

Дискретная математика

Глава 2. Основы математической логики

А. В. Пастор

26.09.2023

Дополнительные материалы по логике

1. Э. Мендельсон, *Введение в математическую логику*. М.: Наука, 1976.

Слайды по дискретной математике будут публиковаться по адресу

<https://logic.pdmi.ras.ru/~pastor/ITMO/2022-23/>

Логические высказывания

- **Высказыванием** называется утверждение (утвердительное повествовательное предложение), про которое можно однозначно сказать, истинно оно или ложно.
 - ▶ То есть истинность или ложность логического высказывания не должна зависеть от каких-либо параметров (значений переменных и т. п.).
 - ▶ Например, " $a > b$ " — это уже не высказывание, поскольку его истинность или ложность зависит от значений a и b .
 - ▶ В то же время, " $3 > 2$ " — это высказывание (истинное).
 - ▶ "Все нечетные числа простые" — это тоже высказывание (ложное).
- Из простых высказываний можно составлять более сложные.
 - ▶ Например, если есть высказывание A , то можно составить высказывание "не A ";
 - ▶ а из высказывания A и высказывания B можно составить высказывания
 - " A и B ",
 - " A или B " (здесь **или** не исключаящее!),
 - "если A , то B ".
- Запишем это более формально.

Булевы функции

- Пусть нам даны высказывания A_1, A_2, \dots, A_n . Мы хотим составить из них новое высказывание. При этом истинность или ложность нового высказывания должна зависеть только от истинности или ложности A_1, A_2, \dots, A_n , но не от того, что именно это за высказывания.
- То есть должна быть функциональная зависимость истинности/ложности нового высказывания от истинности/ложности исходных.

Определение

Булевой функцией от n переменных называется отображение $f: \{0, 1\}^n \rightarrow \{0, 1\}$.

- Всего есть 2^{2^n} булевых функций от n переменных.
- Считается, что ложному высказыванию соответствует значение 0, а истинному — значение 1. Тем самым, любое высказывание, которое можно составить из данных n высказываний, можно выразить булевой функцией от n переменных.

Таблицы истинности

- Булевы функции можно задавать при помощи *таблиц истинности*.
- Таблица истинности — это таблица с 2^n строками (где n — число переменных), первые n столбцов которой соответствуют значениям переменных, а $(n + 1)$ -й столбец содержит значения функции.
- Каждая строка соответствует одной из 2^n возможных комбинаций значений аргументов: соответствующая строка из нулей и единиц записывается в первые n клеток данной строки, а в $(n + 1)$ -й клетке записывается значение функции при данных значениях аргументов (оно также может быть равно либо нулю, либо единице).

Примеры

Приведем таблицы истинности для булевых функций, соответствующих упоминавшимся ранее комбинациям.

1. **Отрицание** (“не x ”, обозначение: $\neg x$ или \bar{x}).

x	$\neg x$
0	1
1	0

Таблицы истинности: примеры

2. **Конъюнкция** (“ x и y ”, обозначение: $x \& y$ или $x \wedge y$)

x	y	$x \& y$
0	0	0
0	1	0
1	0	0
1	1	1

3. **Дизъюнкция** (“ x или y ”, обозначение: $x \vee y$)

x	y	$x \vee y$
0	0	0
0	1	1
1	0	1
1	1	1

4. **Импликация** (“если x , то y ”, обозначение: $x \supset y$, $x \rightarrow y$ или $x \Rightarrow y$)

x	y	$x \supset y$
0	0	1
0	1	1
1	0	0
1	1	1

Булевы функции двух аргументов

- Какие еще бывают булевы функции от двух переменных? Всего их 16.

x	y	0	$x \& y$	$x \& \neg y$	x	$\neg x \& y$	y	$x \oplus y$	$x \vee y$
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

x	y	$x \downarrow y$	$x \equiv y$	$\neg y$	$y \supset x$	$\neg x$	$x \supset y$	$x y$	1
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

- $x \equiv y$ — эквивалентность;
- $x \oplus y$ — сложение по модулю 2;
- $x \downarrow y$ — стрелка Пирса;
- $x | y$ — штрих Шеффера.

Язык исчисления высказываний

- **Алфавит исчисления высказываний** состоит из следующих символов:
 1. **пропозициональные переменные**: как правило, обозначаются латинскими буквами, возможно, с индексами;
 2. **пропозициональные связки**: \neg , $\&$, \vee , \supset ;
 3. **скобки**: $(,)$.
- **Пропозициональной формулой** называется последовательность символов в указанном выше алфавите, которая строится по следующим принципам:
 1. любая переменная — формула;
 2. если A — формула, то $\neg A$ — формула;
 3. если A, B — формулы, то $(A \& B)$, $(A \vee B)$, $(A \supset B)$ — формулы.

Пример

$$((\neg x \vee \neg \neg y) \supset z)$$

Замечание

Иногда в качестве связок используют и другие знаки логических операций.

Например, \equiv , \oplus , \downarrow , $|$.

Эквивалентность формул

- Каждой пропозициональной формуле соответствует булева функция.
- Однако, это соответствие — не биекция. Например, формулам $(\neg x \vee y)$ и $(x \supset y)$ соответствует одна и та же булева функция.

Определение

- Формулы A и B называются *эквивалентными* (или *равнозначными*), если им соответствует одна и та же булева функция.
 - ▶ То есть, если формулы A и B принимают одинаковые значения при любых значениях входящих в них переменных.
- Обозначение: $A \sim B$.

Примеры

1. $\neg\neg x \sim x$;
2. $\neg(x \& y) \sim (\neg x \vee \neg y)$;
3. $\neg(x \vee y) \sim (\neg x \& \neg y)$;
4. $(x \supset y) \sim (\neg x \vee y)$;
5. $\neg(x \supset y) \sim (x \& \neg y)$;
6. $(x \& (y \vee z)) \sim ((x \& y) \vee (x \& z))$;
7. $(x \vee (y \& z)) \sim ((x \vee y) \& (x \vee z))$.

Тавтологии

Определение

- Формула A называется *тавтологией* (или *тождественной истиной*), если при любых значениях переменных принимает значение 1 (т. е. если $A \sim 1$).
- Формула A называется *выполнимой*, если существуют такие значения переменных, при которых A принимает значение 1.
- Формула A называется *противоречием* (или *невыполнимой*), если при любых значениях переменных принимает значение 0 (т. е. если $A \sim 0$).

Примеры

1. $(x \vee \neg x)$ — тавтология (*закон исключенного третьего*);
2. $(x \& \neg x)$ — противоречие;
3. $\neg(x \supset y)$ — выполнимая формула (истинна при $x = 1, y = 0$).

Замечание

Формулы A и B эквивалентны тогда и только тогда, когда формула $(A \equiv B)$ — тавтология.

Нормальные формы

- Даны n пропозициональных переменных x_1, \dots, x_n .
- *Литералом* называется выражение вида x_i , либо $\neg x_i$ (т. е. переменная, либо ее отрицание).
- Выражение вида $(L_1 \vee \dots \vee L_m)$, где L_1, \dots, L_m — литералы, называется *простым дизъюнктом*, а выражение $(L_1 \& \dots \& L_m)$ — *простым конъюнктом*.

Определение

- *Конъюнктивная нормальная форма* (КНФ) — это пропозициональная формула вида $(C_1 \& C_2 \& \dots \& C_k)$, где C_i — простые дизъюнкты.
- *Дизъюнктивная нормальная форма* (ДНФ) — это пропозициональная формула вида $(C_1 \vee C_2 \vee \dots \vee C_k)$, где C_i — простые конъюнкты.
- Подформулы C_i , как в случае КНФ, так и в случае ДНФ, называют также *клозами*.

Примеры

- Формула $(x \vee y \vee \neg z) \& (\neg y \vee z) \& (\neg x \vee \neg z)$ — КНФ;
- формула $(x \& y \& \neg z) \vee (\neg y \& z) \vee (\neg x \& \neg z)$ — ДНФ.

Совершенные формы

Определение

Конъюнктивная или дизъюнктивная нормальная форма называется *совершенной* (СКНФ или СДНФ, соответственно), если выполняются следующие условия:

1. каждая переменная присутствует в каждом клозе ровно один раз;
2. все клозы различны (т. е. нет повторяющихся скобок);
3. в каждом клозе литералы упорядочены по возрастанию индексов (или по алфавиту, если переменные — различные латинские буквы);
4. клозы упорядочены лексикографически (мы считаем, что для любой переменной x_i литерал $\neg x_i$ младше литерала x_i ; любые два клоза упорядочиваются по первому несовпадающему литералу).

Примеры

- Формула $(\neg x \vee y \vee \neg z) \& (\neg x \vee y \vee z) \& (x \vee \neg y \vee \neg z)$ — СКНФ;
- формула $(\neg x \& y \& \neg z) \vee (\neg x \& y \& z) \vee (x \& \neg y \& \neg z)$ — СДНФ.

Представление булевой функции в виде СКНФ и СДНФ

Теорема

Каждая булева функция единственным образом представляется как в виде СКНФ, так и в виде СДНФ.

Доказательство. “ \exists ”: Пусть $f(x_1, \dots, x_n)$ — булева функция.

- Рассмотрим таблицу истинности функции f .
- Выберем из этой таблицы те строки, в которых получается значение 1.
- Каждой строке $(a_1, \dots, a_n) \in \{0, 1\}^n$ можно поставить в соответствие простой конъюнкт $(L_1 \& \dots \& L_n)$, где $L_i = \begin{cases} x_i, & a_i = 1 \\ \neg x_i, & a_i = 0. \end{cases}$
 - ▶ Этот конъюнкт принимает значение 1 тогда и только тогда, когда $x_1 = a_1, \dots, x_n = a_n$.
- Дизъюнкция всех конъюнктов, соответствующих выбранным строкам, будет СДНФ, соответствующей функции f .
- Аналогично, к СДНФ можно привести функцию $\neg f$. Тогда, взяв отрицание полученной формулы, найдем СКНФ, соответствующую f .

Полные системы булевых функций

“!”: СДНФ, соответствующая функции f , единственна, поскольку входящие в нее простые конъюнкты однозначно определяются ее таблицей истинности.

- СКНФ, соответствующая функции f , единственна, поскольку ее отрицание — это СДНФ, соответствующая $\neg f$, а она единственна. □

Определение

Множество \mathcal{F} булевых функций называется *полной системой*, если любую булеву функцию можно выразить через функции из \mathcal{F} при помощи операции композиции.

- Другими словами, любая булева функция должна задаваться пропозициональной формулой, в которой связки соответствуют функциям из \mathcal{F} .

Следствие

Система $\{\&, \vee, \neg\}$ — полная.

Приведение к СКНФ и СДНФ

- Пусть булева функция f задана некоторой пропозициональной формулой.
- Как видно из доказательства теоремы, ее можно привести к СКНФ и СДНФ, при помощи ее таблицы истинности.
- Однако, часто бывает удобнее привести ее к СКНФ и СДНФ эквивалентными преобразованиями.
- Алгоритм приведения пропозициональной формулы F к СКНФ и СДНФ включает в себя следующие шаги.
 1. **Элиминация связок.** Если в формуле F используются связки, отличные от $\&$, \vee , \neg , их нужно заменить на эквивалентные им формулы, использующие только $\&$, \vee , \neg .
 2. **Протаскивание отрицаний.** Многократно применяем эквивалентности $\neg\neg A \sim A$, $\neg(A \& B) \sim (\neg A \vee \neg B)$, $\neg(A \vee B) \sim (\neg A \& \neg B)$ (где A, B — произвольные подформулы) до тех пор, пока в формуле есть отрицания, применяемые к подформулам, отличным от одной переменной. В результате отрицания в формуле будут присутствовать только непосредственно перед переменными.

Приведение к СКНФ и СДНФ

3. **Раскрытие скобок.** Для приведения к СКНФ и к СДНФ скобки нужно раскрывать по-разному.

- Для получения СКНФ нужно использовать эквивалентность $(A \vee (B \& C)) \sim ((B \& C) \vee A) \sim ((A \vee B) \& (A \vee C))$ (где A, B, C — произвольные подформулы). Такие замены производятся до тех пор, пока в формуле есть дизъюнкции, применяемые к подформулам, включающим операцию конъюнкции.
- Для получения СДНФ нужно действовать аналогично, но используя эквивалентность $(A \& (B \vee C)) \sim ((B \vee C) \& A) \sim ((A \& B) \vee (A \& C))$.

По итогам этого шага получится КНФ или ДНФ соответственно, но она может не быть совершенной.

4. **Удаление повторяющихся переменных.**

- Если в каком-либо клезе есть несколько одинаковых литералов, оставляем только один из них.
- Если же в клезе есть разноименные литералы от одной переменной (например, x и $\neg x$), то нужно удалить весь клез.

Приведение к СКНФ и СДНФ

5. Расщепление переменных.

Если кюз C не содержит переменной z , заменяем его на

– $(C \vee z) \& (C \vee \neg z)$ (в случае СКНФ);

– $(C \& z) \vee (C \& \neg z)$ (в случае СДНФ).

6. Удаление повторяющихся кюз.

Если кюз C встречается несколько раз, оставляем лишь один его экземпляр.

7. Сортировка.

В каждом кюзе упорядочиваем литералы по алфавиту (или по номерам индексов); кюзы упорядочиваем лексикографически.

Аксиомы и правила вывода

- Как доказать, что пропозициональная формула является тавтологией?
- Можно написать ее таблицу истинности. Или привести ее к СДНФ.
- Но есть и другой способ: можно вывести данную формулу из аксиом. То есть доказать ее в рамках некоторой *формальной аксиоматической теории*.
- Каждая формальная теория должна включать в себя множество формул, называемых *аксиомами* и конечное множество *правил вывода* (отношений между формулами, позволяющих из некоторых формул выводить другие).

Примеры правил вывода

1. *Modus ponens* (MP) $\frac{A, (A \supset B)}{B}$ это означает, что
из формул A и $(A \supset B)$ мы можем вывести формулу B .

2. *Правило резолюции* $\frac{(x \vee A), (\neg x \vee B)}{(A \vee B)}$,

где x — переменная и A, B — формулы (возможно, пустые).

Формула $(A \vee B)$ называется *резольвентой*.

Пример формальной аксиоматической теории

Формальная аксиоматическая теория \mathcal{L} включает в себя три схемы аксиом и одно правило вывода.

Схемы аксиом:

$$A_1: (A \supset (B \supset A));$$

$$A_2: ((A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C)));$$

$$A_3: ((\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B)).$$

Аксиомой считается любая формула, получаемая из формул A_1 – A_3 подстановкой вместо A , B и C любых формул. Подстановка, заменяющая все вхождения переменной A на формулу F , обозначается так: $[F/A]$.

Правило вывода: Modus ponens (MP)

Определение

- Формула B *выводима* в теории \mathcal{L} из формул A_1, A_2, \dots, A_n , если существует последовательность формул $A_1, A_2, \dots, A_n, F_1, \dots, F_k, B$, в которой каждая формула, начиная с F_1 , либо является аксиомой, либо получается из предыдущих формул при помощи правила MP.
- Обозначение: $A_1, A_2, \dots, A_n \vdash_{\mathcal{L}} B$.

Вывод в формальной аксиоматической теории \mathcal{L}

Определение

- Последовательность $A_1, A_2, \dots, A_n, F_1, \dots, F_k, B$ называется **выводом** формулы B из формул A_1, A_2, \dots, A_n .
- Запись $A_1, A_2, \dots, A_n \vdash_{\mathcal{L}} B$ называется **секвенцией**.
- Если список формул A_1, A_2, \dots, A_n пуст, то говорят, что формула B **выводима** в теории \mathcal{L} (обозначение: $\vdash_{\mathcal{L}} B$).

Пример

Выведем в теории \mathcal{L} формулу $(A \supset A)$.

- | | |
|--|-------------------------------|
| 1. $((A \supset ((A \supset A) \supset A)) \supset ((A \supset (A \supset A)) \supset (A \supset A)))$ | $A_2; [(A \supset A)/B, A/C]$ |
| 2. $(A \supset ((A \supset A) \supset A))$ | $A_1; [(A \supset A)/B]$ |
| 3. $((A \supset (A \supset A)) \supset (A \supset A))$ | MP 2, 1 |
| 4. $(A \supset (A \supset A))$ | $A_1; [A/B]$ |
| 5. $(A \supset A)$ | MP 4, 3 |

• Алфавит

1. предметные константы: a_1, a_2, a_3, \dots ;
2. предметные переменные: x_1, x_2, x_3, \dots ;
3. функциональные символы: $f_1^{m_1}, f_2^{m_2}, f_3^{m_3}, \dots$;
 - в качестве верхнего индекса указывается натуральное число, называемое *местностью* или *арностью* функционального символа;
 - т. е. f_i^m — *m -местный функциональный символ*, ему будет соответствовать функция от m аргументов;
 - предметную константу можно рассматривать как 0-местный функциональный символ;
4. предикатные символы: $P_1^{n_1}, P_2^{n_2}, P_3^{n_3}, \dots$;
 - в качестве верхнего индекса указывается натуральное число, называемое *местностью* или *арностью* предикатного символа;
 - т. е. P_i^n — *n -местный предикатный символ*, ему будет соответствовать n -местное отношение;
5. связки: $\neg, \&, \vee, \supset$;
6. кванторы: \forall, \exists ;
7. скобки: $(,)$.

Язык исчисления предикатов: термы

Замечание

- В принципе, обозначения для предметных констант, переменных, функциональных и предикатных символов могут быть и другими. Но они должны быть заранее определены.
- Список всех используемых предметных констант, функциональных и предикатных символов (с указанием их местности) называется *сигнатурой*.

Термы

- *Термом* называется последовательность символов в указанном выше алфавите, которая строится по следующим принципам:
 1. любая предметная константа и любая предметная переменная — терм;
 2. если f_i^m — m -местный функциональный символ и t_1, \dots, t_m — термы, то $f_i^m(t_1, \dots, t_m)$ — терм;
 3. выражение является термом только в том случае, если это следует из правил 1 и 2.

Язык исчисления предикатов: формулы

- *Элементарной формулой* называется выражение вида $P_j^n(t_1, \dots, t_n)$, где P_j^n — n -местный предикатный символ и t_1, \dots, t_n — термы.
 - ▶ элементарные формулы также называют *атомарными формулами* или просто *атомами*.
- *Формулой* исчисления предикатов называется последовательность символов в указанном выше алфавите, которая строится по следующим принципам:
 1. любая элементарная формула — формула;
 2. если A — формула, то $\neg A$ — формула;
 3. если A, B — формулы, то $(A \& B)$, $(A \vee B)$, $(A \supset B)$ — формулы;
 4. если A — формула и x — предметная переменная, то $\forall x A$ и $\exists x A$ — формулы;
 5. выражение является формулой только в том случае, если это следует из правил 1-4.
- В формулах вида $\forall x A$ и $\exists x A$ выражение A называется *областью действия* квантора $\forall x$ или $\exists x$, соответственно.

Свободные и связанные вхождения переменных

- Вхождение переменной x в формулу F называется *связанным*, если x является переменной входящего в эту формулу квантора $\forall x$ или $\exists x$, либо находится в области действия входящего в эту формулу квантора $\forall x$ или $\exists x$.
- В противном случае, вхождение переменной x в данную формулу называется *свободным*.
- Одна и та же переменная может иметь свободные и связанные вхождения в одну и ту же формулу.

Пример

В формуле $(f(x) = 0 \supset \exists x (g(x, y) = 0))$ первое вхождение переменной x свободно, второе и третье — связаны. Единственное вхождение переменной y свободно.

- Переменная x называется *свободной переменной* формулы F , если в F есть свободное вхождение переменной x . Аналогично, x называется *связанной переменной* формулы F , если в F есть связанное вхождение переменной x .
- Переменная может быть одновременно свободной и связанной в одной и той же формуле.

Свободные и связанные переменные

- Если в формуле нет свободных переменных, она называется *замкнутой*.
- Если свободные вхождения переменных в формуле есть, то можно подставить вместо них какие-либо термы и получить новую формулу.
 - ▶ Пусть F — формула, x_1, \dots, x_n — переменные и t_1, \dots, t_n — термы. Тогда через $F(t_1, \dots, t_n)$ обозначается *результат подстановки термов t_1, \dots, t_n в F вместо всех свободных вхождений переменных x_1, \dots, x_n* .
 - ▶ Также результат подстановки терма t вместо всех свободных вхождений переменной x в формулу F обозначают $[F]_t^x$.
- Терм t называется *свободным для переменной x* в формуле F , если никакое свободное вхождение x в F не находится в области действия никакого квантора $\forall y$ или $\exists y$, где y — переменная, входящая в t .

Пример

Терм $x + y$ свободен для переменной x , но не свободен для переменной y в формуле $(f(x) = 0 \supset \exists x (g(x, y) = 0))$.

Интерпретации

- Выбираем множество D — *область интерпретации*.
- Каждой предметной константе a_i ставим в соответствие элемент $\alpha_i \in D$.
- Каждому n -местному функциональному символу f_i^n ставим в соответствие n -местную операцию на D (т. е. отображение $f_i: D^n \rightarrow D$).
- Каждому n -местному предикатному символу P_i^n ставим в соответствие n -местное отношение на D (т. е. подмножество $P_i \subset D^n$).
- После этого, каждой замкнутой формуле будет соответствовать некоторое высказывание (оно истинно или ложно).
- Каждой незамкнутой формуле будет соответствовать отношение на множестве D (k -местное отношение, где k — число свободных переменных).

Примеры

1. Формула $\exists c (a = b \cdot c)$ при $D = \mathbb{N}$ задает отношение делимости.
 - Но при $D = \mathbb{Q}_+$ получаем универсальное отношение (все пары чисел).
2. Формула $\forall a \exists b (a = b \cdot b)$ при $D = \mathbb{R}$ задает ложное высказывание.
А при $D = \mathbb{C}$ — истинное.

Выполнимость и общезначимость

- Формула F называется *истинной в данной интерпретации*, если соответствующее ей отношение выполняется для всех наборов значений переменных.
 - ▶ Это эквивалентно тому, что *замыкание* формулы F (т. е. формула $\forall x_1 \forall x_2 \dots \forall x_k F$, где x_1, x_2, \dots, x_k — свободные переменные формулы F) в данной интерпретации задает истинное высказывание.
- Формула F называется *(логически) общезначимой*, если она истинна в любой интерпретации.
- Формула F называется *выполнимой в данной интерпретации*, если соответствующее ей отношение выполняется хотя бы для одного набора значений переменных.
 - ▶ То есть формула $\exists x_1 \exists x_2 \dots \exists x_k F$, где x_1, x_2, \dots, x_k — свободные переменные формулы F , в данной интерпретации задает истинное высказывание.
- Формула F называется *выполнимой*, если она выполнима в какой-либо интерпретации.