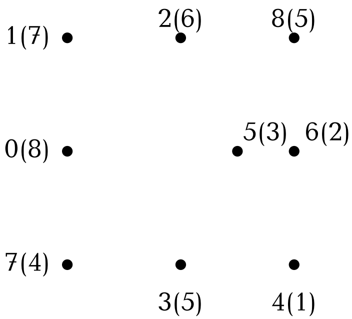


Поиск в глубину на дереве. Нумерация на рисунках соответствует очередности просмотра вершин в процессе поиска в глубину. Номера в скобках соответствуют очередности разметки. ◄

З а м е ч а н и е. Метод поиска в глубину очевидным образом переносится на орграфы – переход к следующей вершине происходит по направлению ориентации дуги, а возврат – в противоположном направлении. Таким образом, при обходе графа мы стремимся проникнуть в глубину графа так далеко, как это возможно, затем отступаем на шаг назад, снова стремимся пройти вперед и т.д. ►

Алгоритм поиска в глубину имеет сложность порядка $O(n+m)$, где n – число вершин, а m – число ребер графа. Действительно, каждая вершина просматривается один раз (всего n), а поиск смежных вершин (для всех n вершин) можно организовать за m просмотров.

Пример 7.11.
Поиск в глубину на произвольном графе. Нумерация вершин на рисунках соответствует очередности просмотра вершин в процессе поиска в глубину. Номера в скобках соответствуют очередности разметки.



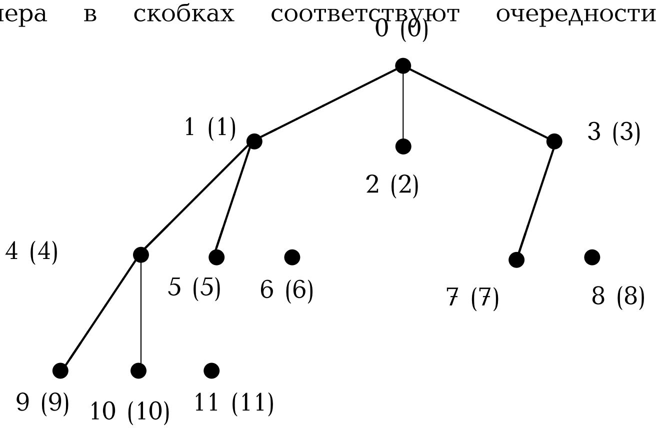
7.5.2 Поиск в ширину на графе

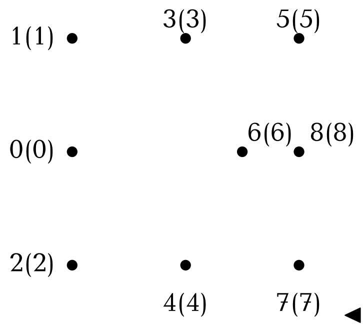
1. Поиск начинается с некоторой фиксированной вершины s.
2. Последовательно просматриваются и помечаются вершины, находящиеся на расстоянии 1 от s (т.е. смежные с s).
3. k-й шаг. Просматриваются и помечаются все вершины, находящиеся на расстоянии k от s.

Процесс поиска продолжается до тех пор, пока все вершины не будут просмотрены и помечены.

Пример 7.12.
Поиск в ширину на дереве и на произвольном графе. Нумерация вершин на рисунках соответствует очередности просмотра вершин в процессе поиска в ширину.

Номера в скобках соответствуют очередности разметки.





7.5.3 Эйлеровы цепи и циклы

Многие задачи теории графов были впервые сформулированы как головоломки, игры и т.д. К одной из этих задач относится и знаменитая задача о кенигсбергских мостах. Она формулируется следующим образом. На реке Преголя в Кенигсберге было два острова, которые соединялись с берегами реки и между собой семью мостами. Граф, соответствующий топографическому плану представлен на рисунке (вершины a, b – берега, c, d – острова, ребра – мосты).
Задача заключалась в том, чтобы начав движение с одного из участков суши, только по одному разу пройти по каждому мосту и вернуться в исходную точку, то есть найти цикл, проходящий через все мосты.

Эйлер обобщил эту задачу на произвольные графы и в 1736 году нашел ее решение.

Определение. *Эйлеровой цепью (циклом) называется цепь (цикл), содержащая (содержащий) все ребра графа. Если в графе существует эйлерова цепь (цикл), то граф называется эйлеровым.*

Если граф G несвязен, то он может иметь (но не обязательно имеет) эйлерову цепь лишь в том случае, если все его компоненты связности, кроме может быть одной, представляют собой голые вершины.

Теорема 7.6 *Эйлерова цепь (цикл) существует тогда и только тогда, когда число вершин с нечетной валентностью равно 2 (0).*

Доказательство. 1. $\Rightarrow$ *Необходимость.* Пусть эйлерова цепь существует и проходит из вершины x в вершину y. Если эта цепь – цикл ($x = y$), то в каждую вершину цепь должна зайти столько же раз, сколько и выйти (т.е. валентности всех вершин – четные). Если $x \neq y$, то из x цепь должна выйти на один раз больше, чем зайти; в y цепь должна зайти на один раз больше, чем выйти (т.е. валентности вершин x и y должны быть нечетными).

2. $\Leftarrow$ *Достаточность.* Пусть валентности всех вершин, за исключением может быть вершин x и y, четные.

Если все валентности четные: начнем цепь из некоторой произвольной вершины x и пойдем по некоторому еще не пройденному ребру к следующей вершине и так до тех пор, пока не вернемся в вершину x и не замкнем цикл. Если пройдены все ребра, то искомый эйлеров цикл C построен. Если в C входят не все ребра графа, то цикл C должен проходить через некоторую вершину z, инцидентную некоторому ребру, не вошедшему в C (т.к. граф связен).

Если удалить все ребра из C, то в оставшемся суграфе вершины будут по-прежнему иметь четные валентности, так как в суграфе C все валентности четные. Начиная теперь с вершины z построим цикл C', начинающийся и кончающийся в z. Если в C' пройдены все оставшиеся ребра, то процесс закончен. Нужный нам эйлеров цикл будет образован частью цикла C от x до z, затем циклом C' и, наконец, частью цикла C от z до x.

Если циклы C и C' содержат не все ребра графа, то строится следующий цикл, и так далее, до тех пор, пока не будет найден эйлеров цикл.

Если вершины x и y имеют нечетные валентности, то сначала находим цепь, соединяющую x и y; удалив ребра этой цепи, построим эйлеров цикл, начиная с вершины x. Эйлерова цепь выходит теперь из вершины x, проходит эйлеров цикл, а затем снова из вершины x по найденной цепи в вершину y. ⊠

На этом доказательстве основан алгоритм нахождения эйлеровой цепи.

Описание алгоритма.

Шаг 1. Находим простую цепь P, соединяющую вершины с нечетной валентностью (если они есть) x и y. Если ее длина больше 0 (т.е. $x \neq y$), то все ребра этой цепи пометим меткой 0.

Шаг 2. Если еще есть непомеченные ребра, то выберем среди них такое u, которое инцидентно хотя бы одной вершине цепи P. В суграфе, порожденном всеми непомеченными ребрами найдем простой цикл, содержащий ребро u (например, с помощью алгоритма нахождения цепей) и все ребра этого цикла пометим меткой 1.

Шаг 3. Если в графе остались непомеченные ребра, то из них выбирается такое v, которое смежно хотя бы с одним помеченным. В суграфе, порожденном непомеченными ребрами выявляем простой цикл, содержащий v, и все ребра этого цикла помечаем меткой $k+1$. Разметка продолжается до тех пор, пока не будут помечены все ребра графа.

Шаг 4. Маршрут строится следующим образом: за начальную вершину выбираем $x_0 = x$. Если уже построен маршрут

$$x_0 u_1 x_1 \ldots x_{k-1} u_k x_k, \ \ (k \geq 0),$$

котором все ребра различны, то

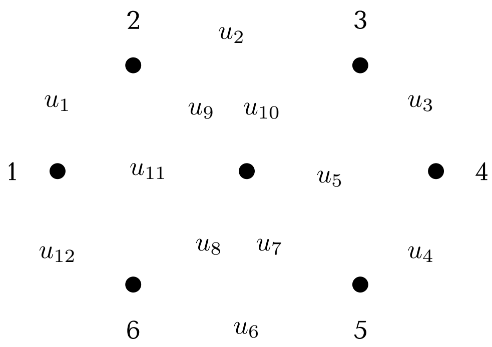
1) в случае $k = m$ процесс закончен (маршрут найден);

2) если $k < m$, то среди ребер, инцидентных x_k и отличных от $u_1, \ldots, u_k$ выбираем такое, которое помечено наибольшей меткой (или любое из них) и добавляем его к маршруту как ребро u_{k+1}, а за x_{k+1} берем ту вершину, с которой u_{k+1} соединяет вершину x_k (возможно $x_{k+1} = x_k$, если u_k – петля).

Сложность алгоритма нахождения эйлеровой цепи (цикла) имеет порядок $O(m)$.

З а м е ч а н и О. чевидно, что обобщение задачи нахождения эйлеровой цепи (если она существует) на взвешенные графы не имеет смысла, так как сумма весов в этой цепи всегда одна и та же. ►

Пример 7.13.



Все вершины имеют четные валентности. Разметка ребер может быть, например, такой:

u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}
1	2	3	3	3	4	4	4	2	2	1	1

Тогда эйлеров цикл (при вершине 1) имеет вид:

1 u_1 2 u_2 3 u_3 4 u_4 5 u_6 6 u_8 7 u_7 5 u_5 3 u_{10} 7 u_9 2 u_{11} 6 u_{12} 1. ◄

7.5.4 Эйлеровы пути

Введем следующие обозначения:

$v^+(x) = s^+(x) + s^\circ(x)$;

$v^-(x) = s^-(x) + s^\circ(x)$.

$v^+(x)$ и $v^-(x)$ – полувалентности вершины x.

Определение. *Путь в орграфе G, содержащий все его дуги, называется эйлеровым путем.*

Теорема 7.7 *Эйлеров путь из вершины s в вершину t существует тогда и только тогда, когда*

1) $x \in X \setminus \{s, t\}$ $[v^+(x) = v^-(x)]$.

2) $s \neq t : v^+(s) - v^-(s) = 1$;

$v^-(t) - v^+(t) = 1$.

Доказательство полностью аналогично предыдущему.

Для решения задачи нахождения эйлерова цикла вручную (например, для решения головоломок) можно использовать следующий алгоритм.

7.5.4.1 Алгоритм Флери

(все валентности – четные).

Маршрут начинается с любой вершины. Пройденное ребро удаляется. Не проходить по ребру, если его удаление приводит к увеличению числа компонент связности (не считая изолированных вершин).

Некоторые обобщения задачи нахождения эйлеровой цепи.

1. Задача китайского почтальона. В графе с неотрицательными весами ребер найти циклический маршрут наименьшей длины, проходящий через каждое ребро графа по крайней мере один раз.

Очевидно, если граф эйлеров, то эйлеров цикл и будет оптимальным.

2. Разбиение графа на эйлеровы суграфы. Если связный граф имеет $2k$ вершин с нечетной валентностью, то минимальное число покрывающих его реберно непересекающихся цепей равно k.

7.5.5 Гамильтоновы цепи и циклы

Определение. *Гамильтоновой цепью (циклом) графа называется простая цепь (простой цикл), содержащая (содержащий) все вершины графа.*

Пока неизвестен какой-либо достаточно простой критерий необходимости и достаточности существования гамильтоновой цепи (цикла) для произвольного графа G. Неизвестен также алгоритм нахождения гамильтоновой цепи (цикла) полиномиальной сложности.

Известен лишь ряд необходимых условий существования гамильтоновой цепи (цикла), из которых мы рассмотрим лишь два (без доказательства).

Теорема 7.8 (Оре) *Пусть в связном обыкновенном графе $G = (X, U)$ имеется наибольшая простая цепь*

$$Q = x_0\ u_1\ \ldots\ x_{l-1}\ u_l\ x_l,$$

такая, что $l \geq 2$ и $s(x_0) + s(x_l) \geq l + 1$.

Тогда в G существует гамильтонов цикл.

Теорема 7.9 *Если для двух несмежных различных вершин x и y связного обыкновенного графа G с $n \geq 3$*

$$s(x) + s(y) \geq n(G),$$

то G имеет гамильтонов цикл.

Если

$$s(x) + s(y) \geq n(G) - 1,$$

то существует гамильтонова цепь.

Следствие. *Если G – связный обыкновенный граф с $n(G) \geq 3$, такой, что для всех его вершин x*

$$s(x) \geq \frac{n(G)}{2},$$ *то в G существует гамильтонов цикл.*

Самый очевидный алгоритм нахождения гамильтоновой цепи состоит в генерировании $n!$ перестановок различных последовательностей вершин, и в проверке каждой из них на наличие простой цепи. Такой алгоритм требует по меньшей мере $n!n$ шагов и его сложность имеет порядок $n!n \approx O(n^n)$.

Продемонстрируем на примере методы поиска в ширину и в глубину для нахождения гамильтоновых цепей (циклов) на орграфе.

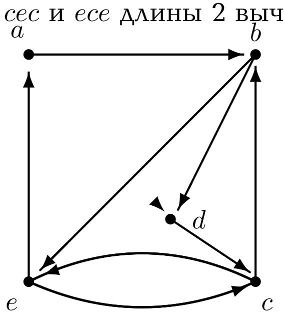
7.5.5.1 Алгоритм поиска гамильтоновой цепи (цикла) в ширину

Пример 7.14. Алгоритм поиска в ширину заключается в последовательном нахождении цепей длины $2, 3, \ldots, n$ и проверке на каждом шаге найденной цепи на допустимость (отсутствие цикла длины меньшей чем n).

Пусть граф G задан как бинарное отношение с помощью массива пар вершин. Цепи длины 2 , 3,... будем находить как степени отношений $R^2, R^3, \ldots$.

$G = (X, R), X = \{a, b, c, d, e\};$

$R = \{ab, ad, bd, be, cb, ce, dc, ea, ec\}$;
$R^2 = \{abd,\ abe,\ adc, bdc,\ bea,\ bec,\ cbd,\ cbe,\ cea, dcb,\ dce,\ eab,\ ead,\ ecb\}$
;
Циклы cec и ece длины 2 вычеркнуты, т.к. их длина меньше $n = 5$.



$R^3 = \{abdc,\ \underline{abea},\ abec,\ adcb, \underline{bdcb},\ bdce,\ \underline{beab}, bead,\ \underline{becb}, \underline{bece},\ \underline{cbdc},\ cbea,$ $\underline{cbec},\ ceab,\ cead,\ \underline{dcbd}, dcbe,\ dcea, \underline{dcec},\ eabd,\ \underline{eabe}, eadc,\ ecbd, \underline{ecbe}\}$.
Подчеркнутые маршруты не включаются в дальнейшее рассмотрение, т.к. они содержат циклы длины меньшей, чем n.
$R^4 = \{\underline{abdcd},\ abdce,\ \underline{abecb},\ \underline{abece}, \underline{adcbd},\ adcbe,\ \underline{adcea}, \underline{adcec}, bdcea, beadc,$ $\underline{cbeab}, cbead, \underline{ceabe}, dcbea, \underline{dcbec}, dceab, \underline{dcead}, eabdc, eadcb, \underline{eadce},\ \underline{ecbdc}\}$.

Неподчеркнутые цепи в R^4 представляют собой гамильтоновы цепи длины 4.

Для нахождения гамильтоновых циклов найдем еще одну степень отношения:

$R^5 = \{abdcea,\ adcbea,\ \underline{adcbec},\ bdceab,\ \underline{bdcead},\ beadcb, \underline{beadce},\ cbeadc,$ $ceabdc,\ \underline{dcbeab}, dcbead, dceabd, \underline{dceabe}, \underline{eadcbd}, eadcbe\}$.

Неподчеркнутые циклы в R^5 представляют собой гамильтоновы циклы. Некоторые из них представляют собой просто циклический сдвиг других циклов. В результате получим только два различных гамильтоновых цикла, например при вершине $a : \{abdcea,\ adcbea\}$. ◄

7.5.5.2 Алгоритм поиска гамильтоновой цепи (цикла) в глубину

Попытаемся найти гамильтоновы цепи или циклы, начиная с вершины a. Поиск в глубину в данном случае заключается в том, чтобы продвигаясь в глубину графа как можно дальше, находить цепи (пути) длины $n-1$ или простые циклы длины n, т.е. исключать в процессе поиска циклы длины меньше, чем n.

Дерево поиска для графа из предыдущего примера имеет следующий вид:

Вставить картинку

Вычеркнутые ребра в дереве поиска соответствуют циклам длины менее, чем n.
Найденные гамильтоновы циклы: $\{abdcea, adcbea\}$.

7.5.5.3 Связь между эйлеровыми и гамильтоновыми циклами

Построим *реберный* граф L ($linegraph$) данного графа G следующим образом:

- Множество вершин графа L соответствует множеству ребер графа G.
- Вершины графа L смежны, если смежны соответствующие ребра в графе G.

Верны следующие утверждения:

1. Если G имеет эйлеров цикл, то его реберный граф L имеет как эйлеров, так и гамильтонов цикл.
2. Если G имеет гамильтонов цикл, то его реберный граф L также имеет гамильтонов цикл.

Утверждения, обратные к 1 , 2 – неверны. Таким образом эти утверждения мало что дают для практики.

Задача нахождения гамильтоновой цепи (цикла) имеет обобщение на взвешенные графы.

Задача коммивояжера (*travelling seller problem, TSP*) формулируется следующим образом: во взвешенном графе (обычно – полном) найти гамильтонову цепь (цикл) наименьшей длины (с наименьшим суммарным весом ребер).

Название задачи происходит от следующей формулировки: имеется n городов и известны расстояния между каждой парой городов; коммивояжер (бродячий торговец), выходящий из какого-либо города, должен посетить $n-1$ других городов и вернуться в исходный. В каком порядке ему нужно посещать города, чтобы общее пройденное расстояние было минимальным ?

Глава 8

Цикломатика графов

8.1 Цикломатическое число

Важной числовой характеристикой графа $G = (X, U; P)$, не зависящей от ориентации и инвариантной относительно изоморфизма является *цикломатическое число* $\lambda(G)$, которое определяется следующим образом:

$\lambda(G) = m(G) - n(G) + \kappa(G)$,

где $n(G) = n = |X|$ – число вершин; $m(G) = m = |U|$ – число ребер; $\kappa(G) = \kappa$ – число компонент связности графа G.

Назовем ребро *цикловым*, если оно содержится хотя бы в одном цикле; в противном случае назовем его *перешейком*.

Лемма 3 *Пусть $G' = (X, U \setminus u; P)$ – суграф, полученный из G удалением ребра u. Тогда*

$$\kappa(G') = \begin{cases} \kappa(G), & \textit{если } u \textit{ – цикловое ребро,} \\ \kappa(G) + 1, & \textit{если } u \textit{ – перешеек.} \end{cases}$$

Д о к а з а т е л ь с т в о . 1. Пусть ребро u соединяет вершины x, y и входит в некоторый цикл

$C \ = \ x_0 \ u_1 \ x_1 \ \ldots \ x \ u \ y \ \ldots \ x_{l-1} \ u_l \ x_0$

Если удалить ребро u, то можно построить цепь

$x \ \ldots \ x_1 \ u_1 \ x_0 \ u_l \ x_{l-1} \ \ldots \ y$,

соединяющую те же вершины x и y и не содержащую u. Следовательно, удаление u не увеличило числа компонент, и граф G' связен, то есть $\kappa(G') \ = \ \kappa(G)$.

2. Пусть ребро u не входит ни в какой цикл графа G. Тогда очевидно, при удалении ребра u из любой цепи, соединяющей вершины x, y, в графе G не найдется ни одной цепи, соединяющей эти же вершины и $\kappa(G') = \kappa(G) + 1$. ☒

Так как $n(G') = n(G)$, $m(G') = m(G) - 1$, получим

$$\lambda(G') = \begin{cases} \lambda(G), \text{ если } u \text{ – перешеек}, \\ \lambda(G) + 1, \text{ если } u \text{ – цикловое ребро}. \end{cases}$$

Теорема 8.1

1. $\lambda(G) \geq 0$.

2. $\lambda(G) = 0$ *тогда и только тогда, когда граф* G *не содержит циклов.*

Доказательство. 1. Пусть граф содержит циклы, по крайней мере одно цикловое ребро. Удалив это ребро, мы уменьшим λ на единицу. Если в оставшемся суграфе еще есть циклы, то снова удалим цикловое ребро и т.д. При этом κ остается без изменения. В конце концов придем к графу без циклов $G^\star$, у которого $\lambda(G^\star) < \lambda(G)$. Отсюда $\lambda(G) > 0$.

Пусть теперь в графе нет циклов. Тогда последовательное удаление ребер, не изменяя λ, превратит граф в пустой (безреберный) E_n, для которого $m = 0$, и $n = \kappa$, откуда

$$\lambda = m - n + \kappa = 0 - n + n = 0.$$

Таким образом, $\lambda(G) \geq 0$.

2. В одну сторону равенство уже доказано (если в графе нет циклов, то $\lambda(G) = 0$).

Пусть теперь $\lambda(G) = 0$. Покажем, что G не содержит циклов ($\lambda(G) = 0$ – неотрицательно).

При удалении ребра λ может только уменьшиться, но не более чем до нуля:

$0 = m - n + \kappa$ или $\kappa = n - m$.

При удалении любого ребра будет

$\lambda' = 0, m' = m - 1, n' = n, \kappa' = \kappa + 1,$
то есть это любое ребро – перешеек, что и означает, что в графе нет циклов. ☒

Следствие. *Если* $\lambda(G) > 0$, *то всякий суграф, полученный из* G *удалением менее* λ *ребер, содержит циклы.*

Теорема 8.2 *Если* $G_1, G_2, \ldots, G_\kappa$ *– компоненты связности графа* G, *то*

$$\lambda(G) = \sum_{i=1}^{\kappa} \lambda(G_i).$$

Доказательство.

$$\lambda(G) = m(G) - n(G) + \kappa(G) = \sum_{i=1}^{\kappa} m(G_i) - \sum_{i=1}^{\kappa} n(G_i) + \kappa =$$

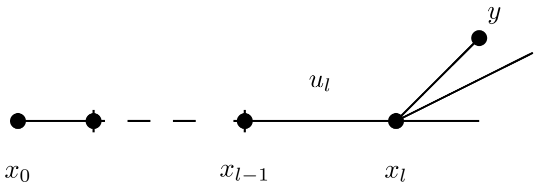
$$= \sum_{i=1}^{\kappa} [m(G_i) - n(G_i) + 1] = \sum_{i=1}^{\kappa} \lambda(G_i). \boxtimes$$

Теорема 8.3 *Граф* G *с* $n(G) \geq 2$ *содержит по крайней мере две вершины, не являющиеся точками сочленения (шарнирами).*

Доказательство. Считаем, что граф связен и $m(G) \geq 1$. Выберем в G простую цепь

$$Q = x_0\ u_1\ x_1 \ldots x_{l-1}\ u_l\ x_l$$

наибольшей длины l (очевидно, $l \geq 1$). Концевые вершины x_0 и x_l этой цепи и являются искомыми: действительно, если бы граф $G \backslash x_l$ (т.е.$G' = (X \setminus \{x_l\}, U';\ P)$) обладал более чем одной компонентой, то вершина x_l в G была бы смежна с некоторой вершиной y, не принадлежащей той компоненте графа $G \backslash x_l$, которая содержит x_0.



Но тогда цепь

$$x_0\ u_1\ x_1\ \ldots\ x_{l-1}\ u_l\ x_l\ (x_l,\ y)\ y$$

в графе G была бы простой и имела длину $l+1$. Это противоречит предположению о том, что Q – цепь наибольшей длины. $\boxtimes$

8.2 Деревья

Связный граф без циклов называется *деревом*.
Граф, все компонеты связности которого – деревья, называется *лесом*.
Висячие вершины дерева называются *листьями*.

Будем обозначать дерево через $T = (X, U)$. Неориентированное дерево есть обыкновенный граф, поэтому в дереве T $\forall x \in X[v(x) = s(x)]$ (в дереве нет петель).

8.2.1 Свойства дерева

1. $\kappa(T) = 1$ & $\lambda(T) = 0$ (определение дерева).
2. $\lambda(T) = 0$ & $n - m = 1$ (из определения).
3. Каждое ребро в T – перешеек (т.е. при удалении любого ребра κ увеличивается на единицу).
4. Для любых двух вершин x, y дерева T существует одна и только одна соединяющая их цепь из x в y, и эта цепь необходимо простая.
5. Соединение любых двух вершин x, y дерева T новым ребром приводит к появлению цикла.
6. Вершина x дерева T является точкой сочленения (шарниром) тогда и только тогда, когда ее степень $s(x) > 1$.
7. Если $n(T) \geq 2$, то в T есть по крайней мере две висячие вершины.

Доказательство.
Свойство 2 непосредственно следует из определения дерева.
Свойство 3 выражает тот факт, что в дереве нет циклов, т.е. следует из определения.
Свойство 4. (Доказательство от противного.) Предположим, что существуют две различные цепи

$$x_0\ u_1\ x_1\ u_2\ x_2\ \ldots\ x_{l-1}\ u_l\ x_l$$

и

$$x_0\ v_1\ y_1\ v_2\ y_2\ \ldots\ y_{k-1}\ v_k\ x_l,$$

соединяющие вершину $x = x_0$ с вершиной $y = x_l$, причем $l \geq k$. При этом первая из этих цепей имеет ненулевую длину. Тогда ребро u_i, где

$$i = \min\{j|\ u_j \neq v_j\}$$

(если $l > k$ и $u_j = v_j$ при $j = 1, 2, \ldots, k$, то положим $i = k + 1$), является цикловым, так как после его удаления из G вершины x_{i-1} и x_i останутся соединенными маршрутом

$$x_{i-1}\ v_i\ y_i\ \ldots\ y_{k-1}\ v_k\ x_l\ u_l\ x_{l-1}\ \ldots\ x_{i+1}\ u_{i+1}\ x_i.$$

Следовательно, граф имеет циклы, что противоречит определению дерева.
Свойство 5 следует из определения дерева; действительно, добавление ребра (разумеется, без добавления вершины) приводит к увеличению λ на единицу, то есть к появлению цикла.
Свойство 6 можно проверить непосредственно, учитывая, что в дереве нет циклов.
Свойство 7 следует из свойства 6 и теоремы 2.3 (вершины, не являющиеся точками сочленения, есть висячие вершины). ⊠

Свойство 4 можно представить в следующей форме:

Теорема 8.4 *Конечный неориентированный граф является деревом тогда и только тогда, когда для любых его вершин x, y существует цепь из x в y, притом простая и единственная.*

При $x = y$ эта цепь имеет нулевую длину.
Следствие. *Из любого графа G можно удалить $\lambda(G)$ ребер так, чтобы полученный суграф T не имел циклов и обладал тем же числом компонент связности, что и G. Всякий же суграф, полученный из G удалением менее чем $\lambda(G)$ ребер имеет циклы.*

8.3 Каркасы

Изучение графов без циклов позволит нам выяснить смысл цикломатического числа в тех случаях, когда оно отлично от нуля.

Определение. *Всякий суграф T графа G, у которого*

$$m(T) = m(G) - \lambda(G),$$

$$\kappa(T) = \kappa(G),$$

$$\lambda(T) = 0,$$

называется каркасом графа G

(синонимы: óстов, стягивающее дерево, *spanning tree*, *ST*).

В определении каркаса любые два условия влекут третье, в силу равенства $n(T) = n(G)$ и определению цикломатического числа.

Существование хотя бы одного каркаса у каждого графа следует из свойств дерева и следствий из них. Справедлива и более сильная теорема о существовании каркаса.

Теорема 8.5 *(теорема Коцига). Пусть T' – произвольный связный суграф графа G, не имеющий циклов. Тогда у графа G есть по крайней мере один каркас T, содержащий все ребра T' (т.е. T' является суграфом T).*

Доказательство. Если данный граф G не содержит циклов, то он сам и есть искомый каркас, то есть $T = G$, так как при $\lambda(G) = 0$ граф G, по определению, каркас. Если же $\lambda(G) > 0$, то в G найдется цикловое ребро, не принадлежащее суграфу T' (иначе T' имел бы циклы). Удаление этого ребра из G уменьшит $\lambda(G)$ на единицу; при этом $\kappa(G)$ не изменится. Если в полученном суграфе еще есть циклы, то снова удалим цикловое ребро, не принадлежащее T' и т.д. После $\lambda(G)$ таких шагов получим искомый каркас T. ⊠

8.3.1 Алгоритм нахождения каркаса графа.

В этом алгоритме мы воспользуемся той же разметкой, которую мы применяли для нахождения кратчайшей цепи в связном графе.

Шаг 1. Выберем произвольную вершину и пометим ее меткой 0.

Шаг 2. Все непомеченные вершины, смежные с вершинами, имеющими метку k, помечаем меткой $k + 1$. Разметка продолжается до тех пор, пока все вершины не будут помечены. При такой разметке метки смежных вершин не могут отличаться более чем на единицу.

Шаг 3. После окончания разметки будем просматривать вершины в любом порядке и удалять некоторые ребра по следующим правилам:

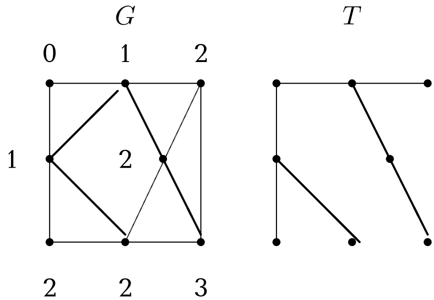
- если в данный момент мы находимся в вершине x с меткой $l(x)$, то удаляем все те еще не удаленные ранее ребра, которые соединяют x с вершинами, имеющими ту же метку;
- из ребер, соединяющих x с вершинами, имеющими метку $l(x) - 1$, удаляем все, кроме любого одного.

Покажем, что суграф T, оставшийся после такого удаления ребер, является каркасом.

1. $\kappa(T) = \kappa(G)$, так как правила удаления таковы, что разметка сохраняется и для суграфа T. Поэтому число вершин, соединимых с вершиной, имеющей метку 0, одинаково в T и в G.

2. Разметка такова, что ребро между двумя вершинами с одинаковыми метками – цикловое. Далее, мы отыскиваем для каждой вершины единственную цепь, соединяющую эту вершину с вершиной, имеющей метку 0, отсекая все другие возможные цепи.

Пример 8.1.



Граф G и один из его возможных каркасов T (на графе G указана разметка для нахождения каркаса). ◀

8.3.2 Кратчайший каркас графа.

Рассмотрим связный обыкновенный граф со взвешенными ребрами $G = (X,\ U)$; вес ребра $(x_i,\ x_j)$ обозначим через c_{ij} или $c(x_i,\ x_j)$. Если нет ребра $(x_i,\ x_j)$, то $c(x_i,\ x_j) = \infty$.

Задача заключается в том, чтобы из всех каркасов графа найти один такой, что сумма весов его ребер – наименьшая. Назовем такой каркас *кратчайшим* (*shortest spannig tree –SST*). Задача нахождения SST возникает, например, в том случае, если какие-либо пункты (города, дома в городе, клеммы электрической сети и т.д.) нужно связать кратчайшей сетью коммуникаций (трубопроводов, электрических проводов, дорог и т.д.). Задача нахождения кратчайшего каркаса (SST) – это одна из немногих задач теории графов, которую можно считать полностью решенной.

Подход к построению алгоритма нахождения SST состоит в следующем. Пусть в процессе построения SST уже построены его поддеревья

$$T_1, T_2,\ \ldots, T_k$$

с соответствующими подмножествами вершин

$$X_1, X_2,\ \ldots, X_k.$$

Для двух поддеревьев $T_i,\ T_j$ рассмотрим все соединяющие их ребра, если они имеются, и выберем кратчайшее из них (с наименьшим весом), т.е. ребро с весом

$$\Delta_{ij} = \min_{x_i \in X_i} [\min c(x_i,\ x_j)],\ i \neq j.$$

Покажем, что это ребро принадлежит SST, и деревья T_i и T_j можно срастить в одно поддерево с включением этого ребра.

Доказательство (от противного). Пусть на некоторой итерации алгоритма, например, на k-й, построены поддеревья, принадлежащие окончательному SST, а ребро $(x_i^\star,\ x_j^\star)$ – кратчайшее между деревьями $T_i,\ T_j$, в SST не содержится. Так как в конце концов деревья T_i и T_j должны быть связаны, то в SST должно существовать некоторое ребро $(x_i,\ x_j)$, такое, что, $x_i\ \in X_i,\ x_j\ \in\ X_j$. Если теперь мы удалим ребро $(x_i,\ x_j)$ и добавим ребро $(x_i^\star\ ,\ x_j^\star)$, то получим новое дерево, более короткое, чем SST, что противоречит определению SST.

Таким образом, ребро $(x_i^\star,\ x_j^\star)$ можно добавить к частично сформированному на k-й итерации SST. Заметим, что результат не зависит от выбора поддеревьев $T_i,\ T_j$. Поскольку на начальном этапе (пока еще никакие ребра не выбраны) предположение о принадлежности ребер к SST автоматически выполнено, то многократно выбирая кратчайшие ребра, в конце концов получим SST. ⊠

8.3.3 Алгоритм Прима.

Алгоритм строит кратчайший каркас, который начинается с одной вершины и разрастается при добавлении на каждой итерации по одному ребру. Вершины при этом рассматриваются как отдельные поддеревья кратчайшего каркаса. Поддерево T постепенно разрастается за счет добавления ребер (x, y), где $x \in T$, а $y \notin T$; при этом добавляемое к каркасу ребро должно иметь наименьший вес $c(x, y)$. Процесс продолжается до тех пор, пока число ребер в T не станет равным $n - 1$. Тогда дерево T и будет искомым SST.

В алгоритме применяется разметка вершин, очень похожая на ту, которая используется в алгоритме Дейкстры нахождения кратчайшей цепи.

Обозначения :

Y – текущее множество вершин, входящих в SST;

V – текущее множество ребер, входящих в SST;

Составная метка вершины x имеет вид $[\ p(x), l(x)]$; здесь $p(x)$ – метка предшествования: вершина, предшествующая x и определяющая ребро $(p(x),\ x)$, а $l(x)$ – метка, соответствующая весу этого ребра.

Представим алгоритм в следующем виде.

Шаг 1. Инициализация.

$Y := \{y\}; (y$ – произвольно выбранная вершина)

$p(y) := y; l(y) := 0$;

$V\ =\ \emptyset$;

Шаг 2. Для каждой вершины $x \notin Y$ найти вершину $q \in Y$, такую, что

$$c(q,\ x) = \min_{z \in Y}[c(z,\ x)];$$

$p(x) := q;\ l(x) := c(q,\ x)$;

Если такой вершины нет, т.е. x несмежна с вершинами из Y, то

$p(x) := x;\ l(x) := \infty$;

Под ∞ здесь понимается любое число, заведомо большее, чем вес любого из ребер графа G.

Шаг 3. Выбрать такую вершину z, что

$$l(z)\ =\ \min_{z \notin Y}\ [\ l(x)]$$

(Вершина z – ближайшая к уже построенному поддереву SST).

(Обновить данные).

$Y := Y \cup \{z\}; V := V \cup \{(p(z), z)\}$;

Если $|Y| = n$, то останов; (Ребра из V образуют SST).

Если $|Y| < n$, то на **Шаг 4**.

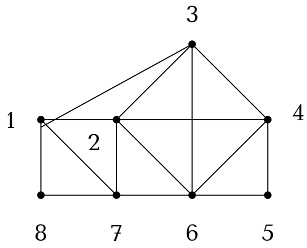
Шаг 4. Для всех $x \notin Y$, таких, что $x \in \Gamma z$ (смежных с вершиной z) обновить метки следующим образом :

если $l(x) > c(z, x)$, то $l(x) := c(z, x); p(x) := z$;

на Шаг 3.

Пример 8.2.

Пусть задан граф



$$C = \begin{array}{c|cccccccc} C & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \hline 1 & 0 & 3 & 4 & \infty & \infty & \infty & 7 & 2 \\ 2 & 3 & 0 & 5 & 6 & \infty & 8 & 8 & \infty \\ 3 & 4 & 5 & 0 & 9 & \infty & 2 & \infty & \infty \\ 4 & \infty & 6 & 9 & 0 & 9 & 3 & \infty & \infty \\ 5 & \infty & \infty & \infty & 9 & 0 & 3 & \infty & \infty \\ 6 & \infty & 8 & 2 & 3 & 3 & 0 & 9 & \infty \\ 7 & 7 & 8 & \infty & \infty & \infty & 9 & 0 & 5 \\ 8 & 2 & \infty & \infty & \infty & \infty & \infty & 5 & 0 \end{array}$$

с матрицей весов C.

Требуется найти кратчайший каркас. Действуем в соответствии с предписаниями алгоритма.

	$\star^1$	$\star^3$	$\star^4$	$\star^6$	$\star^7$	$\star^5$		$\star^2$		
Y	1	2	3	4	5	6	7	8	V	C
1	$[1,0]$	$[1,3]$	$[1,4]$	$[4,\infty]$	$[5,\infty]$	$[6,\infty]$	$[1,7]$	$[1,2]$	$(1,8)$	2
8	$\times$	$[1,3]$	$[1,4]$	$[4,\infty]$	$[5,\infty]$	$[6,\infty]$	$[8,5]$	$\times$	$(1,2)$	3
2	$\times$	$\times$	$[1,4]$	$[2,6]$	$[5,\infty]$	$[2,8]$	$[8,5]$	$\times$	$(1,3)$	4
3	$\times$	$\times$	$\times$	$[2,6]$	$[5,\infty]$	$[3,2]$	$[8,5]$	$\times$	$(3,6)$	2
6	$\times$	$\times$	$\times$	$[6,3]$	$[6,3]$	$\times$	$[8,5]$	$\times$	$(6,4)$	3
4	$\times$	$\times$	$\times$	$\times$	$[6,3]$	$\times$	$[8,5]$	$\times$	$(6,5)$	3
5	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$[8,5]$	$\times$	$(8,7)$	5

Примечания.

Шаг 1. $Y := \{1\}$. Выбираем произвольную вершину 1.

$p(1) = 1; l(1) = 1.$

Отмечаем вершину 1 как вошедшую в каркас с номером 1 знаком $\star^1$.

$V = \varnothing$; Суммарный вес каркаса $C = \infty$.

Шаг 2. Помечаем вершины 2, 3, 7, 8.

$p(2) := 1; l(2) := 3; p(3) := 1; l(3) := 4; p(7) := 1; l(7) := 7;$

$p(8) := 1; l(8) := 2;$

Остальные метки: $p(x) := x; l(x) := \infty.$

Шаг 3. Наименьшая метка $l(x)$ – у вершины 8 ;

$Y := \{1\} \cup \{8\}; V := \{(1,8)\}$; Отмечаем вершины 1 и 8, как включенные в Y. Длина ребра (1,8) равна 2. Текущей вершиной z становится вершина 8. Отмечаем вершину 8 (как новый элемент Y) знаком $\star^2$.

Шаг 4. Для всех вершин, смежных с вершиной 8, обновляем метки:

$p(7) := 8; l(7) := 5$. Остальные метки остаются прежними. На **Шаг 3**.

Шаг 3. Наименьшая метка $l(x)$ – у вершины 2;

$Y := \{1, 8, 2\}; V := \{(1, 8), (1, 2)\}; c := 3; z := 2$.

Отмечаем вершину 2 (как элемент Y) знаком $\star^3$.

Шаг 4. Обновляем метки: $p(4) := 2; l(4) := 6; p(6) := 2; l(6) := 8$.

Остальные метки остаются прежними.

Шаг 3. Наименьшая метка $l(x)$ – у вершины 3.

$Y = \{1, 8, 2, 3\}; V = \{(1, 8), (1, 2), (1, 3)\}; c := 4; z := 3$.

Отмечаем вершину 3 (как элемент Y) знаком $\star^4$.

Шаг 4. Обновляем метки: $p(6) := 3; l(6) := 2$. Остальные метки остались прежними.

Шаг 3. Наименьшая метка $l(x)$ – у вершины 6.

$Y := \{1, 8, 2, 3, 6\}; V := \{(1, 8), (1, 2), (1, 3), (3, 6)\}; c := 2; z := 6$.

Отмечаем вершину 6 знаком $\star^5$.

Шаг 4. Обновляем метки: $p(4) := 6; l(4) := 3; p(5) := 6; l(5) := 3$.

Остальные метки без изменений.

Шаг 3.Наименьшая метка $l(x)$ – у вершины 4.

$Y := \{1, 8, 2, 3, 6, 4\}; V := \{(1, 8), (1, 2), (1, 3), (3, 6), (6, 4)\}; c := 3; z := 4$. Отмечаем вершину 4 знаком $\star^6$.

Шаг 4. Все метки остаются без изменений.

Шаг 3. Наименьшая метка $l(x)$ – у вершины 5;

$Y := \{1, 8, 2, 3, 6, 4, 5\}; V := \{(1, 8), (1, 2), (1, 3), (3, 6), (6, 4), (6, 5)\};$
$c := 3; z := 5$.

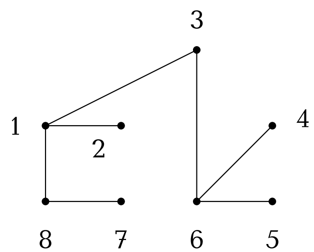
Отмечаем вершину 5 знаком $\star^7$.

Шаг 4. Все метки остаются без изменений.

Шаг 3. Наименьшая метка $l(x)$– у вершины 7.

$Y := \{1, 8, 2, 3, 6, 4, 5\}; V := \{(1, 8), (1, 2), (3, 6), (6, 4), (6, 5), (8, 7)\};$
$c := 5; |Y| = 8 = n$.

Суммарный вес : 22. Кратчайший каркас приведен на рисунке.



◄

8.3.4 Теорема о хорде каркаса.

Ребра графа G, не принадлежащие его каркасу T, называются *хордами* каркаса T в G.

Число ребер каждого из каркасов графа G называется *рангом* графа G и обозначается $\rho(G) = m - \lambda = n - \kappa$.

Если граф связен, то каждый его каркас – дерево.

Теорема 8.6 (о хорде каркаса) *Пусть T – некоторый произвольный каркас графа G, u – произвольная хорда этого каркаса.*

В графе G существует цикл, содержащий u и не содержащий других хорд каркаса T, причем этот цикл простой и единственный.

Доказательство. Доказательство проводится для связного графа, т.к. для несвязного графа можно провести доказательство для каждой из компонент.

Пусть x, y – те вершины графа G, которые соединяет ребро u. Так как T – дерево, то в нем имеется единственная цепь, притом простая, соединяющая x с y (свойство дерева) и эта цепь вместе с хордой u образует тот самый единственный простой цикл. ⊠

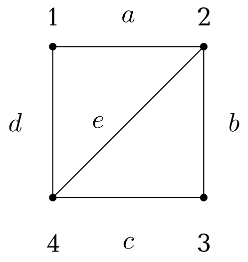
Эта теорема позволяет получать одни каркасы из других. Пусть T – некоторый каркас графа G, а u – какая-либо хорда этого каркаса (u – непетля: если непетель нет, то каркас T – единственный и задача теряет смысл).

По теореме 2.6 ребро u образует в G простой цикл вместе с некоторыми ребрами каркаса. Добавим ребро u к каркасу и удалим из образованного цикла какое-либо другое ребро (ранее принадлежавшее каркасу T; в результате получится новый каркас графа G, отличный от T. С помощью такой операции замены ребер каркаса можно из любого каркаса получить любой другой каркас того же графа (и с тем же ρ).

Пусть T и T' – два разных каркаса графа G. Добавим к T некоторое ребро u' каркаса T', не принадлежащее T (такое ребро всегда найдется – число ребер в T и T' одно и то же и они разные). В образовавшемся простом цикле найдется ребро v, не принадлежащее T' (иначе T' содержал бы цикл). После добавления u' и удаления v каркас T преобразуется в другой каркас T'', у которого число ребер, не принадлежащих T', меньше чем у T.

Если $T'' \neq T$, то повторяем подобную операцию с T'', и так далее, до тех пор, пока не получим T'.

Пример 8.3.



$T = \{d, c, e\}, T' = \{a, b, c\}$;

$T'' := T \cup \{a\} = \{a, d, c, e\}$;

$T'' := T'' \setminus \{e\} = \{a, d, c\}$;

$T'' := T'' \cup \{b\} = \{a, b, c, d\}$;

$T' := T'' \setminus \{d\} = \{a, b, c\}$.

8.3.5 Число каркасов графа.

Пусть задан связный граф $G = (X, U; P)$, где
$X = \{x_1, \ \dots \ , x_n\}$.
Образуем квадратную матрицу

$$S(G) = \left\| \begin{array}{cccc} 2\ s(x_1) - v(x_1) & -s(x_1, x_2) & \cdots & -s(x_1, x_n) \\ -s(x_2, x_1) & 2s(x_2) - v(x_2) & \cdots & -s(x_2, x_n) \\ \cdots & \cdots & \cdots & \cdots \\ -s(x_n, x_1) & -s(x_n, x_2) & \cdots & 2s(x_n) - v(x_n) \end{array} \right\|$$

Пусть M – главный минор матрицы S, полученный при вычеркивании любых i-й строки и i-го столбца. Число различных каркасов графа равно M. (Без доказательства).

Для полного обыкновенного графа число различных каркасов равно n^{n-2}.

8.3.6 Разрезы

Определение. *Разрезом графа $G = (X, U; P)$ называется такое подмножество $U' \subseteq U$ ребер, что суграф, полученный из G удалением U', имеет больше компонент связности, чем G,*

т.е. $\kappa(X,\ U \setminus U';\ P)\ >\ \kappa(X,\ U;\ P)$.
Разрезом также называется и суграф $G' = (X, U'; P)$.

Разрез U' – *простой*, если никакое его собственное подмножество $U'' \subset U'$ не является разрезом графа G.

Теорема 8.7 (о разрезах) *Каковы бы ни были каркас T графа G и ребро u этого каркаса, существует такой единственный простой* *разрез U' графа G, который содержит u, и не содержит других ребер каркаса T.*

Доказательство (для связного графа). Пусть $T_1 = (X_1, V_1)$ и $T_2 = (X_2, V_2)$ те два дерева, на которые распадается каркас $T = (X, V)$ после удаления ребра u, а U' – множество ребер графа G, соединявших X_1 и X_2 (вместе с u) в исходном графе G.

Множество U' (содержащее ребро u) – простой разрез. Но все другие разрезы, составленные из ребра u и каких-либо хорд каркаса T, должны содержать множество U', так как удаление из G каких-либо хорд каркаса не разбивает G на компоненты связности, а удаление хорд вместе с ребром u – разбивает, причем только в том случае, если вместе с u будут удалены и остальные ребра из U' (т.е. соединяющие X_1 и X_2).

Следовательно, это единственный разрез, содержащий u, и не содержащий других ребер каркаса. ⊠

Теорема 8.8 (о циклах и разрезах) *Любой цикл и любой простой разрез имеют четное число (возможно, нуль) общих ребер.*

Доказательство. Пусть U' – простой разрез, а $G_1 = (X_1, U_1;\ P)$, $G_2 = (X_2, U_2;\ P)$ – компоненты, получившиеся после удаления U'. Пусть, далее, C – произвольный цикл в G.

Так как U' – простой разрез, то пара соседних вершин цикла C соединена ребром из U' тогда и только тогда, когда одна из вершин принадлежит X_1, а вторая – X_2. Но таких пар вершин должно быть четное число, так как при полном обходе цикла мы попадем из X_1 в X_2 столько же раз, сколько из X_2 в X_1.

Это число равно 0, когда хотя бы одно из множеств X_1, X_2 не содержит вершин цикла C. ⊠

Замечание. В теореме о циклах и разрезах цикл может быть и не простым.

8.4 Пространства суграфов

Пусть дан граф $G = (X, U; P)$ с пронумерованным множеством ребер $U = \{u_1, u_2, \ldots, u_m\}$, $m \geq 1$. Все его суграфы $G' = (X, U'; P)$ взаимно однозначно соответствуют всевозможным подмножествам ребер $U' \subseteq U$, а каждое такое подмножество (опять же взаимно однозначно) задается вектором

$$\mathbf{a}(G') = \mathbf{a}(U') = (a_1, a_2, \ldots, a_m),$$

у которого

$$a_i = \begin{cases} 1, \text{ если } u_i \in U' \\ 0, \text{ если } u_i \notin U'. \end{cases}$$

В дальнейшем будем называть суграфом также подмножество U' ребер графа и соответствующий ему вектор $\mathbf{a}(U')$.

Определим над суграфами U_1, U_2 графа G операцию сложения:

$U_1 + U_2 = (U_1 \cup U_2) \setminus (U_1 \cap U_2)$,
что соответствует для векторов

$$(a_1, a_2, \ldots, a_m) + (b_1, b_2, \ldots, b_m) = (a_1 + b_1, a_2 + b_2, \ldots, a_m + b_m)$$

покомпонентному сложению по модулю два:

$0 + 0 = 0, 1 + 0 = 1, 0 + 1 = 1, 1 + 1 = 0.$

Это означает, что при сложении суграфов ребра накладываются друг на друга так, что ребро, попадая на ребро, с ним взаимно уничтожается.

Множество суграфов данного графа G образует относительно сложения коммутативную (абелеву) группу с нулем – нуль-вектором $\mathbf{0} = (0, 0, ..., 0)$, соответствующим пустому подмножеству $U' = \varnothing$, а обратным элементом для любого элемента является он сам: **a + a = 0** (по определению операции сложения). Введенную таким образом группу можно рассматривать как линейное пространство над полем коэффициентов $\{0, 1\}$ со сложением по модулю два и обычным умножением $(0 \cdot 0 = 0, 1 \cdot 0 = 0 \cdot 1 = 0, 1 \cdot 1 = 1)$.

Это пространство обозначим через $\mathbf{S}(G)$ и назовем *пространством суграфов* графа G (с фиксированной нумерацией ребер).

Линейная зависимость системы элементов $\mathbf{a}_1$, $\mathbf{a}_2, \ldots$, $\mathbf{a}_k \in \mathbf{S}(G)$ $(k \geq 1)$ означает существование такой непустой подсистемы, которая в сумме дает нулевой элемент $\mathbf{0}$. Элементы

$\mathbf{e}_1 = (1, 0, \ldots, 0)$, $\mathbf{e}_2 = (0, 1, \ldots, 0), \ldots$, $\mathbf{e}_m = (0, 0, \ldots, 1)$
(однореберные графы) линейно независимы, а любой суграф выражается через них линейно:

$(a_1, a_2, \ldots, a_m) = a_1\ \mathbf{e}_1 + a_2\ \mathbf{e}_2 + \ldots + a_m\ \mathbf{e}_m.$

Таким образом, эти m однореберных суграфа образуют базис пространства $\mathbf{S}(G)$, поэтому его размерность

$\dim \mathbf{S}(G) = m = m(G).$

8.4.1 Пространство циклов

Если два суграфа графа G таковы, что множество ребер каждого из них (вместе с инцидентными вершинами) образуют цикл, то их сумма в $\mathbf{S}(G)$ не обязательно дает цикл: например, два цикла без общих вершин, или два совпадающих цикла.

Для того чтобы замкнуть подмножество суграфов – циклов в $\mathbf{S}(G)$ относительно сложения, вводится понятие *квазицикла* – суграфа, все вершины которого обладают четными валентностями. Квазициклом будем также называть множество ребер этого суграфа, и соответствующий вектор.

Из определения квазицикла и операции сложения следует, что сумма двух квазициклов есть квазицикл. Поэтому множество всех квазициклов образует в $\mathbf{S}(G)$ подпространство – *пространство циклов* графа G (с фиксированной нумерацией ребер). Обозначим его $\mathbf{L}(G)$.

Пусть теперь T – некоторый каркас графа G. Каждой хорде этого каркаса поставим в соответствие тот единственный цикл, который она образует вместе с некоторыми ребрами T, и тем самым получим систему $\mathbf{L}(T) \subseteq \mathbf{L}(G)$ из $\lambda(G)$ квазициклов.

Покажем, что $\mathbf{L}(T)$ – базис пространства циклов.

Д о к а з а т е л ь с т в о . 1. Система $\mathbf{L}(T)$ линейно независима, так как каждый ее квазицикл содержит ребро (хорду каркаса T), не входящее ни в один из остальных квазициклов. Это хорошо видно при такой нумерации ребер, когда сначала идут хорды каркаса (их число равно λ), а потом уже ребра каркаса T. Тогда векторы системы имеют вид:

$\mathbf{a}_1 = (1,\ 0,\ 0, \ldots,\ 0,\ a^1_{\lambda+1}, \ldots,\ a^1_m)$,
$\mathbf{a}_2 = (0,\ 1,\ 0, \ldots,\ 0,\ a^2_{\lambda+1}, \ldots,\ a^2_m)$,
. . . .
$\mathbf{a}_\lambda = (0,\ 0,\ 0, \ldots,\ 1,\ a^\lambda_{\lambda+1}, \ldots,\ a^\lambda_m)$.

Далее будем предполагать именно такую нумерацию ребер (до ее отмены).

2. Всякий квазицикл в $\mathbf{L}(G)$

$\mathbf{a} = (a_1, a_2, \ldots, a_\lambda,\ a_{\lambda+1}, \ldots, a_m) \in \mathbf{L}(G)$

есть сумма квазициклов некоторого подмножества системы $\mathbf{L}(T)$ (в случае $\mathbf{a} = \mathbf{0}$ – пустого)

$\mathbf{a} = a_1\mathbf{a}_1 + a_2\ \mathbf{a}_2 + \ldots + a_\lambda\ \mathbf{a}\ _\lambda$.

Чтобы подтвердить это, достаточно показать, что

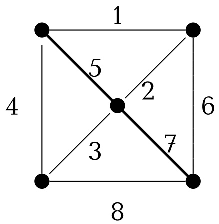
$\mathbf{b} = a_1\, \mathbf{a}_1 \;+\; a_2\, \mathbf{a}_2 \;+\; \ldots \;+\; a_\lambda\, \mathbf{a}_\lambda + \mathbf{a} \;=\; \mathbf{0}$
(так как в $\mathbf{S}(G)$ вычитание равносильно сложению).

Первые λ компонент вектора **b** нулевые, так как они имеют вид $a_i + a_i$, а $0+0 = 1+1 = 0$ (сложение по модулю два). Но вектор **b**, сумма квазициклов, сам есть квазицикл. Так как его первые λ компонент, соответствующие хордам каркаса T, равны нулю, то этот суграф – квазицикл – может содержать только ребра из T. Однако из ребер каркаса невозможно образовать ни одного цикла (все валентности в T не могут быть четными); значит последние $m - \lambda$ компонент вектора **b** – тоже нули. Таким образом, $\mathbf{L}(T)$ – базис пространства циклов $\mathbf{L}(G)$ и размерность его равна
$\dim \mathbf{L}(T) = |\ \mathbf{L}(T)\ | = \lambda(G)$. ☒

Определение. *Матрица* $\Lambda(G) = \Lambda = \Lambda^m_\lambda$, *строки которой представляют собой векторы некоторого базиса пространства циклов* $\mathbf{L}(T)$, *называется цикломатической матрицей.*

Пример 8.4.
Ребра каркаса T выделены на рисунке жирными линиями. Ребра пронумерованы таким образом, что первые 4 номера назначены хордам, а остальные ребрам каркаса ($\lambda(G) = 4$).

Цикломатическая матрица (при такой нумерации и при таком каркасе) для этого графа выглядит так:



$\Lambda =$

Λ	1	2	3	4	5	6	7	8
C_1	1	0	0	0	1	1	1	0
C_2	0	1	0	0	0	1	1	0
C_3	0	0	1	0	0	0	1	1
C_4	0	0	0	1	1	0	1	1

Для примера, цикл $(1\ 1\ 0\ 1\ 0\ 0\ 1\ 1)$ является суммой базисных циклов $C_1 + C_2 + C_4$. ◀

8.4.2 Пространство разрезов.

Элемент пространства суграфов $\mathbf{L}(G)$ назовем *квалиразрезом*, если соответствующий суграф имеет с любым циклом в G четное число (возможно, нуль) общих ребер.

Примеры 8.5.
1. Пустой суграф графа G есть квалиразрез (нулевой элемент пространства суграфов) т.к. имеет с любым циклом 0 общих ребер.

2. Всякий разрез – квалиразрез (по теореме 2.8).

3. Если граф G без циклов, то любой его суграф – квалиразрез (циклов нет и число общих ребер – 0).

4. Если x – произвольная вершина графа G, то множество всех непетель, инцидентных x, образует разрез, который является также квалиразрезом. Такой разрез называется *центральным* с центром в вершине x. ◄Приставка "квали" (по Зыкову) имеет смысл противоположный "квази": если "квазицикл" обобщение цикла, то "квалиразрез" – частный случай разреза (за исключением нулевого).

Теорема 8.9 *Непустое подмножество $U' \subseteq U$ ребер графа G является квалиразрезом тогда и только тогда, когда U' есть объединение попарно непересекающихся простых разрезов.*

Доказательство. 1. $\Leftarrow$ *Достаточность*. Пусть $U' = \sum_{i=1}^{k} U'_i$, где $U'_1, \ldots, U'_k$ – попарно непересекающиеся простые разрезы. Всякий цикл графа G имеет (по теореме 2.7) четное число общих ребер с каждым U'_i, а значит и с U'. Поэтому U' – квалиразрез.

2. $\Rightarrow$ *Необходимость*. Наоборот, пусть $U' \subseteq U$ $(U' \neq \varnothing)$ определяет квалиразрез (т.е. имеет четное число общих ребер с любым циклом, может быть и нуль).

Во-первых, U' – разрез графа G. Действительно, предположим, что U' – не разрез. Тогда ребро $u \in U'$ не может быть перешейком в суграфе G', порожденном подмножеством ребер $(U \setminus U') \cup \{u\}$, поэтому в G' существует цикл, содержащий u, т.е. имеющий в G ровно одно общее ребро с U', что противоречит определению квалиразреза.

Далее, выберем в U' некоторое подмножество U'' – простой разрез графа G. Если $U' \setminus U'' = \varnothing$, то U' есть объединение системы простых разрезов, состоящий из единственного разреза U''.

Если $U' \setminus U'' \neq \varnothing$, то $U' \setminus U''$ – квалиразрез, так как $U' \setminus U''$ имеет четное число общих ребер с любым циклом (U' имеет четное число общих ребер с любым циклом, т.к. U' разрез; U'' – тоже, т.к. U'' простой разрез; тогда и $U' \setminus U''$ имеет четное число общих ребер с любым циклом).

Если $U'\backslash U''$ не простой разрез, то снова в этом суграфе выделим простой разрез U''' и получим $U' \setminus U'' \setminus U'''$ – квалиразрез (по той же причине), и т.д.

В конце концов исходное множество U' окажется представленным в виде объединения попарно непересекающихся простых разрезов, что и требовалось. ☒

Учитывая результат теоремы 2.9, квалиразрез можно определить как подмножество $U' \subseteq U$ ребер графа G, если U' представимо в виде объединения попарно непересекающихся простых разрезов (в частности, $U' = \emptyset$). Как следует из теоремы 2.9 оба определения квалиразреза равносильны.

Следствие. *Сумма двух квалиразрезов есть квалиразрез.*

Доказательство. Пусть U_1 и U_2 – квалиразрезы, а C – множество ребер произвольного квазицикла графа G. Числа $\mid C \cap U_1 \mid, \mid C \cap U_2 \mid$ четны (по определению квалиразреза). Тогда четно и число

$$|C \cap (U_1 + U_2)| = |C \cap [(U_1 \cup U_2) \setminus (U_1 \cap U_2)] =$$

$$= |C \cap U_1| + |C \cap U_2| - 2|C \cap U_1 \cap U_2|.$$

Таким образом, сумма квалиразрезов есть квалиразрез, то есть множество квалиразрезов замкнуто относительно операции сложения. ☒

Поэтому

множество $\mathbf{P}(G)$ всех квалиразрезов образует подпространство в $\mathbf{S}(G)$ – *пространство разрезов*. Покажем, что размерность пространства разрезов $\mathbf{P}(G)$ равна
$$\rho = m - \lambda = n - \kappa,$$

т.е. некоторая система из ρ квалиразрезов образует базис пространства $\mathbf{P}(G)$.

Доказательство. Пусть T – некоторый каркас графа G. Каждому ребру каркаса T отнесем тот единственный простой разрез, который оно образует вместе с некоторыми хордами (по теореме 2.7). Эта система разрезов и образует базис пространства разрезов.

Рассуждения относительно базиса $\mathbf{P}(G)$ двойственны тем, которые мы проводили при определении размерности пространства циклов.

Во-первых, все квалиразрезы в $\mathbf{P}(G)$ линейно независимы, так как каждый из них содержит ребро, не входящее в другие квалиразрезы. Можно перенумеровать все ребра графа таким образом, чтобы ребра каркаса имели первые $\rho = m - \lambda$ номеров, а хорды – остальные λ. Тогда система разрезов

$$\begin{array}{l}
1\ 0\ 0\ \ldots\ 0\ a^1_{\rho+1}\ \ldots\ a^1_m \\
0\ 1\ 0\ \ldots\ 0\ a^2_{\rho+1}\ \ldots\ a^2_m \\
\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
0\ 0\ 0\ \ldots\ 1\ a^\rho_{\rho+1}\ \ldots\ a^\rho_m
\end{array}$$

образует базис пространства разрезов.

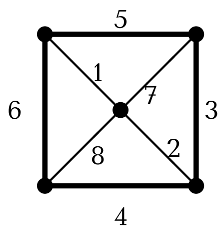
Во-вторых, всякий квалиразрез, отличный от базисного является либо нулевым, либо может быть образован сложением некоторых элементов $\mathbf{P}(T)$. Действительно, выберем элементы в $\mathbf{P}(T)$ так, чтобы их сумма совпадала с заданным квалиразрезом на ребрах каркаса T. Остальные ребра тоже автоматически совпадут, так как в противном случае сложение заданного квалиразреза с построенной суммой дала бы непустой квалиразрез, в котором присутствуют только хорды каркаса. Но это невозможно, так как никакое множество хорд не может образовать разреза графа.

Таким образом, $\mathbf{P}(T)$ – базис пространства $\mathbf{P}(G)$ и
$\dim \mathbf{P}(G) = \rho(G)$ (ранг графа G). ☒

Матрица $P(G) = P = P^\rho_m$ с $\rho(G)$ строками и $m(G)$ столбцами над полем $\mathbf{D}\{0,1\}$ (поле вычетов по модулю два), строки которой представляют собой элементы некоторого базиса пространства разрезов $\mathbf{P}(G)$ называется *матрицей разрезов*.

Пример 8.6. Ребра каркаса T выделены на рисунке жирными линиями. Ребра графа пронумерованы таким образом, что первые 4 номера назначены ребрам каркаса, а остальные хордам; $\rho(G) = 4$.

Матрица разрезов для этого графа (при данной нумерации, и для данного каркаса) выглядит следующим образом.



$P =$

P	1	2	3	4	5	6	7	8
P_1	1	0	0	0	1	1	0	0
P_2	0	1	0	0	1	1	1	1
P_3	0	0	1	0	1	0	1	0
P_4	0	0	0	1	0	1	0	1

Для примера, разрез (11010110) является суммой базисных разрезов $P_1 + P_2 + P_4$. ◀

Элементы $\mathbf{a} = (a_1, \ \dots \ , \ a_m)$, $\mathbf{b} = (b_1, \ \dots \ , b_m)$ пространства суграфов назовем *ортогональными*, если

$\sum\limits_{i=1}^{m} a_i \ b_i \ = \ 0$ (определение ортогональности векторов).

Здесь сложение по модулю два, умножение обычное.

Из существования базиса, состоящего из простых циклов в $\mathbf{L}(G)$, и базиса $\mathbf{P}(G)$ простых разрезов, и того, что любой цикл и любой простой разрез имеют четное число общих ребер, получим, что каждый квазицикл ортогонален простому разрезу. Отсюда следует, что подпространства $\mathbf{L}(G)$ и $\mathbf{P}(G)$ ортогональны.

А это значит, что для любой матрицы разрезов $P(G)$ и любой цикломатической матрицы $\Lambda(G)$, (заданных при одной и той же нумерации ребер)

$$P_{\rho\times m} \ \times \Lambda^T_{m\times\lambda} \ = \mathbf{0}_{\rho\times\lambda}, \ \Lambda_{\lambda\times m} \times \ P^T_{m\times\rho} \ = \mathbf{0}_{\lambda\times\rho}.$$

Глава 9

Потоки в сетях

9.1 Задача о максимальном потоке

9.1.1 Постановка задачи

(*Транспортной*) ***сетью*** называется связный ориентированный униграф без петель, удовлетворяющий следующим условиям:

1. Существует только одна вершина с нулевой степенью захода. Эта вершина называется источником и обозначается s.

2. Существует только одна вершина с нулевой степенью исхода. Эта вершина называется стоком и обозначается t.

3. Каждой дуге $u = (x_i, x_j)$ сопоставляется некоторое неотрицательное число – ***пропускная способность*** дуги. Это число обозначается $c(u)$, или $c(x_i, x_j)$, или просто c_{ij}. Если не существует ребра u, ориентированного из x_i в x_j, то $c_{ij} = 0$.

Потоком f в сети называется функция, сопоставляющая каждому ребру $u = (x_i, x_j)$ неотрицательное число $f_{ij} = f(x_i, x_j)$ так, что

1. $f_{ij} \leq c_{ij}$.

2. $$\sum_{x_j \in \Gamma x_i} f_{ij} - \sum_{x_k \in \Gamma^{-1} x_i} f_{ki} = \begin{cases} v, & x_i = s, \\ -v, & x_i = t, \\ 0, & x_i \neq s, t. \end{cases}$$

Условие 1 – это ограничение по пропускной способности; оно требует, чтобы величина потока по дуге не превышала ее пропускной способности.

Условие 2 – это условие сохранения потока. Для вершин, отличных от s и t, поток, втекающий в вершину, равен потоку, вытекающему из вершины.

В вершинах s и t: из s вытекает поток величины v; в t втекает поток величины v;

$$v = \sum_{x \in \Gamma s} f_{sx} = \sum_{x \in \Gamma^{-1} t} f_{xt} \text{ – величина потока.}$$

Задача заключается в нахождении такого множества потоков по дугам, чтобы при заданных пропускных способностях величина потока v была максимальной.

Некоторый разрез сети будем обозначать парой $(S, \bar{S})$.
Здесь $S \cup \bar{S} \ = \ X, S \cap \bar{S} = \varnothing$, S и $\bar{S}$ – множества вершин, которые разделены разрезом. Будем говорить, что разрез $(S, \bar{S})$ разделяет источник s и сток t, если $s \in S$, а $t \in \bar{S}$. Такой разрез назовем $s - t$-разрезом.

Величина разреза, или просто разрез (пропускная способность) определяется как

$$c(S, \bar{S}) = \sum_{x_i \in S, x_j \in \bar{S}} c_{ij}.$$

Сумму потоков на дугах, ориентированных из S в $\bar{S}$ обозначим и определим как

$$f(S, \bar{S}) = \sum_{x_i \in S, x_j \in \bar{S}} f_{ij}.$$

9.1.2 Теорема о максимальном потоке и минимальном разрезе

Лемма 4 *Пусть f – поток из s в t в сети. Если $(S, \bar{S})$ – разрез, разделяющий s и t, то*

$$v = f(S, \bar{S}) - f(\bar{S}, S)$$

Доказательство. Из определения потока

$$f(s, X) - f(X, s) = v,$$

$$f(x, X) - f(X, x) = 0, \ x \neq s, t,$$

$$f(t, X) - f(X, t) = -v.$$

Просуммируем теперь все потоки по $x \in S$ (помня при этом, что $s \in S, t \in \bar{S}$).

$$v = \sum_{x \in S} [f(x, X) - f(X, x)] = f(S, X) - f(X, S).$$

Так как $S \cup \bar{S} = X$, имеем

$$f(S, S \cup \bar{S}) - f(S \cup \bar{S}, S) =$$

$$= f(S, S) + f(S, \bar{S}) - f(S, S) - f(\bar{S}, S) =$$

$$= f(S, \bar{S}) - f(\bar{S}, S). \boxtimes$$

Следствие 1. $v \leqslant c(S, \bar{S})$.

Доказательство. Так как $f_{ij} \geqslant 0$, то

$$v = f(S, \bar{S}) - f(\bar{S}, S) \leqslant f(S, \bar{S}) \leqslant c(S, \bar{S}). \boxtimes$$

Заметим,что равенство может быть достигнуто, когда $f(\bar{S}, S) = 0$; тогда и $v = f(S, \bar{S}) = c(S, \bar{S})$.

Назовем:

дуга (x_i, x_j) насыщенная,если $f_{ij} = c_{ij}$;

дуга (x_i, x_j) ненасыщенная, если $f_{ij} < c_{ij}$;

дуга (x_i, x_j) положительная, если $f_{ij} > 0$;

дуга (x_i, x_j) нулевая, если $f_{ij} = 0$.

Другими словами, $v = f(S, \bar{S}) = c(S, \bar{S})$ тогда и только тогда, когда дуги из S в $\bar{S}$ являются насыщенными, а дуги из $\bar{S}$ в S – нулевыми.

s-t-разрез называется минимальным, если не существует другого такого разреза $(S', \bar{S}')$, что $c(S', \bar{S}') < c(S, \bar{S})$.

Следствие 2. *Пусть f – поток, $(S, \bar{S})$ – такой s-t-разрез, что $f(S, \bar{S}) = c(S, \bar{S})$.*
Тогда $f(S, \bar{S})$ – максимальный поток, а $(S, \bar{S})$ – минимальный разрез (т.е. $c(S, \bar{S})$ минимальный).

Доказательство. Пусть $f^\star$ – максимальный поток, а $(S^\star, \bar{S}^\star)$ – минимальный разрез. Из следствия 1 $f(S^\star, \bar{S}^\star) = c(S^\star, \bar{S}^\star)$. Поэтому

$$f(S, \bar{S}) \leq f(S^\star, \bar{S}^\star) \leq c(S^\star, \bar{S}^\star) \leq c(S, \bar{S})$$

По предположению, $f(S, \bar{S}) = c(S, \bar{S})$, следовательно,

$$f(S, \bar{S}) = f(S^\star, \bar{S}^\star) = c(S^\star, \bar{S}^\star) = c(S, \bar{S})$$

Таким образом, $f(S, \bar{S})$ – максимальный поток, а $c(S, \bar{S})$ – минимальный разрез. ⊠

В общем-то мы уже доказали, что величина максимального потока равна минимальному разрезу, но мы еще и покажем как найти такой разрез. Классическое конструктивное доказательство принадлежит Форду и Фалкерсону (книга: Л.Форд, Д.Фалкерсон. Потоки в сетях. – М.,Мир, 1966. – 276 с.).

Теорема 9.1 (Форда – Фалкерсона) *(о максимальном потоке и минимальном разрезе).*

Для любой сети максимальная величина потока из s в t равна минимальной пропускной способности разреза, разделяющего s и t.

Доказательство. В силу леммы и следствий достаточно показать, что существуют такие поток f и разрез $(S, \bar{S})$, для которых

$$f(S, \bar{S}) = c(S, \bar{S}).$$

Считаем, что поток максимальный (очевидно, что таковой существует). Найдем такой разрез $(S, \bar{S})$,что

$$f(S, \bar{S}) = c(S, \bar{S})$$

и

$$f(\bar{S}, S) = 0.$$

Итак, пусть f – максимальный поток. Определим S рекуррентно следующим образом:

- включаем s в S: $s \in S$;
- если $x \in S$ и $f(x, y) < c(x, y)$, то включаем y в S: $y \in S$;
- если $x \in S$ и $f(x, y) > 0$, то включаем y в S: $y \in S$.

Процесс включения вершин в S продолжается до тех пор, пока множество S нельзя будет расширить дальше.

Покажем, что при таком способе включения вершин в множество S, вершина t в S не попадет, т.е. $t \in \bar{S}$.

Допустим, что это не так (т.е. t можно включить в S). Тогда из определения S следует, что существует цепь из s в t (это не обязательно ориентированная цепь)

$$s = x_1, x_2, \ldots, x_l = t,$$

такая, что для прямых дуг (x_i, x_{i+1})

$f(x_i, x_{i+1}) < c(x_i, x_{i+1})$,

а для всех обратных дуг $(x_{i+1}, x_i) > 0$.

Пусть $\delta_1 = \min\limits_{x_i, x_{i+1}} [c(x_i, x_{i+1}) - f(x_i, x_{i+1})]$

(минимум по всем прямым дугам цепи);

$\delta_2 = \min\limits_{x_{i+1}, x_i} [f(x_{i+1}, x_i)]$

(минимум по всем обратным дугам цепи);

$\delta = \min [\delta_1, \delta_2]$.

Изменим теперь поток цепи следующим образом: увеличим поток на всех прямых дугах на величину δ, а на всех обратных дугах уменьшим поток на величину δ.

В результате получится новый допустимый поток, имеющий величину $v+\delta$. Добавление δ не приводит к превышению пропускной способности ни по одной дуге, а вычитание не приводит к отрицательным потокам на обратных дугах (по способу выбора δ).

Итак, поток стал $v + \delta$. Но это значит, что поток не был максимальным, и, значит, $t \in \bar{S}$.

Таким образом, $(S, \bar{S})$ есть разрез, разделяющий s и t. Кроме того,

$$\left.\begin{aligned} f(x_i, x_j) &= c(x_i, x_j) \\ f(x_j, x_i) &= 0 \end{aligned}\right\} x_i \in S, x_j \in \bar{S}.$$

Поэтому

$f(S, \bar{S}) = c(S, \bar{S})$,

$f(\bar{S}, S) = 0$. ⊠

9.1.3 Алгоритм Форда – Фалкерсона

Алгоритм начинает работу с произвольного допустимого потока (возможно, нулевого). Затем пытаемся увеличить величину потока с помощью систематического поиска всех возможных цепей из s в t, на которых величину потока можно увеличить (дополняющие цепи).

Поиск дополняющих цепей производится с помощью расстановки меток. Метки указывают, на каких дугах и на сколько можно увеличить поток. Когда найдена одна из таких цепей, поток вдоль нее увеличивается. Затем все метки стираются и вновь полученный поток используется в качестве исходного при новой расстановке меток. Алгоритм заканчивает работу, когда нельзя найти ни одну дополняющую цепь.

Расстановка меток. Вершина может находиться в одном из трех состояний:

- вершина помечена и просмотрена (т.е. вершина имеет метку и все смежные с ней вершины “обработаны");
- вершина помечена, но не просмотрена (т.е. вершина имеет метку, но смежные с ней вершины не “обработаны");
- вершина не помечена.

Метка некоторой вершины x имеет вид $L(x) = [\pm y, \delta(x)]$. Часть метки $+y$ означает, что поток может быть увеличен вдоль дуги (x, y). Часть метки $-y$ означает, что поток может быть уменьшен вдоль дуги (y, x). В обоих случаях $\delta(x)$ показывает максимальную величину, на которую можно увеличить поток от s к x вдоль дополняющей цепи.

Сначала все вершины не имеют меток.

Описание алгоритма.

Шаг 0. Инициализация.

Положим все дуговые потоки равными нулю: $f_{ij} := 0$.

Шаг 1. Назначить вершине s метку $L(s) := [+s, \delta(s) = \infty]$. Теперь вершина s помечена, но не просмотрена. Все остальные вершины не помечены.

Шаг 2. Выбрать некоторую помеченную, но не просмотренную вершину x_i; пусть ее метка будет $[\pm x_k, \delta(x_i)]$.

1. Каждой непомеченной вершине x_j, смежной с x_i $(x_j \in \Gamma x_i)$, для которой $f_{ij} < c_{ij}$ назначить метку $[+x_i, \delta(x_j)]$, где

$$\delta(x_j) = \min[\delta(x_i), c_{ij} - f_{ij}].$$

2. Каждой непомеченной вершине $x_j \in \Gamma^{-1} x_i$, для которой $f_{ji} >$ 0 назначить метку $[-x_i, \delta(x_j)]$, где

$$\delta(x_j) = \min[\delta(x_i), f_{ji}].$$

Теперь вершина x_i помечена и просмотрена, а вершина x_j, метка которой назначена в пунктах 1 или 2, помечена, но не просмотрена. Каким либо образом отмечается, что вершина x_i просмотрена.

Шаг 3. Проверка.

Если на шаге 2 какая-либо вершина помечена, то:

если помечена вершина t, то на **Шаг 4**;

если помечена любая другая вершина, то на **Шаг 2**.

Если на **Шаге 2** нельзя назначить никаких меток, то алгоритм заканчивает работу с некоторым максимальным потоком. При этом множество помеченных вершин – это S, а множество непомеченных вершин – множество $\bar{S}$, и $(S, \bar{S})$ – минимальный разрез.

Шаг 4. Выбрать вершину $x = t$; на **Шаг 5**.

Шаг 5. Увеличение потока.

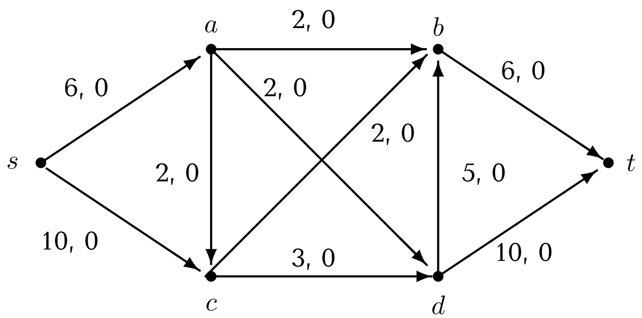
1. Если метка вершины x имеет вид $[+z, \delta(x)]$, то изменить поток вдоль дуги (z, x) с $f(z, x)$ на $f(z, x) + \delta(t)$.

2. Если метка вершины x имеет вид $[-z, \delta(x)]$, то изменить поток вдоль дуги (x, z) с $f(x, z)$ на $f(x, z) - \delta(t)$.

Шаг 6. Если $z = s$, то стереть все метки и вернуться к **Шагу 1**. При этом используется уже увеличенный поток, найденный на **Шаге 5**.

Если $z \neq s$, то положить $x = z$; на **Шаг 5**.

Пример 9.1. На рисунке приведен пример сети; числа на дугах обозначают их пропускные способности и начальные потоки.



Матрица пропускных способностей имеет следующий вид:

$C =$

X	s	a	b	c	d	t
s	0	6	0	10	0	0
a	0	0	2	2	2	0
b	0	0	0	0	0	6
c	0	0	4	0	3	0
d	0	0	5	0	0	10
t	0	0	0	0	0	0

Сначала потоки по всем дугам полагаем равными 0.

Расстановка меток по вершинам.

	s	a	b	c	d	t
1	$+s,\infty$	$+s,6$	$+a,2$	$+s,10$	$+a,2$	$+b,2$
2	$+s,\infty$	$+s,4$	$+c,4$	$+s,10$	$+a,2$	$+b,4$
3	$+s,\infty$	$+s,4$	$+d,3$	$+s,6$	$+c,3$	$+d,3$
4	$+s,\infty$	$+s,2$	$+d,2$	$+s,3$	$+a,2$	$+d,2$
5	$+s,\infty$	$+s,2$		$+s,3$		

Примечания. 1.Увеличиваем поток по дугам $(b,t),(a,b),(s,a)$ на 2, стираем метки и начинаем новую разметку с учетом увеличившегося потока $f(s,a) = 2$, $f(a,b) = 2, f(b,t) = 2$. Потоки по остальным дугам – нулевые.

2. Увеличиваем поток по дугам $(b,t),(c,b),(s,c)$ на 4, стираем метки и начинаем новую разметку с учетом увеличившегося потока:

$f(s,c) = 4, f(c,b) = 4, f(b,t) = 6.$

3. Увеличиваем поток по дугам $(d,t),(c,d),(s,c)$ на 3, стираем метки и начинаем новую разметку с учетом увеличившегося потока:

$f(s,c) = 7, f(c,d) = 3, f(d,t) = 3.$

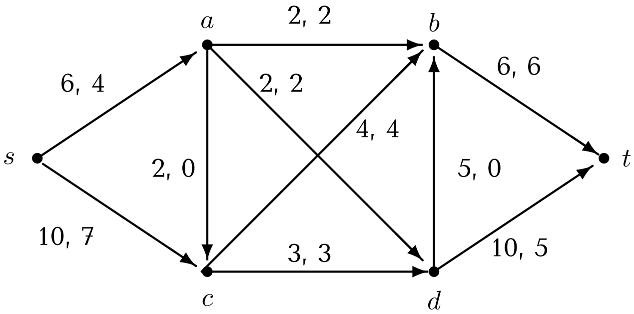
4.Увеличиваем поток по дугам $(d,t),(a,d),(s,a)$ на 2, стираем метки и начинаем новую разметку с учетом увеличившегося потока: $f(s,a) = 4, f(a,d) = 2, f(d,t) = 5.$

5. Другие вершины пометить не удается. Конец разметки.

$S = \{s,a,c\}, \bar{S} = \{b,d,t\}, (S,\bar{S}) = (a,b),(a,d),(c,b),(c,d).$

$v = c(S,\bar{S}) = 11.$

Сеть с распределенными по дугам потоками изображена на рисунке.


a
b
c
d
s
t
2, 2
6, 4
2, 2
6, 6
4, 4
2, 0
5, 0
10, 7
3, 3
10, 5

◄

Глава 10

Экстремальные части графа

10.1 Основные понятия

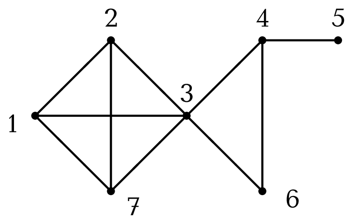
Ряд классических задач теории графов сводится к отысканию частей графа, экстремальных относительно некоторого свойства. Речь может идти также о некоторых подмножествах вершин или ребер и о частях, порожденными этими подмножествами.

Определение. *Часть G' графа G называется максимальной (минимальной) , если в G не существует такой части G'', обладающей этим же свойством, которая содержала бы (содержалась бы в) G', не совпадая с ним.*

Определение. *Наибольшей (наименьшей) частью графа по некоторому свойству называется максимальная (минимальная) часть с наибольшим (наименьшим) числом элементов графа, определяющих это свойство.*

Примеры 10.1.

1. Максимальный полный подграф обыкновенного графа – такой подграф, все вершины которого попарно смежны и он не содержится ни в каком другом подграфе.



Максимальные полные подграфы порождены подмножествами вершин
$\{1, 2, 3, 7\}, \{3, 4, 6\}, \{4, 5\}$.
Наибольший полный подграф порожден подмножеством вершин $\{1, 2, 3, 7\}$.

2. Простой разрез можно считать минимальным, так как мы определили его, как не содержащий никакого другого разреза. Наименьший простой разрез – это простой разрез с наименьшим числом ребер.
3. Максимальный пустой (безреберный) подграф обыкновенного графа – это такой подграф, все вершины которого попарно несмежны и он не содержится ни в каком другом пустом подграфе. Наибольший пустой подграф – это максимальный пустой подграф с наибольшим числом вершин (среди всех максимальных). ◄

Если, кроме того, граф взвешенный (по вершинам или ребрам, либо по тем и другим), то обычно отыскивается наибольшая (наименьшая) часть с наибольшим (наименьшим) весом. Очень часто экстремальная часть не единственна и задача сводится к нахождению либо всех экстремальных частей, либо хотя бы одной экстремальной части.

С экстремальными частями графа связаны различные его числовые характеристики.

Определение. *Граф общего вида называется плотным, если все его вершины попарно смежны.*

Число вершин наибольшего плотного подграфа называется *плотностью* $\varphi(G)$ графа G, а число вершин наибольшего пустого подграфа – неплотностью $\varepsilon(G)$ графа G.

В литературе встречается и другая терминология. Приведем ряд синонимов:

- наибольший полный подграф – *клика*, $\varphi(G)$ – кликовое число;
- наибольший пустой подграф – *внутренне устойчивое множество* (ВУМ), независимое множество; неплотность – *число внутренней устойчивости, число независимости* $\varepsilon(G)$.

Другие числовые характеристики мы рассмотрим ниже – по ходу рассмотрения задач.

Задачи отыскания экстремальных частей могут сводиться друг к другу.

Примеры 10.2.
1. *Дополнительным* графом для обыкновенного графа $G = (X, U)$ называется обыкновенный граф $\bar{G} = (X, \bar{U})$ с тем же множеством вершин X, и вершины в нем смежны тогда и только тогда, когда они несмежны в исходном графе G. Переход от матрицы смежности исходного графа к матрице смежности дополнительного графа очень прост – нужно заменить нули на единицы, а единицы – на нули. Очевидно, что $\varphi(G) = \varepsilon(\bar{G})$ и $\varepsilon(G) = \varphi(\bar{G})$. Таким образом, задачи нахождения наибольшего полного подграфа и наибольшего пустого подграфа сводимы друг к другу.
2. Задача о наибольшем паросочетании.

Паросочетанием называется суграф данного графа G, все ребра которого попарно несмежны. Построим *реберный* граф L (*line graph*) исходного графа G следующим образом:

- каждому ребру исходного графа G поставим в соответствие вершину графа L;
- вершины реберного графа смежны, если соответствующие ребра исходного графа смежны.

Очевидно, что наибольшее паросочетание исходного графа G соответствует наибольшему пустому подграфу его реберного графа (каждое наибольшее паросочетание – наибольшему пустому подграфу). Большинство задач на отыскание экстремальных частей являются NP-полными, т.е. не существует алгоритмов полиномиальной сложности для их решения. Поэтому алгоритмы решения этих задач – переборные алгоритмы с различными способами сокращения перебора; это, как правило, алгоритмы поиска в глубину на дереве решений. Несмотря на сводимость некоторых задач друг к другу, существует множество алгоритмов, использующих те или иные особенности задач.

10.2 Покрытия

Общую задачу о покрытиях (и разбиениях) мы рассмотрели в главе 2. В этом разделе мы рассмотрим ряд задач, связанных с

ЗНП (задачей о наименьшем покрытии). Пусть задан обыкновенный граф $G = (X, U)$.

Определение. *Вершинно-реберное покрытие: множество вершин $S \subseteq X$, такое, что любое ребро графа G инцидентно по крайней мере одной вершине из S*

$\forall u \in U\ \exists x \in S\ [I(x, u)]$

(другие термины – *опорное* множество, или просто *опора*) .

Определение. *Вершинно-вершинное покрытие: множество вершин $S \subseteq S$, такое, что любая вершина из $X \setminus S$ смежна по крайней мере с одной вершиной из S*

$\forall x \in X \setminus S\ \exists y \in S\ [J(x, y)]$

(другие термины – *внешне устойчивое* множество, *доминирующее* множество) .

Определение. *Реберное (реберно-вершинное) покрытие: множество ребер $V \subseteq U$, такое, что любая вершина графа инцидентна по крайней мере одному ребру из V*

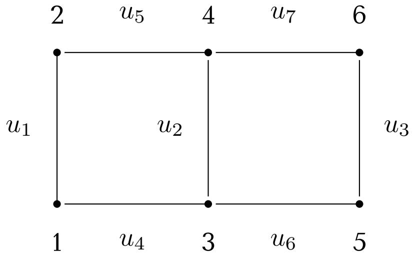
$\forall x \in X\ \exists u \in U\ [I(x, u)]$

(другие термины – *покрытие*, *накрытие*).

Напомним, что предикат $I(x, u)$ определяется как “ребро u инцидентно вершине x", а предикат $J(x, y)$ – как “вершина x смежна с вершиной y".

Покрытия *минимальны*, если они не содержат собственных подмножеств, являющихся покрытиями. *Наименьшие* покрытия – это минимальные покрытия с наименьшим числом элементов.

Пример 10.3.
Для графа, приведенного на рисунке,



минимальные вершинно-реберные покрытия (минимальные опоры): {1, 4, 5}, {2, 3, 6}, {1, 3, 4, 6}, {2, 3, 4, 5}. Из них наименьшие: $\{1,4,5\}$, $\{2,3,6\}$.

Минимальные вершинно-вершинные покрытия: {1, 3, 5}, {1, 4, 5}, {1, 2, 5}, $\{1,2,6\},\dots,\{1,6\},\{3,4\}$. Из них наименьшие – {1, 6}, {3, 4}.

Минимальные реберно-вершинные покрытия: $\{u_1,u_2,u_3\}$, $\{u_1,u_2,u_6,u_7\}$, $\{u_1,u_3,u_4,u_5\}$, $\{u_1,u_3,u_6,u_7\},\dots$ Из них наименьшее – $\{u_1,u_2,u_3\}$. ◀

Введем ряд чисел, характеризующих покрытия.

$\delta(G)$ – *опорное* число, число вершин наименьшего вершинно-реберного покрытия.

$\beta(G)$ – число *внешней устойчивости* (число *всесмежности*), число вершин наименьшего вершинно-вершинного покрытия.

$\iota(G)$ – число ребер наименьшего реберно-вершинного покрытия, *накрывающее* число.

З а м е ч а н и е. Определения, введенные нами для обыкновенных графов, можно распространить и на ориентированные графы без петель (кроме реберного покрытия). Например, вершинно-вершинное покрытие $S \subseteq X$ такое, что для любой вершины $x \in X \backslash S$ существует дуга, идущая из некоторой вершины из S в вершину x. ▶

Теорема 10.1 (Галлаи) *Для графа $G = (X,U)$ множество $S \subseteq X$ есть пустой подграф тогда и только тогда, когда множество $\bar{S} = X \setminus S$ является вершинно-реберным покрытием (опорой).*

Для краткости будем называть пустой подграф – ВУМ (внутренне устойчивое множество), вершинно-реберное покрытие – опорой.

Д о к а з а т е л ь с т в о . $\Leftarrow$ 1. *Достаточность.* Пусть $\bar{S} \subseteq X$ – опора, тогда $S = X \setminus \bar{S}$ – ВУМ. Действительно, любое ребро инцидентно хотя бы одной вершине из $\bar{S}$, поэтому в S вершины попарно несмежны: в противном случае хотя бы одна вершина из этой пары должна была бы войти в опору.

$\Rightarrow$ 2. *Необхдимость.* Пусть $S \subseteq X$ – ВУМ, т.е. в S все вершины попарно несмежны. Другими словами, S – ВУМ, если любое ребро соединяет вершину из S с вершиной из его дополнения, но это и есть определение опоры. ☒

С л е д с т в и е . *Для обыкновенного графа*

$\varepsilon(G) + \delta(G) = n(G)$.

10.2.1 Задача о наименьшем покрытии

Общую задачу о наименьшем покрытии (ЗНП) мы уже рассмотрели в главе о бинарных отношениях. В данном разделе мы рассмотрим лишь каким образом задачи о нахождении наименьших покрытий на графах сводятся к общей задаче ЗНП.

10.2.1.1 Вершинно-реберное покрытие (ВРП)

Рассмотрим траспонированную матрицу инциденций A^{T} графа G. Задача нахождения наименьшего ВРП сводится к нахождению наименьшего числа таких столбцов этой матрицы, чтобы каждая строка матрицы содержала единицу хотя бы в одном из выбранных столбцов (или логическая сумма этих столбцов давала бы единичный столбец). Для решения этой задачи можно использовать общий алгоритм ЗНП.

Пример 10.4. Для графа из предыдущего примера соответствующая матрица A^{T} выглядит следующим образом:

A^{T}	1	2	3	4	5	6
u_1	1	1	0	0	0	0
u_2	0	0	1	1	0	0
u_3	0	0	0	0	1	1
u_4	1	0	1	0	0	0
u_5	0	1	0	1	0	0
u_6	0	0	1	0	1	0
u_7	0	0	0	1	0	1

Наименьшие ВРП – $\{1,4,5\}, \{2,3,6\}$. ◄

10.2.1.2 Вершинно-вершинное покрытие (ВВП)

Для нахождения наименьшего ВВП алгоритм решения ЗНП применяется к матрице $(E+R)$, где R – матрица смежности, E – диагональная матрица.

З а м е ч а н и е. Здесь нужно учитывать, что вершина покрывает саму себя. ►

Пример 10.5.
Для графа из рассматриваемого нами примера матрица

$R^{\star} = (E + R)$ имеет вид:

$R^{\star}$	1	2	3	4	5	6
1	1	1	1	0	0	0
2	1	1	0	1	0	0
3	1	0	1	1	1	0
4	0	1	1	1	0	1
5	0	0	1	0	1	1
6	0	0	0	1	1	1

Наименьшие ВВП – $\{1, 6\}, \{3, 4\}$. ◄

10.2.1.3 Реберно-вершинное покрытие (РВП)

Для нахождения наименьшего РВП алгоритм решения ЗНП применяется к матрице инциденций графа G над $\mathbf{B}\{0, 1\}$.

10.2.1.4 Перечисление покрытий

Рассмотрим задачу нахождения всех минимальных покрытий на примере вершинно-реберных покрытий для обыкновенного графа $G = (X, U)$. Напомним, что ВРП – это такое множество вершин $S \subseteq X$, что любое ребро графа G инцидентно по крайней мере одной вершине из S. Определение ВРП можно выразить формально в в виде высказывания

$$\forall u \in U \ \exists x \in X[x \in S \wedge I(x, u)] = 1, \tag{10.1}$$

где $I(x, u)$ – предикат "вершина x и ребро u инцидентны".

Высказывание (10.1) тождественно истинно, так как является определением. Для множеств $X = \{x_1, \dots, x_n\}$ вершин и $U = \{u_1, \dots, u_m\}$ ребер высказывание (10.1) распишем более подробно:

$$\forall u \in U[x_1 \in S \wedge I(x_1, u) \vee x_2 \in S \wedge I(x_2, u) \vee \ldots \vee x_n \in S \wedge I(x_n, u)] =$$
$$= [x_1 \in S \wedge I(x_1, u_1) \vee x_2 \in S \wedge I(x_2, u_1) \vee \ldots \vee x_n \in S \wedge I(x_n, u_1)]$$
$$\bigwedge[x_1 \in S \wedge I(x_1, u_2) \vee x_2 \in S \wedge I(x_2, u_2) \vee \ldots \vee x_n \in S \wedge I(x_n, u_2)]$$
$$\bigwedge[\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots]$$
$$\bigwedge[x_1 \in S \wedge I(x_1, u_m) \vee x_2 \in S \wedge I(x_2, u_m) \vee \ldots \vee x_n \in S \wedge I(x_n, u_m)]$$

Для упрощения записи вместо высказывания $x \in S$ введем логическую переменную x, такую, что

$$x = \begin{cases} 1, & \text{если} \quad x \in S \\ 0, & \text{если} \quad x \notin S \end{cases}$$

Кроме того, заменим обозначение дизъюнкции $x \vee y$ на сложение $x+y$, конъюнкции $x \wedge y$ – на умножение xy. Тогда высказывание ((10.1)) "S есть вершинно-реберное покрытие" запишется в виде

$$\prod_{j=1}^{m} [x_1 I(x_1, u_j) + x_2 I(x_2, u_j) + \cdots + x_n I(x_n, u_j)] = 1.$$

10.3 Паросочетания

Определение. *Множество $W \subseteq U$ попарно несмежных ребер графа $G = (X, U; P)$ и порожденный им суграф называется паросочетанием.*

Паросочетание максимальное, если оно не содержится ни в каком другом паросочетании, и наибольшее, если оно максимальное и содержит наибольшее число ребер среди всех максимальных паросочетаний графа G. Обычно рассматривается задача нахождения наибольшего паросочетания. Число ребер наибольшего паросочетания графа G обозначим через $\pi(G)$.

Примеры 10.6. На графах, изображенных на рисунках, ребра наибольших паросочетаний выделены жирными линиями. Очевидно, что число ребер в паросочетании $|W| \leq \lfloor n/2 \rfloor$ ($\lfloor x \rfloor$ – целая часть числа x), так как каждое ребро "покрывает" 2 вершины. Для *звездного графа* (*веера*) $|W| = 1$. ◄

Теорема 10.2 (Галлаи) *Для графа без петель и изолированных вершин*

$$\iota(G) + \pi(G) = n(G),$$

где $\iota(G)$ – число ребер наименьшего реберного покрытия.

Доказательство. Зафиксировав граф G, обозначим $\iota = \iota(G), \pi = \pi(G), n = n(G)$.

Для доказательства теоремы покажем

1. $\iota + \pi \leq n$,

2. $\iota + \pi \geq n$.

1. Пусть $W^\star$ – наибольшее паросочетание. $W^\star$ покрывает $2|W^\star|$ вершин графа G, а X' – множество вершин, не покрытых паросочетанием (не инцидентных ни одному ребру из $W^\star$)

$$X' \subseteq X, \ |X'| = n - 2|W^\star|.$$

Если $X' \neq \varnothing$, то для каждого $x \in X'$ выберем одно ребро, инцидентное x и смежное какому-либо ребру из $W^\star$. Выбранное множество ребер обозначим U'. При этом $|X'| = |U'|$. Очевидно, что $V = U' \cup W^\star$ образует некоторое реберное покрытие.

Если $V^\star$ – наименьшее реберное покрытие, то

$$|V^\star| \leq |V| = n - 2|W^\star| + |W^\star| = n - |W^\star| = n - \pi,$$

откуда

$$\iota + \pi \leq n.$$

2. Пусть $V^\star$ – наименьшее реберное покрытие графа G. Заметим, что суграф, порожденный ребрами $V^\star$ имеет $k \geqslant 1$ компонент связности. В каждой из компонент никакие две вершины со степенью больше 1 не могут быть смежны в $V^\star$ (так как в противном случае при удалении соединяющего их ребра получим другое реберное покрытие с меньшим числом элементов). Это означает, что каждая компонента представляет собой веер (звездный граф), может быть однореберный. Тогда

$$\iota = |V^\star| = n - k.$$

Действительно, пусть n_i – число вершин в i-й компоненте; каждый веер – дерево и число ребер в нем равно $n_i - 1$;

$$\sum_{i=1}^{k} (n_i - 1) = n - k \text{ (так как } V^\star \text{ покрывает все вершины).}$$

Теперь образуем множество W из ребер, выбранных по одному из каждой компоненты $V^\star$. Очевидно, W – паросочетание и $|W^\star| \geq |W| = k = n - \iota$,

$$\pi \geq n - \iota,$$

откуда $\iota + \pi \geq n$.

Таким образом, мы доказали оба неравенства и тем самым – утверждение теоремы. ☒

З а м е ч а н и . Из доказательства теоремы следует, что из наибольшего паросочетания легко получить наименьшее реберное покрытие, и наоборот. Действительно, пусть найдено наибольшее паросочетание $W^\star$. Чтобы получить наименьшее реберное покрытие $V^\star$ нужно к ребрам $W^\star$ добавить ребра, покрывающие те вершины, которые не покрываются паросочетанием $W^\star$.

Наоборот, для того чтобы получить наибольшее паросочетание $W^\star$ из наименьшего реберного покрытия $V^\star$ достаточно выбрать по одному ребру из каждой компоненты $V^\star$. ►

Определение. *Паросочетание, которое является реберным покрытием, называется совершенным.*

Другими словами, реберное покрытие, состоящее из однореберных компонент, есть совершенное паросочетание.

10.3.0.5 Чередующиеся цепи

Пусть W – паросочетание в обыкновенном графе $G = (X, U)$.

Ребра паросочетания и инцидентные им вершины назовем *сильными*, а остальные ребра и вершины – *слабыми*

(темные и светлые, жирные и тонкие, синие и красные и т.д.)

Определение. *Простая цепь ненулевой длины, содержащая попеременно сильные и слабые ребра, называется* ***чередующейся*** *относительно паросочетания W.*

Определение. *Чередующаяся цепь, соединяющая две различные вершины, называется увеличивающей цепью (W-увеличитель, аугментальная цепь).*

Это название происходит из того факта, что если в графе G есть увеличивающая цепь, то можно построить паросочетание с бóльшим числом ребер, чем в W.

В самом деле, в такой цепи число слабых ребер на единицу больше, чем число сильных. Заменив слабые ребра на сильные, то есть исключив из W сильные ребра и включив в W слабые ребра увеличивающей цепи, получим новое паросочетание W' с числом ребер на единицу больше:

$|W'| = |W| + 1$

Теорема 10.3 *Паросочетание W в графе G является наибольшим тогда и только тогда, когда в графе нет увеличивающих относительно W цепей.*

Доказательство. $\Rightarrow$ Пусть W – наибольшее паросочетание. Проведем доказательство от противного. Предположим, что существует увеличивающая цепь P. Пусть P обозначает также множество ребер этой цепи. Никакое ребро в $W \setminus W \cap P$ не может быть смежным ни с каким ребром из P, так как конечные вершины цепи P слабые (по предположению), а любая другая вершина в P сильная. Следовательно, множество ребер

$$W' = (W \setminus W \cap P) \cup (P \setminus W \cap P) = (W \cup P) \setminus W \cap P$$

является паросочетанием, полученным в результате замены ребер из $P \cap W$ (входящих в W) на ребра из $P \setminus W \cap P$ (не входящих в W). Так как конечные вершины цепи слабые, то есть конечные ребра из P не входят в W, то число ребер в $P \setminus W \cap P$ на единицу больше чем в $W \cap P$. Отсюда $|W'| = |W| + 1$, то есть паросочетание не было наибольшим.

$\Leftarrow$ (от противного). Пусть W – паросочетание, относительно которого нет увеличивающей цепи. Предположим, что W не наибольшее и существует W^* – наибольшее паросочетание. Построим суграф G_W на множестве ребер

$$W \cup W^* \setminus W \cap W^*$$

Никакая вершина в G_W не может иметь степень бóльшую чем 2. В противном случае два или более ребер из W или из W^* будут смежными, что противоречит определению паросочетания.

Таким образом, суграф G_W состоит из одной или более компонент, каждая из которых либо пустая, либо есть простая цепь, либо простой цикл, как это изображено на рисунке. Цепи типа 2 и 3 в G_W существовать не могут, так как являются увеличивающими относительно W^* и W сотвѣтственно. Цикл типа 5 с нечетным числом ребер не может существовать в G_W, так как тогда либо два ребра из W, либо два ребра из W^* были бы смежными, что противоречит определению паросочетания.

Остаются возможными компоненты суграфа G_W типа 1 (пустые), типа 4 и циклы типа 5 с четным числом ребер. Для каждого из таких суграфов число ребер из W^* равно числу ребер из W. Следовательно, $|W| = |W^*|$, то есть W – наибольшее паросочетание. $\boxtimes$

Таким образом, идея алгоритма нахождения наибольшего паросочетания сводится к следующему.

Начиная с некоторого произвольного паросочетания W строить последовательность паросочетаний $W_1, W_2, \ldots$, всякий раз отыскивая увеличивающую чередующуюся цепь и увеличивая паросочетание.

Глава 11

Раскраска вершин графа

11.1 Хроматическое число

Определение. *Граф называется k-хроматическим, если его вершины могут быть раскрашены с помощью k цветов так, чтобы смежные вершины не были окрашены в один цвет.*

Другими словами, граф является k-хроматическим, если существует такое разбиение множества его вершин на k непересекающихся классов

$$X_1, X_2, \ldots, X_k, \ X = \bigcup_{i=1}^{k} X_i, X_i \cap X_j = \varnothing, i \neq j,$$

что вершины в каждом классе попарно несмежны, то есть ребра в G соединяют вершины только из разных классов. Такое разбиение называется *раскраской* вершин графа. В таком представлении раскраска есть не что иное как покрытие графа G некоторой системой пустых подграфов.

Замечание. Для раскраски несущественны ориентация ребер, кратность ребер, а существенно лишь отношение смежности. Поэтому для графа общего вида можно перейти к его скелету. ►

$\boxed{\text{Вставить примеры}}$

Наименьшее число цветов, в которое могут быть окрашены вершины графа G называется ***хроматическим*** числом и обозначается $\gamma(G)$.

Некоторые свойства хроматического числа

1. Если $G_1, G_2, \ldots, G_\kappa$ – компоненты связности графа G, то

   $\gamma(G) = \max\{\gamma(G_1), \ldots, \gamma(G_\kappa)\}$

2. Если $G = (X, U)$ – произвольный обыкновенный граф, а G^+ – надграф, полученный добавлением к G вершины x_0, смежной со всеми вершинами графа G, то

   $\gamma(G^+) = \gamma(G) + 1.$

11.2 Оценки хроматического числа

1. Очевидно,

   $\boxed{1 \leq \gamma(G) \leq n(G)}$

   Эта оценка точная, так как нижняя граница достигается для пустого, а верхняя для полного графа.

2. Нижняя оценка

   $\boxed{\varphi(G) \leq \gamma(G)}$

   также является точной, так как достигается для полных графов. Верхнюю же оценку в терминах плотности указать в общем случае нельзя, так как при любой плотности ≥ 2 можно построить граф со сколь угодно большим хроматическим числом.

   Оценки через степени вершин графа

   Обозначим через $\sigma(G) = \max\limits_{x_i \in X} s(x_i)$ наибольшую степень вершины в G и назовем ее *степенью* графа.

3. Оценка Брукса

   $\boxed{\gamma(G) \leq \sigma(G) + 1}$

   является точной, так как для полного графа выполняется равенство.

Известны также следующие оценки. Упорядочим степени вершин графа в неубывающем порядке

$s(1) \geq s(2) \geq \ldots \geq s(n)$

4. Верхняя оценка

   $\boxed{\gamma(G) \leq \max\limits_{1 \leq n} \min\{s(i) + 1, i\}}$ (оценка Powell, Walsh)

5. Нижняя оценка. Пусть a_j определена рекурсивно

   $a_j = n - s(\sum\limits_{i=1}^{j-1} a_i + 1)$, где $\sum\limits_{i}^{j} a_i = 0$, если $j < i$.

   Пусть k – некоторое целое, для которого $\sum\limits_{j=1}^{k-1} < n$; тогда $k \leq \gamma(G)$ (оценка Bondy).

Использование оценок для раскраски вершин Нижние оценки хроматического числа используются в тех алгоритмах раскраски вершин, которые начинают раскраску с минимально возможного числа цветов (с нижней оценки) и постепенно добавляют их.

Верхние – если алгоритм использует перекрашивание в меньшее число цветов, начиная с заведомо большего чем $\gamma(G)$.

11.3 Точные алгоритмы раскраски вершин

Так как при любой допустимой раскраске вершин графа G множество вершин, окрашиваемых в один цвет должно быть пустым подграфом, то всякую раскраску можно интерпретировать как разбиение множества вершин графа на пустые подграфы. Далее, если пустые подграфы расширить до максимальных, то раскраску можно интерпретировать как покрытие вершин графа максимальными пустыми подграфами. В этом случае некоторые вершины могут войти более чем в один максимальный пустой подграф. Это означает существование различных раскрасок (вершину из разных X_i, X_j можно окрасить в любой из цветов i, j).

Алгоритм раскраски вершин графа тогда сводится к следующему. Каким-либо методом находятся все максимальные пустые подграфы (например, как дополнение к вершинно-реберному покрытию), а затем