

- Функция $f : \{0, 1\}^n \rightarrow \{0, 1\}$ — **n -местная** (n -арная) **булева функция**
 - будем писать $f(x_1, \dots, x_n)$ или $f(\vec{x})$, если n известно или несущественно
 - также принято сокращать слова «булева функция» до б.ф.
- n -местная б.ф. f переводит строки из n бит в битовые значения, то есть
 - ★ задает **n -местную операцию** на множестве $\{0, 1\}$
 - ★ вычисляет **n -местный предикат** на множестве $\{0, 1\}$
 - ★ задает **n -местное отношение** на множестве $\{0, 1\}$
 - ★ распознает язык $L_f \subseteq \{0, 1\}^n$
- Прямолинейный (неэкономичный) способ задания б.ф. — таблица значений
 - также называемая **таблицей истинности**
 - ★ n -местную б.ф. f можно задать битовой строкой $F[0..2^n-1]$, где $F[i]$ — значение f на строке, являющейся двоичной записью числа i
- **Пример:** рассмотрим бинарные б.ф. $f(x, y)$, они же строки $F[0..3]$

x, y	0	\wedge	$>$	x	$<$	y	$+$	\vee	\downarrow	\sim	\bar{y}	\leftarrow	\bar{x}	\rightarrow	$'$	1
00	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
01	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
10	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
11	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

- $x \downarrow y = \overline{x \vee y}$ — **стрелка Пирса**, $x' y = \overline{x \wedge y}$ — **штрих Шефера**
- $x > y, x < y$ — обычные бинарные отношения,
- $x + y$ — сложение по модулю 2 (xor), $x \sim y = [x = y]$ — **эквиваленция**
- $x \rightarrow y, x \leftarrow y$ — прямая и обратная **импликации**

Примеры многоместных булевых функций

Есть несколько важных серий n -местных булевых функций для произвольного n

- $\text{PROJ}_i(\vec{x}) = x_i$ — проекции
 - ★ проекция вытаскивает нужный бит из вектора аргументов
 - чтобы вычислить $(a + b) \bmod 2$, где числа a и b заданы двоичными представлениями, нужно взять по младшему биту из a и b и выполнить xor
- $\bigoplus \vec{x}, \bigvee \vec{x}, \bigwedge \vec{x}$ — n -арные аналоги коммутативных бинарных функций
- $\text{MOD}_p(\vec{x}) = [x_1 + \dots + x_n \text{ делится на } p]$ — модулярные функции
 - ★ важный частный случай — MOD_2 (или PARITY) — четность числа единиц
 - ! постройте ДКА, который вычисляет функцию MOD_p
- $T_i(\vec{x}) = [x_1 + \dots + x_n \geq i]$ — пороговые функции
 - ★ $T_1(\vec{x}) = \bigvee \vec{x}, T_n(\vec{x}) = \bigwedge \vec{x}$
 - ★ $T_{\lfloor n/2 \rfloor + 1}(\vec{x})$ — функции большинства (или голосования)
 - ★ пороговая функция — ключевой элемент персептрона (простейшей нейронной сети)
- ★ Функция $f(x_1, \dots, x_n)$ может зависеть от значений только части переменных
 - ★ например, в списке бинарных функций есть $x, \bar{x}, y, \bar{y}, 0, 1$
 - переменная x_i функции f — фиктивная, если для любых $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$
 $f(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) = f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)$
 - переменная, не являющаяся фиктивной, называется существенной

- ★ Функции n переменных громоздко задавать таблицами; что делать?
- Сложные функции можно представлять как **суперпозицию** более простых
 - ★ рассматриваются функции нескольких переменных \Rightarrow одни и те же функции могут образовывать много различных суперпозиций
 - $g(y_1, \dots, y_k)$ можно подставить в $f(x_1, \dots, x_n)$ вместо любого аргумента
 - какие-то из аргументов g можно отождествить с какими-то из аргументов f
 - **пример**: подстановкой $g(z, t) = z \wedge t$ в $f(x, y) = x \rightarrow y$ можно получить $(z \wedge t) \rightarrow y$, $(z \wedge y) \rightarrow y$, $x \rightarrow (z \wedge t)$, $x \rightarrow (x \wedge t)$
- Удобный способ записи булевых функций:
 - зафиксировать маленький набор функций B , смотреть на функции из B как на операции
 - записать **булеву формулу** — формальное выражение, содержащее переменные, символы операций из B и скобки
 - вычислять значение функции для каждого вектора значений переменных, выполняя операции
 - ★ булева формула = слово над конечным алфавитом, которое удовлетворяет набору ограничений (**синтаксис**)
 - ★ булева формула **задает** булеву функцию (**семантика**)
- ★ Разные формулы могут задавать одну и ту же функцию
 - **пример**: $(x \vee y) \wedge z$ и $(x \wedge z) \vee (y \wedge z)$
 - формулы, задающие одну и ту же функцию, называются **эквивалентными**
 - **тавтология** — это формула, задающая константу 1
 - **противоречие** — это формула, задающая константу 0
 - формула, задающая функцию, отличную от константы 0, называется **выполнимой**

Дизъюнктивные нормальные формы

- Разные формулы задают одну функцию \Rightarrow нужны «канонические» формулы
- Такие формулы называют **нормальными формами**
- Нормальная форма должна
 - существовать для любой функции
 - эффективно вычисляться (например по таблице истинности)
 - быть удобной для хранения и вычислений
- **Литерал** — это формула вида x или \bar{x} , где x — переменная
- **Дизъюнктивная нормальная форма (ДНФ)** — это формула вида $\bigvee_{i=1}^n F_i$, $n \geq 1$
 - где $F_i = \bigwedge_{j=1}^{k_i} L_{ij}$, L_{ij} — литерал, $k_i \geq 1$
 - F_i называют **элементарными конъюнкциями**
 - ★ ДНФ — **иерархическая** формула: отрицание применяется только к переменным, конъюнкция — к литералам, дизъюнкция — к элементарным конъюнкциям
- ДНФ от k переменных называется **совершенной (k -СДНФ)**, если
 - все элементарные конъюнкции F_i различны
 - каждая F_i состоит из k литералов, соответствующих различным переменным

Теорема

Любая булева функция, не равная константе 0, задается некоторой СДНФ.

- ★ Иными словами, любая выполнимая формула эквивалентна СДНФ
- ★ **Следствие:** любая булева функция задается некоторой ДНФ
 - $x \wedge \bar{x}$ — ДНФ, задающая 0

Дизъюнктивные нормальные формы (2)

- Функцию $x \sim y$ иногда удобно записывать как x^y
 - ★ свойства: $x^1 = x$, $x^0 = \bar{x}$, $1^y = y$, $0^y = \bar{y}$
- Доказательство теоремы об СДНФ:
 - пусть $f(x_1, \dots, x_k)$ отлична от константы 0
 - построим СДНФ F по следующему правилу:
 - ★ для каждого битового вектора $\vec{b} = (b_1, \dots, b_k)$ такого, что $f(b_1, \dots, b_k) = 1$, поместим в F элементарную конъюнкцию $C_{\vec{b}} = x_1^{b_1} \wedge \dots \wedge x_k^{b_k}$
 - построенная формула — СДНФ по определению
 - докажем, что F задает f
 - ★ функция, заданная ДНФ, равна 1 \Leftrightarrow одна из элементарных конъюнкций равна 1
 - ★ $f(b_1, \dots, b_k) = 1 \Leftrightarrow C_{\vec{b}}(b_1, \dots, b_k) = 1$ □

Пример построения СДНФ:

x_1, x_2, x_3	f
000	1
001	0
010	0
011	1
100	0
101	1
110	0
111	0

$\Rightarrow F = (\bar{x}_1 \wedge \bar{x}_2 \wedge \bar{x}_3) \vee (\bar{x}_1 \wedge x_2 \wedge x_3) \vee (x_1 \wedge \bar{x}_2 \wedge x_3)$

- ★ СДНФ — очень громоздкая формула
 - если у функции половина значений — единицы, то k -СДНФ состоит из $k \cdot 2^{k-1}$ литералов
 - ★ проще хранить 2^k бит таблицы значений
- ★ важная задача — построение **кратчайшей** ДНФ для данной функции
 - к сожалению, эта задача не только важная, но и **трудная**
- ★ Для б.ф. 2-4 переменных кратчайшую ДНФ строят при помощи **карт Карно**
- Карта Карно функции f — это специальная запись таблицы значений f
 - карта — это прямоугольная таблица из 2^k клеток, где k — арность f
 - строки и столбцы проиндексированы так, что каждой клетке однозначно соответствует набор значений переменных, в клетке пишется значение функции (обычно пишут только единицы)
 - ★ кратчайшую ДНФ строят, покрывая все клетки с единицами прямоугольниками такими, что
 - число клеток в прямоугольнике — степень двойки
 - если клеток 2^i , то $k - i$ переменных принимают в этих клетках одно значение (определяют элементарную конъюнкцию), а остальные i — все наборы значений

пример:

$$\begin{array}{c}
 \begin{array}{cc}
 & y & \bar{y} \\
 x & \boxed{1} & \boxed{1} \\
 x & \boxed{1} & \boxed{1} \\
 \bar{x} & \boxed{1} & \\
 \bar{x} & &
 \end{array}
 \begin{array}{c}
 z \\
 \bar{z} \\
 \bar{z} \\
 z
 \end{array}
 \end{array}
 \implies (x \wedge \bar{z}) \vee (y \wedge \bar{z}) \vee (x \wedge \bar{y}) \implies (y \wedge \bar{z}) \vee (x \wedge \bar{y})$$

Карты Карно (2)

- ★ Еще одна проблема с кратчайшей ДНФ — она не единственна
- Вот иллюстрация из Википедии с картами Карно для 4 переменных:

AB \ CD		AB			
		00	01	11	10
CD	10	1	1		
	11		1	1	
	01			1	1
	00	1			1

AB \ CD		AB			
		00	01	11	10
CD	10	1	1		
	11		1	1	
	01			1	1
	00	1			1

$$\begin{aligned}(\bar{A} \wedge \bar{B} \wedge \bar{D}) \vee (\bar{A} \wedge B \wedge C) \vee (A \wedge B \wedge D) \vee (A \wedge \bar{B} \wedge \bar{C}) = \\ (\bar{A} \wedge C \wedge D) \vee (B \wedge C \wedge D) \vee (A \wedge \bar{C} \wedge D) \vee (\bar{B} \wedge \bar{C} \wedge \bar{D})\end{aligned}$$

- Конъюнктивная нормальная форма (КНФ) — это формула вида $\bigwedge_{i=1}^n F_i$, $n \geq 1$
 - где $F_i = \bigvee_{j=1}^{k_i} L_{ij}$, L_{ij} — литерал, $k_i \geq 1$
 - F_i называют **элементарными дизъюнкциями** или **клозами** (clause)
 - ★ КНФ — тоже **иерархическая** формула: отрицание применяется только к переменным, дизъюнкция — к литералам, конъюнкция — к клозам
- КНФ от k переменных называется **совершенной (k -СКНФ)**, если
 - все клозы F_i различны
 - каждый F_i состоит из k литералов, соответствующих различным переменным

Теорема

Любая булева функция, не равная константе 1, задается некоторой СКНФ.

- ★ Иными словами, любая не-тавтология эквивалентна СКНФ
- ★ **Следствие:** любая булева функция задается некоторой КНФ
 - $x \vee \bar{x}$ — КНФ, задающая 1

... симметрично доказательству для СДНФ

• **Доказательство:**

- пусть $f(x_1, \dots, x_k)$ отлична от константы 1
- построим СКНФ F по следующему правилу:
 - ★ для каждого битового вектора $\vec{b} = (b_1, \dots, b_k)$ такого, что $f(b_1, \dots, b_k) = 0$, поместим в F элементарную дизъюнкцию $D_{\vec{b}} = x_1^{\bar{b}_1} \vee \dots \vee x_k^{\bar{b}_k}$
 - построенная формула — СКНФ по определению
 - докажем, что F задает f
 - ★ функция, заданная КНФ, равна 0 \Leftrightarrow одна из элементарных дизъюнкций равна 0
 - ★ $f(b_1, \dots, b_k) = 0 \Leftrightarrow D_{\vec{b}}(b_1, \dots, b_k) = 0$



Пример построения СКНФ (функция — та же, что и для примера с СДНФ):

x_1, x_2, x_3	f
000	1
001	0
010	0
011	1
100	0
101	1
110	0
111	0

$$\Rightarrow F = (x_1 \vee x_2 \vee \bar{x}_3) \wedge (x_1 \vee \bar{x}_2 \vee x_3) \wedge (\bar{x}_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_3)$$

- ★ Симметрия распространяется и на оптимизацию КНФ при помощи карт Карно: прямоугольники соответствуют клозам и покрывают множество нулей