Математическая логика и теория алгоритмов
Лекция 2. Алгебра логики

Куценко Дмитрий Александрович

Белгородский государственный технологический университет
имени В. Г. Шухова

Институт информационных технологий и управляющих систем

Кафедра программного обеспечения вычислительной техники
и автоматизированных систем

18 февраля 2013 г.

Алгебра логики

В математической логике используется терминология, принятая в математике.

Алгебра логики

В математической логике используется терминология, принятая в математике.

Вводятся следующие понятия:

Алгебра логики

В математической логике используется терминология, принятая в математике.

Вводятся следующие понятия:

- пропозициональная переменная (см. предыдущую лекцию),

Алгебра логики

В математической логике используется терминология, принятая в математике.

Вводятся следующие понятия:

- пропозициональная переменная (см. предыдущую лекцию),
- логическая операция,

Алгебра логики

В математической логике используется терминология, принятая в математике.

Вводятся следующие понятия:

- пропозициональная переменная (см. предыдущую лекцию),
- логическая операция,
- алгебра логики,

Алгебра логики

В математической логике используется терминология, принятая в математике.

Вводятся следующие понятия:

- пропозициональная переменная (см. предыдущую лекцию),
- логическая операция,
- алгебра логики,
- формула алгебры логики.

Логические операции и алгебра логики

Понятию «пропозициональная связка» соответствует понятие «логическая операция».

Логические операции и алгебра логики

Понятию «пропозициональная связка» соответствует понятие «логическая операция».

Множество пропозициональных переменных $\mathscr{P} = \{A, B, C, \ldots\}$ с заданными над ним логическими операциями $\mathscr{F} = \{\overline{}, \&, \vee, \rightarrow, \leftrightarrow\}$ формируют алгебру логики:

Логические операции и алгебра логики

Понятию «пропозициональная связка» соответствует понятие «логическая операция».

Множество пропозициональных переменных $\mathscr{P} = \{A, B, C, \ldots\}$ с заданными над ним логическими операциями $\mathscr{F} = \{\overline{}, \&, \vee, \rightarrow, \leftrightarrow\}$ формируют алгебру логики:

$$\mathscr{A}_{\text{Л}} = \langle \mathscr{P}, \mathscr{F} \rangle.$$

Логические операции и алгебра логики

Понятию «пропозициональная связка» соответствует понятие «логическая операция».

Множество пропозициональных переменных $\mathscr{P} = \{A, B, C, \ldots\}$ с заданными над ним логическими операциями $\mathscr{F} = \{\overline{\ \ }, \&, \vee, \to, \leftrightarrow\}$ формируют алгебру логики:

$$\mathscr{A}_{\text{Л}} = \langle \mathscr{P}, \mathscr{F} \rangle.$$

Алгебру логики называют также алгеброй высказываний.

Формулы алгебра логики

Всякому сложному высказыванию соответствует формула алгебры логики.

Формулы алгебра логики

Всякому сложному высказыванию соответствует формула алгебры логики.
Формулы будем обозначать заглавными готическими буквами $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}, \ldots$, чтобы отличать их от пропозициональных переменных.

Формулы алгебра логики

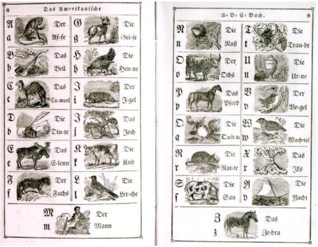
Всякому сложному высказыванию соответствует формула алгебры логики.
Формулы будем обозначать заглавными готическими буквами $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}, \ldots$, чтобы отличать их от пропозициональных переменных.

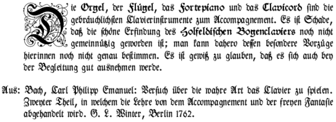
Фрактура

Фрактура — разновидность готического письма, широко употреблявшаяся в Германии до начала XX века. Запрещена во время Третьего рейха в 1941 году.

Фрактура

Фрактура — разновидность готического письма, широко употреблявшаяся в Германии до начала XX века. Запрещена во время Третьего рейха в 1941 году.




Die Orgel, der Flügel, das Fortepiano und das Clavicord sind die gebräuchlichsten Clavierinstrumente zum Accompagnement. Es ist Schade, daß die schöne Erfindung des Holfeldischen Bogenclaviers noch nicht gemeinnützig geworden ist; man kann dahero dessen besondere Vorzüge hierinnen noch nicht genau bestimmen. Es ist gewiß zu glauben, daß es sich auch bey der Begleitung gut ausnehmen werde.

Aus: Bach, Carl Philipp Emanuel: Versuch über die wahre Art das Clavier zu spielen. Zweyter Theil, in welchem die Lehre von dem Accompagnement und der freyen Fantasie abgehandelt wird. G. L. Winter, Berlin 1762.

Немецкий алфавит

НЕМЕЦКИЙ АЛФАВИТ.

§ 1. *Немецкий алфавит* состоит из 26 букв:

Латинские буквы		Готические буквы		Название буквы	Латинские буквы		Готические буквы		Название буквы
Печатные	Рукописные	Печатные	Рукописные		Печатные	Рукописные	Печатные	Рукописные	
A a	𝒜 𝒶	𝔄 𝔞	𝒜 𝒶	а	N n	𝒩 𝓃	𝔑 𝔫	𝒩 𝓃	эн
B b	ℬ 𝒷	𝔅 𝔟	ℬ 𝒷	бэ	O o	𝒪 ℴ	𝔒 𝔬	𝒪 ℴ	о
C c	𝒞 𝒸	ℭ 𝔠	𝒞 𝒸	цэ	P p	𝒫 𝓅	𝔓 𝔭	𝒫 𝓅	пэ
D d	𝒟 𝒹	𝔇 𝔡	𝒟 𝒹	дэ	Q q	𝒬 𝓆	𝔔 𝔮	𝒬 𝓆	ку
E e	ℰ ℯ	𝔈 𝔢	ℰ ℯ	э	R r	ℛ 𝓇	ℜ 𝔯	ℛ 𝓇	эр
F f	ℱ 𝒻	𝔉 𝔣	ℱ 𝒻	эф	S s	𝒮 𝓈	𝔖 ſ 𝔰	𝒮 ſ 𝓈	эс
G g	𝒢 ℊ	𝔊 𝔤	𝒢 ℊ	гэ	T t	𝒯 𝓉	𝔗 𝔱	𝒯 𝓉	тэ
H h	ℋ 𝒽	ℌ 𝔥	ℋ 𝒽	ха	U u	𝒰 𝓊	𝔘 𝔲	𝒰 𝓊	у
I i	ℐ 𝒾	ℑ 𝔦	ℐ 𝒾	и	V v	𝒱 𝓋	𝔙 𝔳	𝒱 𝓋	фау
J j	𝒥 𝒿	𝔍 𝔧	𝒥 𝒿	иот	W w	𝒲 𝓌	𝔚 𝔴	𝒲 𝓌	вэ
K k	𝒦 𝓀	𝔎 𝔨	𝒦 𝓀	ка	X x	𝒳 𝓍	𝔛 𝔵	𝒳 𝓍	икс
L l	ℒ 𝓁	𝔏 𝔩	ℒ 𝓁	эль	Y y	𝒴 𝓎	𝔜 𝔶	𝒴 𝓎	ипсилон
M m	ℳ 𝓂	𝔐 𝔪	ℳ 𝓂	эм	Z z	𝒵 𝓏	ℨ 𝔷	𝒵 𝓏	цэт

В готическом алфавите кроме этих знаков употребляются слитные буквы 𝔠𝔥 (𝔠 + 𝔥), 𝔠𝔨 (𝔠 + 𝔨), ß (ſ + 𝔷), 𝔱𝔷 (𝔱 + 𝔷)

Рукописные заглавные готические буквы

Формальное определение формулы

Формула алгебры логики формально определяется следующим образом:

Формальное определение формулы

Формула алгебры логики формально определяется следующим образом:

1. Пропозициональная переменная — (атомарная) формула.

Формальное определение формулы

Формула алгебры логики формально определяется следующим образом:

1. Пропозициональная переменная — (атомарная) формула.
2. Если $\mathfrak{A}$ и $\mathfrak{B}$ — формулы, то $(\mathfrak{A} \,\&\, \mathfrak{B})$, $(\mathfrak{A} \vee \mathfrak{B})$, $(\mathfrak{A} \to \mathfrak{B})$, $(\mathfrak{A} \leftrightarrow \mathfrak{B})$ — формулы.

Формальное определение формулы

Формула алгебры логики формально определяется следующим образом:

1. Пропозициональная переменная — (атомарная) формула.
2. Если $\mathfrak{A}$ и $\mathfrak{B}$ — формулы, то $(\mathfrak{A} \mathbin{\&} \mathfrak{B})$, $(\mathfrak{A} \vee \mathfrak{B})$, $(\mathfrak{A} \to \mathfrak{B})$, $(\mathfrak{A} \leftrightarrow \mathfrak{B})$ — формулы.
3. Если $\mathfrak{A}$ — формула, то $(\overline{\mathfrak{A}})$ — формула.

Формальное определение формулы

Формула алгебры логики формально определяется следующим образом:

1. Пропозициональная переменная — (атомарная) формула.
2. Если $\mathfrak{A}$ и $\mathfrak{B}$ — формулы, то $(\mathfrak{A} \,\&\, \mathfrak{B})$, $(\mathfrak{A} \vee \mathfrak{B})$, $(\mathfrak{A} \rightarrow \mathfrak{B})$, $(\mathfrak{A} \leftrightarrow \mathfrak{B})$ — формулы.
3. Если $\mathfrak{A}$ — формула, то $(\,\overline{\mathfrak{A}}\,)$ — формула.
4. Никаких других формул не существует.

Формальное определение формулы

Формула алгебры логики формально определяется следующим образом:

1. Пропозициональная переменная — (атомарная) формула.
2. Если $\mathfrak{A}$ и $\mathfrak{B}$ — формулы, то $(\mathfrak{A} \mathbin{\&} \mathfrak{B})$, $(\mathfrak{A} \vee \mathfrak{B})$, $(\mathfrak{A} \to \mathfrak{B})$, $(\mathfrak{A} \leftrightarrow \mathfrak{B})$ — формулы.
3. Если $\mathfrak{A}$ — формула, то $(\,\overline{\mathfrak{A}}\,)$ — формула.
4. Никаких других формул не существует.

Например, выражение $((X \mathbin{\&} Y) \leftrightarrow (\,\overline{Z}\,))$ является формулой.

Скобки и приоритет операций

Для формирования сложных формул используют вспомогательные символы — скобки «(» и «)».

Скобки и приоритет операций

Для формирования сложных формул используют вспомогательные символы — скобки «(» и «)». Лишние наружные скобки, как правило, опускают.

Скобки и приоритет операций

Для формирования сложных формул используют вспомогательные символы — скобки «(» и «)».
Лишние наружные скобки, как правило, опускают.
Также вводится приоритет операций:

Скобки и приоритет операций

Для формирования сложных формул используют вспомогательные символы — скобки «(» и «)».
Лишние наружные скобки, как правило, опускают.
Также вводится приоритет операций:

1. Отрицание $\overline{A}$ — наивысший приоритет;

Скобки и приоритет операций

Для формирования сложных формул используют вспомогательные символы — скобки «(» и «)».
Лишние наружные скобки, как правило, опускают.
Также вводится приоритет операций:

1. Отрицание $\overline{A}$ — наивысший приоритет;
2. Конъюнкция $A \,\&\, B$;

Скобки и приоритет операций

Для формирования сложных формул используют вспомогательные символы — скобки «(» и «)».
Лишние наружные скобки, как правило, опускают.
Также вводится приоритет операций:

1. Отрицание $\overline{A}$ — наивысший приоритет;
2. Конъюнкция $A \mathbin{\&} B$;
3. Дизъюнкция $A \vee B$;

Скобки и приоритет операций

Для формирования сложных формул используют вспомогательные символы — скобки «(» и «)».
Лишние наружные скобки, как правило, опускают.
Также вводится приоритет операций:

1. Отрицание $\overline{A}$ — наивысший приоритет;
2. Конъюнкция $A \,\&\, B$;
3. Дизъюнкция $A \vee B$;
4. Импликация $A \to B$;

Скобки и приоритет операций

Для формирования сложных формул используют вспомогательные символы — скобки «(» и «)».
Лишние наружные скобки, как правило, опускают.
Также вводится приоритет операций:

1. Отрицание $\overline{A}$ — наивысший приоритет;
2. Конъюнкция $A \mathbin{\&} B$;
3. Дизъюнкция $A \vee B$;
4. Импликация $A \to B$;
5. Эквивалентность $A \leftrightarrow B$ — самый низкий приоритет.

Скобки и приоритет операций

Для формирования сложных формул используют вспомогательные символы — скобки «(» и «)».
Лишние наружные скобки, как правило, опускают.
Также вводится приоритет операций:

1. Отрицание $\overline{A}$ — наивысший приоритет;
2. Конъюнкция $A \,\&\, B$;
3. Дизъюнкция $A \vee B$;
4. Импликация $A \to B$;
5. Эквивалентность $A \leftrightarrow B$ — самый низкий приоритет.

Например, формулу $\Big(\big(A \vee (D \,\&\, (\overline{B}))\big) \to (C \leftrightarrow E)\Big)$ можно записать так: $A \vee D \,\&\, \overline{B} \to (C \leftrightarrow E)$.

Булевы функции

Каждой формуле алгебры логики соответствует функция, принимающая значения из множества $\{0, 1\}$, аргументы которой также принимают значения из множества $\{0, 1\}$:

Булевы функции

Каждой формуле алгебры логики соответствует функция, принимающая значения из множества $\{0, 1\}$, аргументы которой также принимают значения из множества $\{0, 1\}$:

$$f\colon \{0, 1\}^n \longrightarrow \{0, 1\}.$$

Булевы функции

Каждой формуле алгебры логики соответствует функция, принимающая значения из множества $\{0, 1\}$, аргументы которой также принимают значения из множества $\{0, 1\}$:

$$f\colon \{0, 1\}^n \longrightarrow \{0, 1\}.$$

Такие функции называют
булевыми функциями,
или логическими функциями,
или функциями алгебры логики.

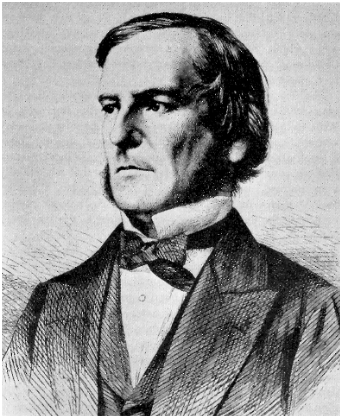
Булевы функции

Каждой формуле алгебры логики соответствует функция, принимающая значения из множества $\{0,1\}$, аргументы которой также принимают значения из множества $\{0,1\}$:

$$f\colon \{0,1\}^n \longrightarrow \{0,1\}.$$

Такие функции называют
булевыми функциями,
или логическими функциями,
или функциями алгебры логики.

Булевыми эти функции названы в честь Джорджа Буля — основателя алгебры логики.



Джордж Буль (1815—1864) — английский математик и логик.

Таблицы истинности

Каждая булева функция может быть изображена посредством таблицы истинности:

Таблицы истинности

Каждая булева функция может быть изображена посредством таблицы истинности:

A	B	$A \mathbin{\&} B$	$A \vee B$	$A \to B$	$A \leftrightarrow B$	$\overline{A}$
0	0	0	0	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	0	0
1	1	1	1	1	1	0

Таблицы истинности

Каждая булева функция может быть изображена посредством таблицы истинности:

A	B	$A \,\&\, B$	$A \vee B$	$A \to B$	$A \leftrightarrow B$	$\overline{A}$
0	0	0	0	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	0	0
1	1	1	1	1	1	0

Это табличный способ задания булевых функций.

Недостаток табличного способа задания функции

Для n-местной булевой функции $f(X_1, \ldots, X_n)$ таблица истинности будет содержать 2^n строк значений.

Недостаток табличного способа задания функции

Для n-местной булевой функции $f(X_1, \ldots, X_n)$ таблица истинности будет содержать 2^n строк значений.

Поэтому табличный способ не применяется для задания функций с большим количеством аргументов.

Недостаток табличного способа задания функции

Для n-местной булевой функции $f(X_1, \ldots, X_n)$ таблица истинности будет содержать 2^n строк значений.

Поэтому табличный способ не применяется для задания функций с большим количеством аргументов.

Например, для формулы $A \mathbin{\&} B \mathbin{\&} C$ таблица истинности соответствующей булевой функции будет содержать всего 8 строк,

а для формулы $A \mathbin{\&} \overline{B} \mathbin{\&} C \vee D \mathbin{\&} \overline{E} \mathbin{\&} F$ — уже 64 строки.

Задание функций с помощью формул

Пусть $X_1, \ldots, X_n$ — некоторый фиксированный список переменных.

Задание функций с помощью формул

Пусть $X_1, \ldots, X_n$ — некоторый фиксированный список переменных.
Каждой формуле $\mathfrak{A}$, содержащей переменные только из этого списка, соответствует n-местная функция $f(X_1, \ldots, X_n)$.

Задание функций с помощью формул

Пусть $X_1, \ldots, X_n$ — некоторый фиксированный список переменных.
Каждой формуле $\mathfrak{A}$, содержащей переменные только из этого списка, соответствует n-местная функция $f(X_1, \ldots, X_n)$.

Двоичным набором будем называть упорядоченное множество значений 0 и 1.

Задание функций с помощью формул

Пусть $X_1, \ldots, X_n$ — некоторый фиксированный список переменных.
Каждой формуле $\mathfrak{A}$, содержащей переменные только из этого списка, соответствует n-местная функция $f(X_1, \ldots, X_n)$.

Двоичным набором будем называть упорядоченное множество значений 0 и 1.

Например, для $n = 2$ имеем 4 различных двоичных набора: $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$.

Задание функций с помощью формул

Пусть $X_1, \dots, X_n$ — некоторый фиксированный список переменных.
Каждой формуле $\mathfrak{A}$, содержащей переменные только из этого списка, соответствует n-местная функция $f(X_1, \dots, X_n)$.

Двоичным набором будем называть упорядоченное множество значений 0 и 1.

Например, для $n = 2$ имеем 4 различных двоичных набора: $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$.

Пусть $(\sigma_1, \dots, \sigma_n)$ — некоторый двоичный набор.

Задание функций с помощью формул

Пусть $X_1, \dots, X_n$ — некоторый фиксированный список переменных.
Каждой формуле $\mathfrak{A}$, содержащей переменные только из этого списка, соответствует n-местная функция $f(X_1, \dots, X_n)$.

Двоичным набором будем называть упорядоченное множество значений 0 и 1.

Например, для $n = 2$ имеем 4 различных двоичных набора: $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$.

Пусть $(\sigma_1, \dots, \sigma_n)$ — некоторый двоичный набор.
Если переменные $X_1, \dots, X_n$ имеют истинностные значения $\sigma_1, \dots, \sigma_n$ соответственно, то, произведя вычисления в соответствии с операциями, указанными в формуле $\mathfrak{A}$, можно получить значение 0 или 1, которое и считается значением функции f на двоичном наборе $(\sigma_1, \dots, \sigma_n)$.

Задание функций с помощью формул

Пусть $X_1, \dots, X_n$ — некоторый фиксированный список переменных.
Каждой формуле $\mathfrak{A}$, содержащей переменные только из этого списка, соответствует n-местная функция $f(X_1, \dots, X_n)$.

Двоичным набором будем называть упорядоченное множество значений 0 и 1.

Например, для $n = 2$ имеем 4 различных двоичных набора: $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$.

Пусть $(\sigma_1, \dots, \sigma_n)$ — некоторый двоичный набор.
Если переменные $X_1, \dots, X_n$ имеют истинностные значения $\sigma_1, \dots, \sigma_n$ соответственно, то, произведя вычисления в соответствии с операциями, указанными в формуле $\mathfrak{A}$, можно получить значение 0 или 1, которое и считается значением функции f на двоичном наборе $(\sigma_1, \dots, \sigma_n)$.

Говорят, что формула $\mathfrak{A}$ реализует функцию f.

Пример задания функции формулой

Например, формула $(X \mathbin{\&} Y) \leftrightarrow \overline{Z}$ реализует функцию $h(X, Y, Z)$:

Пример задания функции формулой

Например, формула $(X \mathbin{\&} Y) \leftrightarrow \overline{Z}$ реализует функцию $h(X, Y, Z)$:

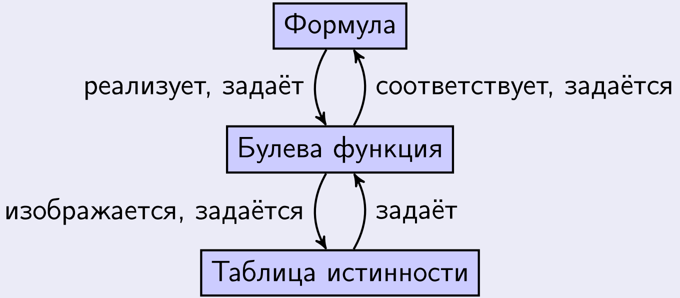
X	Y	Z	$h(X, Y, Z)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Связи между понятиями

Связь между понятиями «формула», «булева функция» и «таблица истинности» изображена на рисунке:

Связи между понятиями

Связь между понятиями «формула», «булева функция» и «таблица истинности» изображена на рисунке:



Равносильные формулы

Пусть $\mathfrak{A}$ и $\mathfrak{B}$ — две формулы, содержащие лишь переменные из списка $X_1, \ldots, X_n$, и пусть $f(X_1, \ldots, X_n)$ и $g(X_1, \ldots, X_n)$ — функции, реализуемые формулами $\mathfrak{A}$ и $\mathfrak{B}$ соответственно.

Равносильные формулы

Пусть $\mathfrak{A}$ и $\mathfrak{B}$ — две формулы, содержащие лишь переменные из списка $X_1, \ldots, X_n$, и пусть $f(X_1, \ldots, X_n)$ и $g(X_1, \ldots, X_n)$ — функции, реализуемые формулами $\mathfrak{A}$ и $\mathfrak{B}$ соответственно.

Формулы $\mathfrak{A}$ и $\mathfrak{B}$ называют равносильными, если функции f и g совпадают, т. е. совпадают их таблицы истинности.

Равносильные формулы

Пусть $\mathfrak{A}$ и $\mathfrak{B}$ — две формулы, содержащие лишь переменные из списка $X_1, \ldots, X_n$, и пусть $f(X_1, \ldots, X_n)$ и $g(X_1, \ldots, X_n)$ — функции, реализуемые формулами $\mathfrak{A}$ и $\mathfrak{B}$ соответственно.

Формулы $\mathfrak{A}$ и $\mathfrak{B}$ называют равносильными, если функции f и g совпадают, т. е. совпадают их таблицы истинности.

Для равносильных формул будем использовать обозначение:

$$\mathfrak{A} \equiv \mathfrak{B}.$$

Важные равенства (1—4)

1. Законы коммутативности (переместительности):
   a. $A \,\&\, B \equiv B \,\&\, A$ — коммутативность конъюнкции;
   б. $A \vee B \equiv B \vee A$ — коммутативность дизъюнкции.

Важные равенства (1—4)

1. Законы коммутативности (переместительности):
   - а) $A \mathbin{\&} B \equiv B \mathbin{\&} A$ — коммутативность конъюнкции;
   - б) $A \vee B \equiv B \vee A$ — коммутативность дизъюнкции.
2. Законы ассоциативности (сочетательности):
   - а) $(A \mathbin{\&} B) \mathbin{\&} C \equiv A \mathbin{\&} (B \mathbin{\&} C)$ — ассоциативность конъюнкции;
   - б) $(A \vee B) \vee C \equiv A \vee (B \vee C)$ — ассоциативность дизъюнкции.

Важные равенства (1—4)

1. Законы коммутативности (переместительности):
   - а) $A \mathbin{\&} B \equiv B \mathbin{\&} A$ — коммутативность конъюнкции;
   - б) $A \vee B \equiv B \vee A$ — коммутативность дизъюнкции.
2. Законы ассоциативности (сочетательности):
   - а) $(A \mathbin{\&} B) \mathbin{\&} C \equiv A \mathbin{\&} (B \mathbin{\&} C)$ — ассоциативность конъюнкции;
   - б) $(A \vee B) \vee C \equiv A \vee (B \vee C)$ — ассоциативность дизъюнкции.
3. Законы идемпотентности:
   - а) $A \mathbin{\&} A \equiv A$ — идемпотентность конъюнкции;
   - б) $A \vee A \equiv A$ — идемпотентность дизъюнкции.

   По законам идемпотентности в логике нет коэффициентов и показателей степени.

Важные равенства (1—4)

1. Законы коммутативности (переместительности):
   а) $A \mathbin{\&} B \equiv B \mathbin{\&} A$ — коммутативность конъюнкции;
   б) $A \vee B \equiv B \vee A$ — коммутативность дизъюнкции.
2. Законы ассоциативности (сочетательности):
   а) $(A \mathbin{\&} B) \mathbin{\&} C \equiv A \mathbin{\&} (B \mathbin{\&} C)$ — ассоциативность конъюнкции;
   б) $(A \vee B) \vee C \equiv A \vee (B \vee C)$ — ассоциативность дизъюнкции.
3. Законы идемпотентности:
   а) $A \mathbin{\&} A \equiv A$ — идемпотентность конъюнкции;
   б) $A \vee A \equiv A$ — идемпотентность дизъюнкции.

   По законам идемпотентности в логике нет коэффициентов и показателей степени.
4. Законы дистрибутивности (распределительности):
   а) $A \mathbin{\&} (B \vee C) \equiv (A \mathbin{\&} B) \vee (A \mathbin{\&} C)$ — дистрибутивность конъюнкции относительно дизъюнкции;
   б) $A \vee (B \mathbin{\&} C) \equiv (A \vee B) \mathbin{\&} (A \vee C)$ — дистрибутивность дизъюнкции относительно конъюнкции.

Важные равенства (5—7)

5. Свойства нуля и единицы:
   - а) $A \,\&\, 0 \equiv 0$;
   - б) $A \vee 0 \equiv A$;
   - в) $A \,\&\, 1 \equiv A$;
   - г) $A \vee 1 \equiv 1$.

Важные равенства (5—7)

5. Свойства нуля и единицы:
   а) $A \mathbin{\&} 0 \equiv 0$;
   б) $A \vee 0 \equiv A$;
   в) $A \mathbin{\&} 1 \equiv A$;
   г) $A \vee 1 \equiv 1$.

6. $\overline{\overline{A}} \equiv A$ — правило снятия двойного отрицания;

Важные равенства (5—7)

5. Свойства нуля и единицы:
   а. $A \,\&\, 0 \equiv 0$;
   б. $A \vee 0 \equiv A$;
   в. $A \,\&\, 1 \equiv A$;
   г. $A \vee 1 \equiv 1$.

6. $\overline{\overline{A}} \equiv A$ — правило снятия двойного отрицания;

7. Законы де Мóргана:
   а. $\overline{A \,\&\, B} \equiv \overline{A} \vee \overline{B}$;
   б. $\overline{A \vee B} \equiv \overline{A} \,\&\, \overline{B}$.

Огастес (Август) де Морган (1806—1871) — шотландский математик и логик.

Важные равенства (8—11)

8. $A \,\&\, \overline{A} \equiv 0$ — закон противоречия.

Важные равенства (8—11)

8. $A \,\&\, \overline{A} \equiv 0$ — закон противоречия.

9. $A \vee \overline{A} \equiv 1$ — закон исключённого третьего.

Важные равенства (8—11)

8. $A \mathbin{\&} \overline{A} \equiv 0$ — закон противоречия.
9. $A \vee \overline{A} \equiv 1$ — закон исключённого третьего.
10. Законы поглощения:
    а. $A \mathbin{\&} (A \vee B) \equiv A$;
    б. $A \vee (A \mathbin{\&} B) \equiv A$.

Важные равенства (8—11)

8. $A \mathbin{\&} \overline{A} \equiv 0$ — закон противоречия.
9. $A \vee \overline{A} \equiv 1$ — закон исключённого третьего.
10. Законы поглощения:
    а. $A \mathbin{\&} (A \vee B) \equiv A$;
    б. $A \vee (A \mathbin{\&} B) \equiv A$.
11. Правила склеивания-расщепления:
    а. $(A \vee \overline{B}) \mathbin{\&} (A \vee B) \equiv A$;
    б. $(A \mathbin{\&} \overline{B}) \vee (A \mathbin{\&} B) \equiv A$.

    Если данные правила применяются для уменьшения числа операций, то говорят, что производится склеивание, а если наоборот, то расщепление.

Важные равенства (12—14)

12. $A \to B \equiv \overline{A} \vee B$ — выражение импликации через дизъюнкцию и отрицание.

Важные равенства (12—14)

12. $A \to B \equiv \overline{A} \vee B$ — выражение импликации через дизъюнкцию и отрицание.

13. $A \leftrightarrow B \equiv (A \to B) \mathbin{\&} (B \to A)$ — выражение эквивалентности через конъюнкцию и импликацию.

Важные равенства (12—14)

12. $A \to B \equiv \overline{A} \vee B$ — выражение импликации через дизъюнкцию и отрицание.
13. $A \leftrightarrow B \equiv (A \to B) \,\&\, (B \to A)$ — выражение эквивалентности через конъюнкцию и импликацию.
14. Выражение эквивалентности через конъюнкцию, дизъюнкцию и отрицание:
    - а) $A \leftrightarrow B \equiv (A \,\&\, B) \vee (\overline{A} \,\&\, \overline{B})$;
    - б) $A \leftrightarrow B \equiv (\overline{A} \vee B) \,\&\, (A \vee \overline{B})$.

Правило подстановки

Пусть $\mathfrak{A} \equiv \mathfrak{B}$, пусть $X_1, \ldots, X_n$ — все переменные, фигурирующие в формулах $\mathfrak{A}$ и $\mathfrak{B}$, и пусть формула $\mathfrak{A}'$ получается из $\mathfrak{A}$, а формула $\mathfrak{B}'$ — из $\mathfrak{B}$ путём подстановки некоторых формул $\mathfrak{C}_1, \ldots, \mathfrak{C}_n$ вместо переменных $X_1, \ldots, X_n$.

Правило подстановки

Пусть $\mathfrak{A} \equiv \mathfrak{B}$, пусть $X_1, \ldots, X_n$ — все переменные, фигурирующие в формулах $\mathfrak{A}$ и $\mathfrak{B}$, и пусть формула $\mathfrak{A}'$ получается из $\mathfrak{A}$, а формула $\mathfrak{B}'$ — из $\mathfrak{B}$ путём подстановки некоторых формул $\mathfrak{C}_1, \ldots, \mathfrak{C}_n$ вместо переменных $X_1, \ldots, X_n$. Тогда $\mathfrak{A}' \equiv \mathfrak{B}'$.

Правило подстановки

Пусть $\mathfrak{A} \equiv \mathfrak{B}$, пусть $X_1, \ldots, X_n$ — все переменные, фигурирующие в формулах $\mathfrak{A}$ и $\mathfrak{B}$, и пусть формула $\mathfrak{A}'$ получается из $\mathfrak{A}$, а формула $\mathfrak{B}'$ — из $\mathfrak{B}$ путём подстановки некоторых формул $\mathfrak{C}_1, \ldots, \mathfrak{C}_n$ вместо переменных $X_1, \ldots, X_n$. Тогда $\mathfrak{A}' \equiv \mathfrak{B}'$.

В более узком смысле правило подстановки звучит так:

Если в равносильных формулах вместо всех вхождений некоторой переменной подставить одну и ту же формулу, то получатся новые равносильные формулы.

Правило замены

Пусть $\mathfrak{A} \equiv \mathfrak{B}$, а формула $\mathfrak{C}$ содержит $\mathfrak{A}$ в качестве составной части (т. е. $\mathfrak{A}$ является подформулой формулы $\mathfrak{C}$).

Правило замены

Пусть $\mathfrak{A} \equiv \mathfrak{B}$, а формула $\mathfrak{C}$ содержит $\mathfrak{A}$ в качестве составной части (т. е. $\mathfrak{A}$ является подформулой формулы $\mathfrak{C}$).
Пусть $\mathfrak{C}'$ получается из $\mathfrak{C}$ подстановкой формулы $\mathfrak{B}$ вместо $\mathfrak{A}$.

Правило замены

Пусть $\mathfrak{A} \equiv \mathfrak{B}$, а формула $\mathfrak{C}$ содержит $\mathfrak{A}$ в качестве составной части (т. е. $\mathfrak{A}$ является подформулой формулы $\mathfrak{C}$).
Пусть $\mathfrak{C}'$ получается из $\mathfrak{C}$ подстановкой формулы $\mathfrak{B}$ вместо $\mathfrak{A}$.
Тогда $\mathfrak{C}' \equiv \mathfrak{C}$.

Правило замены

Пусть $\mathfrak{A} \equiv \mathfrak{B}$, а формула $\mathfrak{C}$ содержит $\mathfrak{A}$ в качестве составной части (т. е. $\mathfrak{A}$ является подформулой формулы $\mathfrak{C}$).
Пусть $\mathfrak{C}'$ получается из $\mathfrak{C}$ подстановкой формулы $\mathfrak{B}$ вместо $\mathfrak{A}$.
Тогда $\mathfrak{C}' \equiv \mathfrak{C}$.

Правило замены иначе можно сформулировать так:

Если в формуле заменить некоторую подформулу на равносильную, то получится новая формула, равносильная исходной.

Правило замены

Пусть $\mathfrak{A} \equiv \mathfrak{B}$, а формула $\mathfrak{C}$ содержит $\mathfrak{A}$ в качестве составной части (т. е. $\mathfrak{A}$ является подформулой формулы $\mathfrak{C}$).
Пусть $\mathfrak{C}'$ получается из $\mathfrak{C}$ подстановкой формулы $\mathfrak{B}$ вместо $\mathfrak{A}$.
Тогда $\mathfrak{C}' \equiv \mathfrak{C}$.

Правило замены иначе можно сформулировать так:

Если в формуле заменить некоторую подформулу на равносильную, то получится новая формула, равносильная исходной.

В математике преобразование формулы в равносильную называется эквивалентным преобразованием.

Интерпретации

Пусть задана некоторая формула $\mathfrak{A}$, реализующая булеву функцию $f(X_1, \ldots, X_n)$.

Интерпретации

Пусть задана некоторая формула $\mathfrak{A}$, реализующая булеву функцию $f(X_1, \ldots, X_n)$.

Интерпретацией формулы $\mathfrak{A}$ будем называть некоторый двоичный набор $(\sigma_1, \ldots, \sigma_n)$, который соответствует значениям переменных $X_1, \ldots, X_n$.

Интерпретации

Пусть задана некоторая формула $\mathfrak{A}$, реализующая булеву функцию $f(X_1, \ldots, X_n)$.

Интерпретацией формулы $\mathfrak{A}$ будем называть некоторый двоичный набор $(\sigma_1, \ldots, \sigma_n)$, который соответствует значениям переменных $X_1, \ldots, X_n$.

Пример

Рассмотрим высказывание: «Петя сильный и умный».

Интерпретации

Пусть задана некоторая формула $\mathfrak{A}$, реализующая булеву функцию $f(X_1, \ldots, X_n)$.

Интерпретацией формулы $\mathfrak{A}$ будем называть некоторый двоичный набор $(\sigma_1, \ldots, \sigma_n)$, который соответствует значениям переменных $X_1, \ldots, X_n$.

Пример

Рассмотрим высказывание: «Петя сильный и умный».
Ему соответствует формула $\mathfrak{A} = S \,\&\, U$, реализующая булеву функцию $f(S, U)$.

Интерпретации

Пусть задана некоторая формула $\mathfrak{A}$, реализующая булеву функцию $f(X_1, \ldots, X_n)$.

Интерпретацией формулы $\mathfrak{A}$ будем называть некоторый двоичный набор $(\sigma_1, \ldots, \sigma_n)$, который соответствует значениям переменных $X_1, \ldots, X_n$.

Пример

Рассмотрим высказывание: «Петя сильный и умный».
Ему соответствует формула $\mathfrak{A} = S \mathbin{\&} U$, реализующая булеву функцию $f(S, U)$.
Если в рассматриваемом контексте Петя сильный, но глупый, то такая интерпретация будет соответствовать двоичному набору $(1, 0)$, т. е. $S = 1$ и $U = 0$.

Интерпретации

Пусть задана некоторая формула $\mathfrak{A}$, реализующая булеву функцию $f(X_1, \ldots, X_n)$.

Интерпретацией формулы $\mathfrak{A}$ будем называть некоторый двоичный набор $(\sigma_1, \ldots, \sigma_n)$, который соответствует значениям переменных $X_1, \ldots, X_n$.

Пример

Рассмотрим высказывание: «Петя сильный и умный».
Ему соответствует формула $\mathfrak{A} = S \mathbin{\&} U$, реализующая булеву функцию $f(S, U)$.
Если в рассматриваемом контексте Петя сильный, но глупый, то такая интерпретация будет соответствовать двоичному набору $(1, 0)$, т. е. $S = 1$ и $U = 0$.
При этом очевидно, что $f(1, 0) = 0$.

Общезначимые, противоречивые и выполнимые формулы

Формула $\mathfrak{A}$ называется общезначимой, или тождественно истинной, или тавтологией, если $\mathfrak{A} \equiv 1$.

Общезначимые, противоречивые и выполнимые формулы

Формула $\mathfrak{A}$ называется общезначимой,
или тождественно истинной, или тавтологией, если $\mathfrak{A} \equiv 1$.

Очевидно, что в этом случае $f = 1$ на любой интерпретации.

Общезначимые, противоречивые и выполнимые формулы

Формула $\mathfrak{A}$ называется общезначимой,
или тождественно истинной, или тавтологией, если $\mathfrak{A} \equiv 1$.

Очевидно, что в этом случае $f = 1$ на любой интерпретации.

Для общезначимой формулы используется запись
с использованием знака выводимости: $\vdash \mathfrak{A}$.

Общезначимые, противоречивые и выполнимые формулы

Формула $\mathfrak{A}$ называется общезначимой,
или тождественно истинной, или тавтологией, если $\mathfrak{A} \equiv 1$.

Очевидно, что в этом случае $f = 1$ на любой интерпретации.
Для общезначимой формулы используется запись
с использованием знака выводимости: $\vdash \mathfrak{A}$.

Формула $\mathfrak{A}$ называется невыполнимой,
или тождественно ложной, или противоречием, если $\mathfrak{A} \equiv 0$.

Общезначимые, противоречивые и выполнимые формулы

Формула $\mathfrak{A}$ называется общезначимой,
или тождественно истинной, или тавтологией, если $\mathfrak{A} \equiv 1$.

Очевидно, что в этом случае $f = 1$ на любой интерпретации.
Для общезначимой формулы используется запись
с использованием знака выводимости: $\vdash \mathfrak{A}$.

Формула $\mathfrak{A}$ называется невыполнимой,
или тождественно ложной, или противоречием, если $\mathfrak{A} \equiv 0$.

Очевидно, что в этом случае $f = 0$ на любой интерпретации.

Общезначимые, противоречивые и выполнимые формулы

Формула $\mathfrak{A}$ называется общезначимой,
или тождественно истинной, или тавтологией, если $\mathfrak{A} \equiv 1$.

Очевидно, что в этом случае $f = 1$ на любой интерпретации.
Для общезначимой формулы используется запись
с использованием знака выводимости: $\vdash \mathfrak{A}$.

Формула $\mathfrak{A}$ называется невыполнимой,
или тождественно ложной, или противоречием, если $\mathfrak{A} \equiv 0$.

Очевидно, что в этом случае $f = 0$ на любой интерпретации.

Формула $\mathfrak{A}$, не являющаяся ни противоречием, ни тавтологией, называется выполнимой.

Проблема разрешимости

Для доказательства того, что формула $\mathfrak{A}$ является выполнимой, необходимо и достаточно найти такие интерпретации $(\sigma_1, \ldots, \sigma_n)$ и $(\tau_1, \ldots, \tau_n)$, что $f(\sigma_1, \ldots, \sigma_n) = 0$ и $f(\tau_1, \ldots, \tau_n) = 1$.

Проблема разрешимости

Для доказательства того, что формула $\mathfrak{A}$ является выполнимой, необходимо и достаточно найти такие интерпретации $(\sigma_1, \ldots, \sigma_n)$ и $(\tau_1, \ldots, \tau_n)$, что $f(\sigma_1, \ldots, \sigma_n) = 0$ и $f(\tau_1, \ldots, \tau_n) = 1$.

Пусть дана произвольная формула $\mathfrak{A}$.
Можно ли как-то проверить, что она является общезначимой?

Проблема разрешимости

Для доказательства того, что формула $\mathfrak{A}$ является выполнимой, необходимо и достаточно найти такие интерпретации $(\sigma_1, \ldots, \sigma_n)$ и $(\tau_1, \ldots, \tau_n)$, что $f(\sigma_1, \ldots, \sigma_n) = 0$ и $f(\tau_1, \ldots, \tau_n) = 1$.

Пусть дана произвольная формула $\mathfrak{A}$.
Можно ли как-то проверить, что она является общезначимой?
Если существует такой способ (алгоритм), позволяющий за конечное число шагов убедиться в этом, то говорят, что проблема проверки общезначимости формул (проблема разрешимости) алгебры логики разрешима.

Проблема разрешимости

Для доказательства того, что формула $\mathfrak{A}$ является выполнимой, необходимо и достаточно найти такие интерпретации $(\sigma_1, \ldots, \sigma_n)$ и $(\tau_1, \ldots, \tau_n)$, что $f(\sigma_1, \ldots, \sigma_n) = 0$ и $f(\tau_1, \ldots, \tau_n) = 1$.

Пусть дана произвольная формула $\mathfrak{A}$.
Можно ли как-то проверить, что она является общезначимой? Если существует такой способ (алгоритм), позволяющий за конечное число шагов убедиться в этом, то говорят, что проблема проверки общезначимости формул (проблема разрешимости) алгебры логики разрешима.

Теорема

Проблема разрешимости в алгебре логики имеет положительное решение.

Доказательство теоремы о проблеме разрешимости

Так как формуле $\mathfrak{A}$ однозначно соответствует n-местная булева функция $f(X_1, \ldots, X_n)$, причём n — конечное число, то имеем 2^n различных двоичных наборов $(\sigma_1, \ldots, \sigma_n)$.

Доказательство теоремы о проблеме разрешимости

Так как формуле $\mathfrak{A}$ однозначно соответствует n-местная булева функция $f(X_1, \ldots, X_n)$, причём n — конечное число, то имеем 2^n различных двоичных наборов $(\sigma_1, \ldots, \sigma_n)$.
Для каждого набора можно вычислить значение функции f за конечное число шагов.

Доказательство теоремы о проблеме разрешимости

Так как формуле $\mathfrak{A}$ однозначно соответствует n-местная булева функция $f(X_1, \ldots, X_n)$, причём n — конечное число, то имеем 2^n различных двоичных наборов $(\sigma_1, \ldots, \sigma_n)$.
Для каждого набора можно вычислить значение функции f за конечное число шагов.
Найдя значения функции, мы узнаем, все ли они равны 1, т. е. истинна ли формула $\mathfrak{A}$ на всех интерпретациях.

Доказательство теоремы о проблеме разрешимости

Так как формуле $\mathfrak{A}$ однозначно соответствует n-местная булева функция $f(X_1, \ldots, X_n)$, причём n — конечное число, то имеем 2^n различных двоичных наборов $(\sigma_1, \ldots, \sigma_n)$.
Для каждого набора можно вычислить значение функции f за конечное число шагов.
Найдя значения функции, мы узнаем, все ли они равны 1, т. е. истинна ли формула $\mathfrak{A}$ на всех интерпретациях.
Если это так, то $\vdash \mathfrak{A}$.

Доказательство теоремы о проблеме разрешимости

Так как формуле $\mathfrak{A}$ однозначно соответствует n-местная булева функция $f(X_1, \ldots, X_n)$, причём n — конечное число, то имеем 2^n различных двоичных наборов $(\sigma_1, \ldots, \sigma_n)$.
Для каждого набора можно вычислить значение функции f за конечное число шагов.
Найдя значения функции, мы узнаем, все ли они равны 1, т. е. истинна ли формула $\mathfrak{A}$ на всех интерпретациях.
Если это так, то $\vdash \mathfrak{A}$.
Таким образом, мы можем проверить общезначимость любой формулы за конечное число шагов.

Доказательство теоремы о проблеме разрешимости

Так как формуле $\mathfrak{A}$ однозначно соответствует n-местная булева функция $f(X_1, \ldots, X_n)$, причём n — конечное число, то имеем 2^n различных двоичных наборов $(\sigma_1, \ldots, \sigma_n)$.
Для каждого набора можно вычислить значение функции f за конечное число шагов.
Найдя значения функции, мы узнаем, все ли они равны 1, т. е. истинна ли формула $\mathfrak{A}$ на всех интерпретациях.
Если это так, то $\vdash \mathfrak{A}$.
Таким образом, мы можем проверить общезначимость любой формулы за конечное число шагов.
Теорема доказана.

Нормальные формы

Пропозициональную переменную с отрицанием или без него будем называть литералом:

Нормальные формы

Пропозициональную переменную с отрицанием или без него будем называть литералом:

$$X^\sigma = \begin{cases} X & \text{при } \sigma = 1, \\ \overline{X} & \text{при } \sigma = 0. \end{cases}$$

Нормальные формы

Пропозициональную переменную с отрицанием или без него будем называть литералом:

$$X^{\sigma} = \begin{cases} X & \text{при } \sigma = 1, \\ \overline{X} & \text{при } \sigma = 0. \end{cases}$$

Формула вида $X_1^{\sigma_1} \mathbin{\&} \ldots \mathbin{\&} X_n^{\sigma_n}$, где $(\sigma_1, \ldots, \sigma_n)$ — двоичный набор ($n \geqslant 1$), а среди переменных могут быть одинаковые, называется элементарной конъюнкцией или конъюнктом.

Нормальные формы

Пропозициональную переменную с отрицанием или без него будем называть литералом:

$$X^\sigma = \begin{cases} X & \text{при } \sigma = 1, \\ \overline{X} & \text{при } \sigma = 0. \end{cases}$$

Формула вида $X_1^{\sigma_1} \,\&\, \ldots \,\&\, X_n^{\sigma_n}$, где $(\sigma_1, \ldots, \sigma_n)$ — двоичный набор ($n \geqslant 1$), а среди переменных могут быть одинаковые, называется элементарной конъюнкцией или конъюнктом.

Всякая дизъюнкция элементарных конъюнкций называется дизъюнктивной нормальной формой (ДНФ).

Нормальные формы

Пропозициональную переменную с отрицанием или без него будем называть литералом:

$$X^\sigma = \begin{cases} X & \text{при } \sigma = 1, \\ \overline{X} & \text{при } \sigma = 0. \end{cases}$$

Формула вида $X_1^{\sigma_1} \mathbin{\&} \ldots \mathbin{\&} X_n^{\sigma_n}$, где $(\sigma_1, \ldots, \sigma_n)$ — двоичный набор ($n \geqslant 1$), а среди переменных могут быть одинаковые, называется элементарной конъюнкцией или конъюнктом.

Всякая дизъюнкция элементарных конъюнкций называется дизъюнктивной нормальной формой (ДНФ).

Аналогично, формула вида $X_1^{\sigma_1} \vee \ldots \vee X_n^{\sigma_n}$ называется элементарной дизъюнкцией или дизъюнктом.

Нормальные формы

Пропозициональную переменную с отрицанием или без него будем называть литералом:

$$X^\sigma = \begin{cases} X & \text{при } \sigma = 1, \\ \overline{X} & \text{при } \sigma = 0. \end{cases}$$

Формула вида $X_1^{\sigma_1} \,\&\, \ldots \,\&\, X_n^{\sigma_n}$, где $(\sigma_1, \ldots, \sigma_n)$ — двоичный набор ($n \geqslant 1$), а среди переменных могут быть одинаковые, называется элементарной конъюнкцией или конъюнктом.

Всякая дизъюнкция элементарных конъюнкций называется дизъюнктивной нормальной формой (ДНФ).

Аналогично, формула вида $X_1^{\sigma_1} \vee \ldots \vee X_n^{\sigma_n}$ называется элементарной дизъюнкцией или дизъюнктом.

Конъюнкция элементарных дизъюнкций называется конъюнктивной нормальной формой (КНФ).

Алгоритм построения ДНФ и КНФ

С помощью эквивалентных преобразований для всякой формулы можно построить равносильные ей ДНФ и КНФ:

Алгоритм построения ДНФ и КНФ

С помощью эквивалентных преобразований для всякой формулы можно построить равносильные ей ДНФ и КНФ:

1. С помощью равенств $A \leftrightarrow B \equiv (A \to B)\,\&\,(B \to A)$ и $A \to B \equiv \overline{A} \vee B$ удалить все знаки импликации и эквивалентности.

Алгоритм построения ДНФ и КНФ

С помощью эквивалентных преобразований для всякой формулы можно построить равносильные ей ДНФ и КНФ:

1. С помощью равенств $A \leftrightarrow B \equiv (A \to B)\,\&\,(B \to A)$ и $A \to B \equiv \overline{A} \vee B$ удалить все знаки импликации и эквивалентности.
2. Применяя правило $\overline{\overline{A}} \equiv A$ и законы де Моргана $\overline{A\,\&\,B} \equiv \overline{A} \vee \overline{B}$ и $\overline{A \vee B} \equiv \overline{A}\,\&\,\overline{B}$, можно добиться того, чтобы знак отрицания стоял только над переменными и чтобы не было многократных отрицаний.

Алгоритм построения ДНФ и КНФ

С помощью эквивалентных преобразований для всякой формулы можно построить равносильные ей ДНФ и КНФ:

1. С помощью равенств $A \leftrightarrow B \equiv (A \to B) \,\&\, (B \to A)$ и $A \to B \equiv \overline{A} \vee B$ удалить все знаки импликации и эквивалентности.
2. Применяя правило $\overline{\overline{A}} \equiv A$ и законы де Моргана $\overline{A \,\&\, B} \equiv \overline{A} \vee \overline{B}$ и $\overline{A \vee B} \equiv \overline{A} \,\&\, \overline{B}$, можно добиться того, чтобы знак отрицания стоял только над переменными и чтобы не было многократных отрицаний.
3. С помощью законов дистрибутивности $A \,\&\, (B \vee C) \equiv (A \,\&\, B) \vee (A \,\&\, C)$ и $A \vee (B \,\&\, C) \equiv (A \vee B) \,\&\, (A \vee C)$ получить КНФ или ДНФ.

Пример построения КНФ

Привести формулу $(A \to B) \to C$ к КНФ.

Пример построения КНФ

Привести формулу $(A \to B) \to C$ к КНФ.

Решение

Пример построения КНФ

Привести формулу $(A \to B) \to C$ к КНФ.

Решение

$(A \to B) \to C \equiv (\overline{A} \vee B) \to C \equiv$

Пример построения КНФ

Привести формулу $(A \to B) \to C$ к КНФ.

Решение

$$
\begin{aligned}
(A \to B) \to C &\equiv \underline{(\overline{A} \vee B)} \to C \equiv \\
&\equiv \overline{(\overline{A} \vee B)} \vee C \equiv
\end{aligned}
$$

Пример построения КНФ

Привести формулу $(A \to B) \to C$ к КНФ.

Решение

$$
\begin{aligned}
(A \to B) \to C &\equiv \underline{(\overline{A} \vee B)} \to C \equiv \\
&\equiv \overline{(\overline{A} \vee B)} \vee C \equiv \\
&\equiv (\overline{\overline{A}} \,\&\, \overline{B}) \vee C \equiv
\end{aligned}
$$

Пример построения КНФ

Привести формулу $(A \to B) \to C$ к КНФ.

Решение

$$
\begin{aligned}
(A \to B) \to C &\equiv (\overline{A} \vee B) \to C \equiv \\
&\equiv \overline{(\overline{A} \vee B)} \vee C \equiv \\
&\equiv (\overline{\overline{A}} \,\&\, \overline{B}) \vee C \equiv \\
&\equiv (A \,\&\, \overline{B}) \vee C \equiv
\end{aligned}
$$

Пример построения КНФ

Привести формулу $(A \to B) \to C$ к КНФ.

Решение

$$
\begin{aligned}
(A \to B) \to C &\equiv (\overline{A} \vee B) \to C \equiv \\
&\equiv \overline{(\overline{A} \vee B)} \vee C \equiv \\
&\equiv (\overline{\overline{A}} \,\&\, \overline{B}) \vee C \equiv \\
&\equiv (A \,\&\, \overline{B}) \vee C \equiv \\
&\equiv (A \vee C) \,\&\, (\overline{B} \vee C).
\end{aligned}
$$

Пример построения КНФ

Привести формулу $(A \to B) \to C$ к КНФ.

Решение

$$
\begin{aligned}
(A \to B) \to C &\equiv (\overline{A} \vee B) \to C \equiv \\
&\equiv \overline{(\overline{A} \vee B)} \vee C \equiv \\
&\equiv (\overline{\overline{A}} \mathbin{\&} \overline{B}) \vee C \equiv \\
&\equiv (A \mathbin{\&} \overline{B}) \vee C \equiv \\
&\equiv (A \vee C) \mathbin{\&} (\overline{B} \vee C).
\end{aligned}
$$

Ответ: $(A \vee C) \mathbin{\&} (\overline{B} \vee C)$.

Совершенные КНФ и ДНФ. Определения

Совершенные КНФ и ДНФ. Определения

Совершенная ДНФ (СДНФ) — ДНФ, в которой нет равных конъюнктов; все конъюнкты содержат одни и те же переменные, причём каждую переменную — только один раз (включая её вхождение под знаком отрицания).

Совершенные КНФ и ДНФ. Определения

Совершенная ДНФ (СДНФ) — ДНФ, в которой нет равных конъюнктов; все конъюнкты содержат одни и те же переменные, причём каждую переменную — только один раз (включая её вхождение под знаком отрицания).

Совершенная КНФ (СКНФ) — КНФ, в которой нет равных дизъюнктов; все дизъюнкты содержат одни и те же переменные, причём каждую переменную — только один раз (включая её вхождение под знаком отрицания).

СДНФ

Для каждой функции алгебры логики $f(X_1, \ldots, X_n)$, не равной тождественно 0, можно построить реализующую её СДНФ, причём единственную:

$$f(X_1, \ldots, X_n) = \bigvee_{f(\sigma_1, \ldots, \sigma_n) = 1} X_1^{\sigma_1} \,\&\, \ldots \,\&\, X_n^{\sigma_n},$$

СДНФ

Для каждой функции алгебры логики $f(X_1, \dots, X_n)$, не равной тождественно 0, можно построить реализующую её СДНФ, причём единственную:

$$f(X_1, \dots, X_n) = \bigvee_{f(\sigma_1,\dots,\sigma_n)=1} X_1^{\sigma_1} \,\&\, \dots \,\&\, X_n^{\sigma_n},$$

Дизъюнкция берётся по тем двоичным наборам $(\sigma_1, \dots, \sigma_n)$, на которых функция f равна 1.

СДНФ

Для каждой функции алгебры логики $f(X_1, \ldots, X_n)$, не равной тождественно 0, можно построить реализующую её СДНФ, причём единственную:

$$f(X_1, \ldots, X_n) = \bigvee_{f(\sigma_1,\ldots,\sigma_n)=1} X_1^{\sigma_1} \,\&\, \ldots \,\&\, X_n^{\sigma_n},$$

Дизъюнкция берётся по тем двоичным наборам $(\sigma_1, \ldots, \sigma_n)$, на которых функция f равна 1.

Построение СДНФ с помощью таблицы истинности

СДНФ

Для каждой функции алгебры логики $f(X_1, \dots, X_n)$, не равной тождественно 0, можно построить реализующую её СДНФ, причём единственную:

$$f(X_1, \dots, X_n) = \bigvee_{f(\sigma_1, \dots, \sigma_n) = 1} X_1^{\sigma_1} \,\&\, \dots \,\&\, X_n^{\sigma_n},$$

Дизъюнкция берётся по тем двоичным наборам $(\sigma_1, \dots, \sigma_n)$, на которых функция f равна 1.

Построение СДНФ с помощью таблицы истинности

1. В таблице истинности отыскиваются строки, в которых функция f равна 1.

СДНФ

Для каждой функции алгебры логики $f(X_1, \ldots, X_n)$, не равной тождественно 0, можно построить реализующую её СДНФ, причём единственную:

$$f(X_1, \ldots, X_n) = \bigvee_{f(\sigma_1,\ldots,\sigma_n)=1} X_1^{\sigma_1} \mathbin{\&} \ldots \mathbin{\&} X_n^{\sigma_n},$$

Дизъюнкция берётся по тем двоичным наборам $(\sigma_1, \ldots, \sigma_n)$, на которых функция f равна 1.

Построение СДНФ с помощью таблицы истинности

1. В таблице истинности отыскиваются строки, в которых функция f равна 1.
2. Каждой такой строке соответствует конъюнкт, состоящий из переменных, причём переменная входит в него под отрицанием, если в рассматриваемой строке значение данной переменной равно 0, и без отрицания, если её значение равно 1.

СКНФ

Каждая функция алгебры логики $f(X_1, \ldots, X_n)$, не равная тождественно 1, реализуется следующей СКНФ, причём единственной:

$$f(X_1, \ldots, X_n) = \mathop{\&}_{f(\sigma_1, \ldots, \sigma_n) = 0} \overline{X_1^{\sigma_1}} \vee \ldots \vee \overline{X_n^{\sigma_n}},$$

СКНФ

Каждая функция алгебры логики $f(X_1, \ldots, X_n)$, не равная тождественно 1, реализуется следующей СКНФ, причём единственной:

$$f(X_1, \ldots, X_n) = \mathop{\&}_{f(\sigma_1, \ldots, \sigma_n)=0} \overline{X_1^{\sigma_1}} \vee \ldots \vee \overline{X_n^{\sigma_n}},$$

Конъюнкция берётся по тем двоичным наборам $(\sigma_1, \ldots, \sigma_n)$, на которых функция f равна 0.

СКНФ

Каждая функция алгебры логики $f(X_1, \ldots, X_n)$, не равная тождественно 1, реализуется следующей СКНФ, причём единственной:

$$f(X_1, \ldots, X_n) = \mathop{\&}_{f(\sigma_1, \ldots, \sigma_n) = 0} \overline{X_1^{\sigma_1}} \vee \ldots \vee \overline{X_n^{\sigma_n}},$$

Конъюнкция берётся по тем двоичным наборам $(\sigma_1, \ldots, \sigma_n)$, на которых функция f равна 0.

Построение СКНФ с помощью таблицы истинности

СКНФ

Каждая функция алгебры логики $f(X_1, \ldots, X_n)$, не равная тождественно 1, реализуется следующей СКНФ, причём единственной:

$$f(X_1, \ldots, X_n) = \mathop{\&}_{f(\sigma_1, \ldots, \sigma_n) = 0} \overline{X_1^{\sigma_1}} \vee \ldots \vee \overline{X_n^{\sigma_n}},$$

Конъюнкция берётся по тем двоичным наборам $(\sigma_1, \ldots, \sigma_n)$, на которых функция f равна 0.

Построение СКНФ с помощью таблицы истинности

1. В таблице истинности отыскиваются строки, в которых функция f равна 0.

СКНФ

Каждая функция алгебры логики $f(X_1, \ldots, X_n)$, не равная тождественно 1, реализуется следующей СКНФ, причём единственной:

$$f(X_1, \ldots, X_n) = \mathop{\&}_{f(\sigma_1, \ldots, \sigma_n)=0} \overline{X_1^{\sigma_1}} \vee \ldots \vee \overline{X_n^{\sigma_n}},$$

Конъюнкция берётся по тем двоичным наборам $(\sigma_1, \ldots, \sigma_n)$, на которых функция f равна 0.

Построение СКНФ с помощью таблицы истинности

1. В таблице истинности отыскиваются строки, в которых функция f равна 0.
2. Каждой такой строке соответствует дизъюнкт, состоящий из переменных, причём переменная входит в него под отрицанием, если в рассматриваемой строке значение данной переменной равно 1, и без отрицания, если её значение равно 0.

Пример нахождения СКНФ и СДНФ

Формула $(X \mathbin{\&} Y) \leftrightarrow \overline{Z}$ реализует функцию $h(X, Y, Z)$:

X	Y	Z	$h(X, Y, Z)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Пример нахождения СКНФ и СДНФ

Формула $(X \,\&\, Y) \leftrightarrow \overline{Z}$ реализует функцию $h(X, Y, Z)$:

X	Y	Z	$h(X, Y, Z)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

СДНФ этой формулы:

$$(\overline{X} \,\&\, \overline{Y} \,\&\, Z) \vee (\overline{X} \,\&\, Y \,\&\, Z) \vee (X \,\&\, \overline{Y} \,\&\, Z) \vee (X \,\&\, Y \,\&\, \overline{Z}).$$

Пример нахождения СКНФ и СДНФ

Формула $(X \mathbin{\&} Y) \leftrightarrow \overline{Z}$ реализует функцию $h(X, Y, Z)$:

X	Y	Z	$h(X, Y, Z)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

СДНФ этой формулы:

$$(\overline{X} \mathbin{\&} \overline{Y} \mathbin{\&} Z) \vee (\overline{X} \mathbin{\&} Y \mathbin{\&} Z) \vee (X \mathbin{\&} \overline{Y} \mathbin{\&} Z) \vee (X \mathbin{\&} Y \mathbin{\&} \overline{Z}).$$

СКНФ этой формулы:

$$(X \vee Y \vee Z) \mathbin{\&} (X \vee \overline{Y} \vee Z) \mathbin{\&} (\overline{X} \vee Y \vee Z) \mathbin{\&} (\overline{X} \vee \overline{Y} \vee \overline{Z}).$$

Алгоритм получения СДНФ/СКНФ с использованием эквивалентных преобразований

Алгоритм получения СДНФ/СКНФ с использованием эквивалентных преобразований

1. Для формулы $\mathfrak{A}$ получить любую ДНФ (КНФ).

Алгоритм получения СДНФ/СКНФ с использованием эквивалентных преобразований

1. Для формулы $\mathfrak{A}$ получить любую ДНФ (КНФ).
2. Если в ДНФ (КНФ) есть конъюнкт (дизъюнкт) $\mathfrak{B}$, не содержащий переменную X_i, то необходимо её добавить, используя правила расщепления: $\mathfrak{B} \equiv (\mathfrak{B} \,\&\, X_i) \vee (\mathfrak{B} \,\&\, \overline{X_i})$ — для ДНФ, $\mathfrak{B} \equiv (\mathfrak{B} \vee X_i) \,\&\, (\mathfrak{B} \vee \overline{X_i})$ — для КНФ.

Алгоритм получения СДНФ/СКНФ с использованием эквивалентных преобразований

1. Для формулы $\mathfrak{A}$ получить любую ДНФ (КНФ).
2. Если в ДНФ (КНФ) есть конъюнкт (дизъюнкт) $\mathfrak{B}$, не содержащий переменную X_i, то необходимо её добавить, используя правила расщепления: $\mathfrak{B} \equiv (\mathfrak{B} \mathbin{\&} X_i) \vee (\mathfrak{B} \mathbin{\&} \overline{X_i})$ — для ДНФ,
   $\mathfrak{B} \equiv (\mathfrak{B} \vee X_i) \mathbin{\&} (\mathfrak{B} \vee \overline{X_i})$ — для КНФ.
3. Если в ДНФ (КНФ) встречаются равные конъюнкты (дизъюнкты), то лишние нужно отбросить.

Алгоритм получения СДНФ/СКНФ с использованием эквивалентных преобразований

1. Для формулы $\mathfrak{A}$ получить любую ДНФ (КНФ).
2. Если в ДНФ (КНФ) есть конъюнкт (дизъюнкт) $\mathfrak{B}$, не содержащий переменную X_i, то необходимо её добавить, используя правила расщепления: $\mathfrak{B} \equiv (\mathfrak{B} \mathbin{\&} X_i) \vee (\mathfrak{B} \mathbin{\&} \overline{X_i})$ — для ДНФ, $\mathfrak{B} \equiv (\mathfrak{B} \vee X_i) \mathbin{\&} (\mathfrak{B} \vee \overline{X_i})$ — для КНФ.
3. Если в ДНФ (КНФ) встречаются равные конъюнкты (дизъюнкты), то лишние нужно отбросить.
4. Если в ДНФ (КНФ) в некотором конъюнкте (дизъюнкте) литерал X_i встречается несколько раз, то лишние нужно отбросить.

   Если в ДНФ (КНФ) в некотором конъюнкте (дизъюнкте) литерал $\overline{X_i}$ встречается несколько раз, то лишние нужно отбросить.

Алгоритм получения СДНФ/СКНФ с использованием эквивалентных преобразований

1. Для формулы $\mathfrak{A}$ получить любую ДНФ (КНФ).
2. Если в ДНФ (КНФ) есть конъюнкт (дизъюнкт) $\mathfrak{B}$, не содержащий переменную X_i, то необходимо её добавить, используя правила расщепления: $\mathfrak{B} \equiv (\mathfrak{B} \,\&\, X_i) \vee (\mathfrak{B} \,\&\, \overline{X_i})$ — для ДНФ, $\mathfrak{B} \equiv (\mathfrak{B} \vee X_i) \,\&\, (\mathfrak{B} \vee \overline{X_i})$ — для КНФ.
3. Если в ДНФ (КНФ) встречаются равные конъюнкты (дизъюнкты), то лишние нужно отбросить.
4. Если в ДНФ (КНФ) в некотором конъюнкте (дизъюнкте) литерал X_i встречается несколько раз, то лишние нужно отбросить.

   Если в ДНФ (КНФ) в некотором конъюнкте (дизъюнкте) литерал $\overline{X_i}$ встречается несколько раз, то лишние нужно отбросить.
5. Если в ДНФ некоторый конъюнкт содержит конъюнкцию $X_i \,\&\, \overline{X_i}$, то данный конъюнкт нужно отбросить.

   Если в КНФ некоторый дизъюнкт содержит дизъюнкцию $X_i \vee \overline{X_i}$, то данный дизъюнкт нужно отбросить.

Пример получения СКНФ

Получить СКНФ формулы $(A \vee C) \,\&\, (\overline{B} \vee C)$.

Пример получения СКНФ

Получить СКНФ формулы $(A \vee C) \mathbin{\&} (\overline{B} \vee C)$.

Решение

$$
\begin{aligned}
(A \vee C) \mathbin{\&} (\overline{B} \vee C) &\equiv (A \vee C \vee B) \mathbin{\&} (A \vee C \vee \overline{B}) \mathbin{\&} (\overline{B} \vee C) \equiv \\
&\equiv (A \vee C \vee B) \mathbin{\&} (A \vee C \vee \overline{B}) \mathbin{\&} (\overline{B} \vee C \vee A) \mathbin{\&} (\overline{B} \vee C \vee \overline{A}) \equiv \\
&\equiv (A \vee B \vee C) \mathbin{\&} (A \vee \overline{B} \vee C) \mathbin{\&} (\overline{A} \vee \overline{B} \vee C).
\end{aligned}
$$

Ответ: $(A \vee B \vee C) \mathbin{\&} (A \vee \overline{B} \vee C) \mathbin{\&} (\overline{A} \vee \overline{B} \vee C)$.