Введение в математическую логику
Мех-мат МГУ, 1-й курс, весна 2008 г.
Конспект лекций 1 и 2

Л.Д. Беклемишев*

1 Логика высказываний

1.1 Алфавит, буква, слово

Определение 1.1. *Алфавитом* будем называть любое непустое множество. Его элементы называются *символами* (*буквами*).

Определение 1.2. *Словом* в алфавите Σ называется конечная последовательность элементов Σ.

Пример 1.3. Рассмотрим алфавит $\Sigma = \{a, b, c\}$. Тогда $baaa$ является словом в алфавите Σ.

Определение 1.4. Слово, не содержащее ни одного символа (то есть последовательность длины 0), называется *пустым словом* и обозначается ε.

Определение 1.5. *Длина* слова w, обозначаемая $|w|$, есть число символов в w, причём каждый символ считается столько раз, сколько раз он встречается в w.

Определение 1.6. Если x и y — слова в алфавите Σ, то слово xy (результат приписывания слова y в конец слова x) называется *конкатенацией* слов x и y.

*Данный конспект лекций составлен с использованием лекционных материалов ряда сотрудников кафедры математической логики и теории алгоритмов МГУ, в частности конспекта лекций профессора М.Р. Пентуса (2006 г.), на основе программы, разработанной коллективом кафедры.

Определение 1.7. Если x — слово и $n \in \mathbb{N}$, то через x^n обозначается слово

$$\underbrace{xx\ldots x}_{n \text{ раз}}.$$

Положим $x^0 \rightleftharpoons \varepsilon$ (знак $\rightleftharpoons$ читается «равно по определению»).

Пример 1.8. По принятым соглашениям $ba^3 = baaa$ и $(ba)^3 = bababa$.

Определение 1.9. Множество всех слов в алфавите Σ обозначается Σ^*.

Определение 1.10. Подмножества множества Σ^* для некоторого алфавита Σ называются *словарными множествами*. В лингвистике и информатике словарные множества также часто называют *языками*.

Утверждение 1.11. *Если алфавит Σ конечен или счётен, то множество Σ^* счётно.*

В самом деле, для любого конечного подмножества $\Sigma_0 \subseteq \Sigma$ множество всех слов фиксированной длины в алфавите Σ_0 конечно. Следовательно, Σ_0^* является объединением счётного числа конечных множеств, а значит, таково и множество $\Sigma^* = \bigcup_{\Sigma_0 \subseteq \Sigma} \Sigma_0^*$.

1.2 Высказывания и логические операции

Логика высказываний формализует определённые представления о (реальных) высказываниях и логических операциях.

Определение 1.12. *Высказыванием* называется повествовательное предложение, для которого имеет смысл говорить о его истинности или ложности.

Пример 1.13. Предложение «Лиссабон — столица Испании» является высказыванием.

Определение 1.14. Существуют два *истинностных значения* — «истина» и «ложь». Мы будем обозначать их И и Л, соответственно; считаем 1 и 0 синонимами И и Л.

Некоторые сложные высказывания строятся из более простых с помощью *логических операций*, таких как отрицание «не», конъюнкция «и», дизъюнкция «или», импликация «если …, то …».

Определение 1.15. *Логическая операция* — это такой способ построения сложного высказывания из данных высказываний, при котором истинностное значение сложного высказывания полностью определяется истинностными значениями исходных высказываний.

Пример 1.16. Отрицание является логической операцией. Предложение «Неверно, что Лиссабон — столица Испании» построено из высказывания «Лиссабон — столица Испании» с помощью отрицания.

Замечание 1.17. Употребляемую в естественном языке импликацию «если A, то B» нельзя в полной мере считать логической операцией, поскольку она, среди прочего, указывает и на причинно-следственную связь между высказываниями A и B, то есть не выражается только лишь через истинностные значения высказываний A и B. Более того, высказывание «если A, то B» *полисемично*, то есть может пониматься по-разному в разных контекстах.

В математическом языке используется *материальная импликация*, которая является логической связкой. При этом высказывание «если A, то B» считается ложным в том и *только том* случае, если A истинно и B ложно.

1.3 Синтаксис логики высказываний

1.3.1 Переменные и связки

Пусть задан некоторый алфавит Var символов, называемых *пропозициональными*[1] *переменными*. Пропозициональные переменные будем обозначать буквами P, Q и т. д. (возможно, с индексами). Интуитивно, пропозициональные переменные интерпретируются как высказывания.

Знаки $\neg$, $\wedge$, $\vee$, $\to$ (и аналогичные знаки, которые будут введены позже) называются *пропозициональными связками* или *булевыми связками*. Интуитивно, связки интерпретируются как логические операции (соответственно, как отрицание, конъюнкция, дизъюнкция, импликация).

1.3.2 Формулы

Формулы логики высказываний являются словами в алфавите, состоящем из пропозициональных переменных, пропозициональных связок и скобок: (и). Множество всех формул индуктивно определяется следующим образом.

[1] Propositio (лат.) = предложение.

Определение 1.18. *Множество формул* Fm логики высказываний порождается из множества Var по следующим правилам:

1. Если $P \in \mathrm{Var}$, то P — формула.
2. Если A — формула, то $\neg A$ — формула.
3. Если A и B — формулы, то $(A \wedge B)$, $(A \vee B)$, $(A \to B)$ — формулы.

Другими словами, множество формул есть наименьшее множество, замкнутое относительно этих трёх правил.

Определение 1.19. *Построением* формулы A называем последовательность формул, каждый элемент которой есть либо переменная, либо получается из предыдущих по правилам 2 или 3, и последний элемент которой есть A.

Смысл определения 1.18 состоит в том, что формулами считаются те и только те слова, которые имеют построение. Определение множества объектов как наименьшего множества, замкнутого относительно некоторых правил образования, называется *индуктивным.* Такого рода определения часто используются в алгебре и логике.

Пример 1.20. Последовательность P, Q, $(P \to Q)$, $(Q \wedge (P \to Q))$ есть построение формулы $(Q \wedge (P \to Q))$.

Формулы логики высказываний будем обозначать буквами A, B и т. д. (возможно, с индексами).

Определение 1.21. *Подформулами* формулы A называются все те формулы, которые возникают при некотором построении A. Подформула формулы A, отличная от самой формулы A, называется *собственной подформулой* формулы A.

Замечание 1.22. Не следует путать подформулы с их *вхождениями* в формулу. Одна и та же подформула может иметь несколько вхождений, например подформула P входит три раза в формулу $(P \to (P \wedge P))$.

Предложение 1.23 (однозначность разбора, без доказательства**).** *Каждая пропозициональная формула, не являющаяся переменной, может быть представлена единственным образом как* $\neg A$, $(A \wedge B)$, $(A \vee B)$ *или* $(A \to B)$.

Доказательство этого утверждения основывается на соображениях баланса скобок в формуле. Следствием является, например, тот факт, что множество подформул данной формулы не зависит от её построения.

1.3.3 Сокращённая запись формул

Для удобства записи формул принято использовать некоторые сокращения. С формальной точки зрения, такие сокращения являются приёмами изложения, а не элементами языка логики высказываний. Мы рассмотрим два важных вида сокращений.

А. *Соглашения о скобках.*

Во-первых, можно опустить внешнюю пару скобок. Например, запись $P \to (Q \to P)$ обозначает формулу $(P \to (Q \to P))$.

Во-вторых, если в сокращённой записи рядом находятся две операции $\wedge$, то при отсутствии скобок внутренней считается та, которая находится левее. Другими словами, связка $\wedge$ считается *левоассоциативной*. Например, $P \wedge Q \wedge R$ и $(P \wedge Q) \wedge R$ обозначают одну и ту же формулу (длина этой формулы — 9 символов). Однако в записи $P \wedge (Q \wedge R)$ ни одной скобки опустить нельзя. Связка $\vee$ тоже является левоассоциативной, но связка $\to$ не является ни левоассоциативной, ни правоассоциативной (в этом курсе).

В-третьих, если в сокращённой записи рядом находятся разные связки, то при отсутствии скобок внутренней считается та, которая имеет более высокий приоритет согласно следующему списку, составленному в порядке убывания приоритетов: $\neg$, $\wedge$, $\vee$, $\to$. Иными словами, связки с более высоким приоритетом связывают сильнее.

Разрешается также добавить внешнюю пару скобок. Например, запись $(\neg P)$ обозначает формулу $\neg P$. Добавление скобок пригодится, например, в определении 1.45.

Б. *Введение новых логических связок.*

Логическую связку *эквивалентности* $\leftrightarrow$ часто определяют как сокращение. При этом для любых формул A, B запись $A \leftrightarrow B$ понимается как обозначение для формулы $(A \to B) \wedge (B \to A)$. Аналогичным образом можно ввести и другие логические связки, в частности:

$$\begin{aligned}
\bot &\rightleftharpoons (P_0 \wedge \neg P_0), \quad \text{где } P_0 \text{ — фиксированная переменная;}\\
\top &\rightleftharpoons \neg\bot;\\
\bigwedge_{i=1}^{n} A_i &\rightleftharpoons (A_1 \wedge A_2 \wedge \dots \wedge A_n);\\
\bigvee_{i=1}^{n} A_i &\rightleftharpoons (A_1 \vee A_2 \vee \dots \vee A_n).
\end{aligned}$$

1.3.4 Другие варианты синтаксиса

Помимо стандартного, изложенного выше, способа записи формул логики высказываний существуют и другие варианты. Отметим три важных способа представления формул.

А. *Польская запись.* Логические связки (как унарные, так и бинарные) записываются префиксным образом, например, вместо $(A \wedge B)$ пишем $\wedge AB$; при этом скобки не употребляются. Так, формула $(A \to (B \wedge C))$ может быть записана «по-польски» как $\to A \wedge BC$. (Почему для польской записи имеет место теорема об однозначности разбора?)

Б. *Представление формул деревьями.* С каждой формулой можно однозначно связать бинарное дерево, называемое иногда *деревом разбора*, листья которого помечены пропозициональными переменными, а внутренние вершины – связками. Вершины этого дерева находятся во взаимно-однозначном соответствии со вхождениями подформул в данную формулу (соответетвующей корню дерева).

Интересно отметить, что одна из первых формулировок логики высказваний, данная в XIX веке немецким учёным Г. Фреге, использовала вариант графического изображения деревьев в качестве записи формул. (Напрашивается сравнение с иероглифическим письмом.)

В. *Представление формул ориентированными ациклическими графами.* Если отождествить в дереве разбора формулы вершины, соответствующие вхождениям одной и той же подформулы, то получится структура, называемая ориентированным ациклическим графом (DAG). При таком представлении вершины графа соответствуют подформулам данной формулы, а стрелки соединяют каждые две подформулы, одна из которых является максимальной собственной подформулой другой. Этот способ представления формул является наиболее экономным и распостранённым вариантом представления формул в памяти компьютера.

1.4 Таблицы истинности

Определение 1.24. Обозначим $\mathbb{B} \rightleftharpoons \{\text{И}, \text{Л}\} \rightleftharpoons \{0, 1\}$. Функции $f : \mathbb{B}^n \to \mathbb{B}$ называются *булевыми функциями.*

Определение 1.25. *Оценкой пропозициональных переменных* (или просто *оценкой*) называется произвольная функция $f : \mathrm{Var} \to \mathbb{B}$.

Определение 1.26. *Истинностное значение* (или просто *значение*) формулы при данной оценке f определяется индукцией по построению формулы в соответствии со следующими таблицами.

A	$\neg A$
Л	И
И	Л

A	B	$A \wedge B$	$A \vee B$	$A \to B$
Л	Л	Л	Л	И
Л	И	Л	И	И
И	Л	Л	И	Л
И	И	И	И	И

С формальной точки зрения, оценка $f : \mathrm{Var} \to \mathbb{B}$ продолжается до функции $f : \mathrm{Fm} \to \mathbb{B}$, определённой на множестве всех формул, по следующим правилам.

$$\begin{aligned} f(\neg A) = \text{И} \quad &\iff \quad f(A) = \text{Л}; \\ f(A \wedge B) = \text{И} \quad &\iff \quad f(A) = \text{И и } f(B) = \text{И}; \\ f(A \vee B) = \text{И} \quad &\iff \quad f(A) = \text{И или } f(B) = \text{И}; \\ f(A \to B) = \text{И} \quad &\iff \quad f(A) = \text{Л или } f(B) = \text{И}. \end{aligned}$$

Если $f(A) = \text{И}$, то говорят, что формула A *истинна* при оценке f. Иначе формула A *ложна* при данной оценке.

Специально рассмотрим случай, когда число переменных конечно, то есть $\mathrm{Var} = \{P_1, \ldots, P_n\}$. Оценка f определяется набором своих истинностных значений на переменных $P_1, \ldots, P_n$. Данному набору $\vec{x} = (x_1, \ldots, x_n) \in \mathbb{B}^n$ сопоставим оценку $f_{\vec{x}}$, определяемую таблицей

P_1	P_2	$\ldots$	P_n
x_1	x_2	$\ldots$	x_n

Таким образом, существует взаимно-однозначное соответствие между оценками и наборами из $\mathbb{B}^n$.

Определение 1.27. *Таблицей истинности* (или *истинностной таблицей*) формулы A над переменными $P_1, \ldots, P_n$ называется таблица, указывающая значения формулы A при всех возможных оценках переменных $P_1, \ldots, P_n$. (Существует 2^n таких оценок, каждая из них записывается в отдельной строке. Обычно оценки $f_{\vec{x}}$ упорядочены в соответствии с лексикографическим порядком на наборах $\vec{x}$.)

Пример 1.28.

P_1	P_2	$P_1 \leftrightarrow P_2$
Л	Л	И
Л	И	Л
И	Л	Л
И	И	И

Таким образом, таблица истинности формулы A над n переменными задаёт булеву функцию $\varphi_A : \mathbb{B}^n \to \mathbb{B}$. Функция φ_A определяется равенством

$$\varphi_A(\vec{x}) = f_{\vec{x}}(A),$$

верным для всех наборов $\vec{x} \in \mathbb{B}^n$.

1.5 Функциональная полнота

Всякую ли булеву функцию можно задать некоторой формулой? Ответ даёт следующая теорема.

Теорема 1.29 (о функциональной полноте). *Для любой функции* $\varphi : \mathbb{B}^n \to \mathbb{B}$ *найдётся такая формула* A *от* n *переменных, что* $\varphi = \varphi_A$*. При этом можно считать, что* A *содержит лишь связки* $\neg$ *и* $\vee$*.*

Эта теорема показывает, что известных нам логических операций $\vee$, $\neg$ в принципе достаточно, чтобы определить все возможные логические операции.

Доказательство. Равенство $\varphi = \varphi_A$ означает, что для всех $\vec{x} \in \mathbb{B}^n$

$$\varphi(\vec{x}) = \varphi_A(\vec{x}) = f_{\vec{x}}(A).$$

Для $x \in \mathbb{B}$ положим

$$P^x = \begin{cases} P, & \text{если } x = \text{И}; \\ \neg P, & \text{если } x = \text{Л}. \end{cases}$$

Для произвольного $\vec{x} = (x_1, \ldots, x_n) \in \mathbb{B}^n$ обозначим

$$A_{\vec{x}} \rightleftharpoons \bigwedge_{i=1}^{n} P_i^{x_i}.$$

Легко видеть, что формула $A_{\vec{x}}$ истинна лишь при оценке $f_{\vec{x}}$. Другими словами, для любой оценки f

$$f(A_{\vec{x}}) = \text{И} \iff f = f_{\vec{x}}. \tag{1}$$

Для данной функции φ пусть список $\vec{x}_1, \ldots, \vec{x}_m$ исчерпывает все наборы $\vec{x} \in \mathbb{B}^n$ для которых $\varphi(\vec{x}) = \text{И}$, то есть

$$\varphi(\vec{x}) = \text{И} \iff \exists j\ \vec{x} = \vec{x}_j. \tag{2}$$

Положим теперь

$$A \rightleftharpoons \bigvee_{j=1}^{m} A_{\vec{x}_j},$$

тогда

$$\begin{aligned} f_{\vec{x}}(A) = \text{И} \quad &\Longleftrightarrow \quad \exists j\ f_{\vec{x}}(A_{\vec{x}_j}) = \text{И} \\ &\Longleftrightarrow \quad \exists j\ \vec{x} = \vec{x}_j \quad \text{по } (1) \\ &\Longleftrightarrow \quad \varphi(\vec{x}) = \text{И} \quad \text{по } (2). \end{aligned}$$

Заметим теперь, что конъюнкция выражается через дизъюнкцию и импликацию, поскольку формула $A \wedge B$ равносильна $\neg(\neg A \vee \neg B)$ (см. ниже раздел 1.7). Поэтому, формулы $A_{\vec{x}}$ могут быть переписаны без использования знака $\wedge$. ⊠

1.6 Выполнимые формулы, тавтологии, логическое следование

1.6.1 Выполнимые формулы и тавтологии

Определение 1.30. Пропозициональная формула, истинная хотя бы при одной оценке пропозициональных переменных, называется *выполнимой*. Множество формул Γ назывется *выполнимым*, если существует оценка f, при которой истинны одновременно все формулы из Γ.

Определение 1.31. Пропозициональная формула, истинная при каждой оценке пропозициональных переменных, называется *тавтологией* (*тождественно истинной*).

Важность понятия тавтологии с точки зрения оснований математики (и логики в целом) состоит в том, что они выражают *универсальные законы* логики, верные независимо от содержания составляющих их высказываний. Запись $\vDash A$ выражает тот факт, что A — тавтология.

Определение 1.32. Пропозициональная формула, ложная при каждой оценке пропозициональных переменных, называется *тождественно ложной.*

Предложение 1.33. *Следующие условия равносильны.*

(i) *Формула A тождественно ложна.*

(ii) *Формула A не является выполнимой.*

(iii) *Формула* $\neg A$ — *тавтология.*

Доказательство. Предложение непосредственно следует из определений. $\boxtimes$

1.6.2 Проверка формулы на выполнимость

В приложениях часто встречается задача проверки пропозициональной формулы на выполнимость. Наиболее прямолинейный алгоритм её решения состоит в построении всей таблицы истинности формулы, то есть перебора 2^n всех возможных оценок. Этот алгоритм работает экспоненциальное число шагов от числа переменных исходной формулы. Существуют более изощрённые и несколько более эффективные алгоритмы решения этой задачи, однако все они имеют экспоненциальную нижнюю оценку сложности.

Важной открытой проблемой является вопрос о существовании полиномиального по числу шагов алгоритма решения этой задачи. Поскольку выполнимость пропозициональной формулы является классическим примером так называемой NP-полной задачи, этот вопрос эквивалентен знаменитой проблеме P=NP? — одной из самых важных открытых математических проблем. В настоящее время доминирует гипотеза о том, что такого полиномиального алгоритма не существует.

1.6.3 Логическое следование

Определение 1.34. Пусть Γ — некоторое множество формул логики высказываний и A — формула логики высказываний. Говорят, что формула A *логически следует* (или *семантически следует*) из множества Γ (обозначение $\Gamma \vDash A$), если формула A истинна при каждой оценке пропозициональных переменных, при которой истинны все формулы из Γ.

Пример 1.35. $\{P \vee Q, R, \neg Q\} \vDash P \wedge R$.

Предложение 1.36. (i) A — *тавтология* $\iff \varnothing \vDash A$.

(ii) Γ *выполнимо* $\iff \Gamma \nvDash \bot$.

(iii) $\Gamma \vDash A \iff \Gamma \cup \{\neg A\}$ *не выполнимо.*

Предложение 1.37. $\{B_1, \ldots, B_n\} \vDash A$ *тогда и только тогда, когда формула* $(\bigwedge_{i=1}^{n} B_i) \to A$ *является тавтологией.*

Оба этих предложения непосредственно следуют из определений.

1.7 Равносильные формулы в логике высказываний

Определение 1.38. Формулы A и B называются *равносильными* (*эквивалентными*), обозначение $A \equiv B$, если при каждой оценке пропозициональных переменных значение A совпадает со значением B. Другими словами, если $\varphi_A = \varphi_B$.

Пример 1.39. $P \to Q \equiv \neg Q \to \neg P$; $P \to Q \not\equiv \neg P \to \neg Q$.

Непосредственно из определений вытекают следующие факты.

Утверждение 1.40. (i) *Отношение $\equiv$ рефлексивно, симметрично и транзитивно.*

(ii) *Формулы A и B равносильны тогда и только тогда, когда формула $A \leftrightarrow B$ является тавтологией.*

(iii) *Формула A — тавтология тогда и только тогда, когда $A \equiv \top$.*

Основные равносильности:

$A \wedge B$	$\equiv$	$B \wedge A$	$A \vee B$	$\equiv$	$B \vee A$
$A \wedge (B \wedge C)$	$\equiv$	$(A \wedge B) \wedge C$	$A \vee (B \vee C)$	$\equiv$	$(A \vee B) \vee C$
$A \wedge A$	$\equiv$	A	$A \vee A$	$\equiv$	A
$A \wedge (B \vee C)$	$\equiv$	$(A \wedge B) \vee (A \wedge C)$	$A \vee (B \wedge C)$	$\equiv$	$(A \vee B) \wedge (A \vee C)$
$A \vee (A \wedge B)$	$\equiv$	A	$A \wedge (A \vee B)$	$\equiv$	A
$\neg(A \wedge B)$	$\equiv$	$\neg A \vee \neg B$	$\neg(A \vee B)$	$\equiv$	$\neg A \wedge \neg B$
$\neg\neg A$	$\equiv$	A	$A \to B$	$\equiv$	$\neg A \vee B$

Упражнение 1.41. Равносильны ли формулы $P \to (Q \to R)$ и $(P \to Q) \to R$? Ответ: Нет. Рассмотрим такую оценку g, что $g(P) = g(Q) = g(R) = \text{Л}$.

Упражнение 1.42. Равносильны ли формулы $P \to (Q \to R)$ и $(P \wedge Q) \to R$? Ответ: Да.

Замечание 1.43. В задаче на упрощение формулы необходимо найти равносильную, но более короткую формулу. При этом длина понимается как общее количество всех вхождений символов в формулу.

Упражнение 1.44. *Упростить формулы:*

(i) $(P \leftrightarrow Q) \to P$. *Ответ:* $P \vee Q$.

(ii) $\neg P \to \neg Q$. *Ответ:* $Q \to P$.

(iii) $(P \vee Q) \wedge (P \vee R) \wedge (Q \vee R \vee \neg P)$. *Ответ:* $(P \vee (Q \wedge R)) \wedge (Q \vee R)$.

1.8 Правила подстановки и замены подформулы на эквивалентную

Для доказательства равносильности формул, помимо основных равносильностей, перечисленных в таблице, и правил, соответствующих рефлексивности, симметричности и транзитивности отношения $\equiv$, мы пользуемся правилами подстановки и замены на подформулы на эквивалентную.

Определение 1.45. Если C и D — формулы, а P — пропозициональная переменная, то через $C[P/D]$ обозначим результат подстановки формулы D вместо P в формулу C.

Формальное определение даётся с помощью индукции по построению формулы C.

$$
\begin{aligned}
&P[P/D] \rightleftharpoons D,\\
&Q[P/D] \rightleftharpoons Q, \text{ если } Q \text{ — переменная, отличная от } P,\\
&(\neg A)[P/D] \rightleftharpoons \neg(A[P/D]),\\
&(A \wedge B)[P/D] \rightleftharpoons (A[P/D] \wedge B[P/D]),\\
&(A \vee B)[P/D] \rightleftharpoons (A[P/D] \vee B[P/D]),\\
&(A \to B)[P/D] \rightleftharpoons (A[P/D] \to B[P/D]).
\end{aligned}
$$

Пример 1.46. Пусть $C = (P_1 \to P_2) \to P_2$ и $D = P_3 \to P_2$. Тогда

$$C[P_2/D] = (P_1 \to (P_3 \to P_2)) \to (P_3 \to P_2).$$

Теорема 1.47 (о подстановке). *Если A — тавтология, B — произвольная формула, а P — пропозициональная переменная, то $A[P/B]$ — тавтология.*

Доказательство. Рассмотрим произвольную оценку g. Обозначим через g' оценку, полученную из g присвоением переменной P значения $g(B)$. Индукцией по построению C можно доказать, что $g(C[P/B]) = g'(C)$ для любой формулы C. Положим $C = A$. Так как формула A истинна при оценке g', то формула $A[P/B]$ истинна при оценке g. ⊠

Пример 1.48. Для любой формулы B формула $B \vee \neg B$ является тавтологией. Например, формула $(P_3 \leftrightarrow P_1) \vee \neg(P_3 \leftrightarrow P_1)$ является тавтологией.

Теорема 1.49. *Пусть A, B, C — формулы, а P — пропозициональная переменная. Если $A \equiv B$, то $A[P/C] \equiv B[P/C]$.*

Доказательство. Пусть $A \equiv B$. В силу 1.40 $A \leftrightarrow B$ — тавтология. По теореме 1.47 $(A \leftrightarrow B)[P/C]$ — тавтология. Из определений следует, что $(A \leftrightarrow B)[P/C]$ совпадает с $A[P/C] \leftrightarrow B[P/C]$. В силу 1.40 $A[P/C] \equiv B[P/C]$. ⊠

Пример 1.50. Пусть $A = (P_1 \to P_2) \to P_2$, $B = P_1 \vee P_2$, $C = P_3 \to P_2$. Так как $(P_1 \to P_2) \to P_2 \equiv P_1 \vee P_2$, то $(P_1 \to (P_3 \to P_2)) \to (P_3 \to P_2) \equiv P_1 \vee (P_3 \to P_2)$.

Лемма 1.51. *Если* $A \equiv B$, *то* $\neg A \equiv \neg B$. *Если* $A_1 \equiv B_1$ *и* $A_2 \equiv B_2$, *то* $A_1 \wedge A_2 \equiv B_1 \wedge B_2$, $A_1 \vee A_2 \equiv B_1 \vee B_2$, $A_1 \to A_2 \equiv B_1 \to B_2$.

Теорема 1.52 (о замене подформулы на эквивалентную). *Пусть* A, B, C *— формулы, а* P *— пропозициональная переменная. Если* $A \equiv B$, *то* $C[P/A] \equiv C[P/B]$.

Доказательство. Теорема доказывается индукцией по построению формулы C. ⊠

Пример 1.53. Пусть $A = Q \vee Q$, $B = Q$, $C = P \wedge R$. Так как $Q \vee Q \equiv Q$, то $(Q \vee Q) \wedge R \equiv Q \wedge R$.

Пример 1.54. Существуют ли такие выполнимые формулы A и B, что формула $A[P_1/B]$ не является выполнимой? Ответ: Да. Например, $A = \neg P_1$, $B = P_2 \vee \neg P_2$.