Алгебра высказываний

1. Минимизация булевых функций с использованием законов булевой алгебры

Основные законы булевой алгебры.

Здесь f1, f2, f3 – любые булевы функции.

1. *Ассоциативность:*

   $f_1\&(f_2\&f_3) = (f_1\&f_2)\&f_3 = f_1\&f_2\&f_3$ $\qquad$ $f_1\vee(f_2\vee f_3) = (f_1\vee f_2)\vee f_3 = f_1\vee f_2\vee f_3$
2. *Коммутативность:*

   $f_1\&f_2 = f_2\&f_1$ $\qquad$ $f_1\vee f_2 = f_2\vee f_1$
3. *Дистрибутивность*

   конъюнкции относительно дизъюнкции: $f_1\&(f_2\vee f_3) = (f_1\&f_2)\vee(f_1\&f_3)$

   дизъюнкции относительно конъюнкции: $f_1\vee(f_2\&f_3) = (f_1\vee f_2)\&(f_1\vee f_3)$
4. *Идемпотентность* $\quad f_1\&f_1 = f_1 \qquad f_1\vee f_1 = f_1$
5. *Двойное отрицание:* $\quad \overline{\overline{f_1}} = f_1$
6. *Свойства констант:* $\quad f_1\&1 = f_1 \qquad f_1\vee 1 = 1 \qquad \overline{0} = 1$

   $f_1\&0 = 0 \qquad f_1\vee 0 = f_1 \qquad \overline{1} = 0$
7. *Законы де Моргана:* $\quad \overline{f_1\&f_2} = \overline{f_1}\vee\overline{f_2} \qquad \overline{f_1\vee f_2} = \overline{f_1}\&\overline{f_2}$
8. *Закон противоречия:* $\quad f_1\&\overline{f_1} = 0$
9. *Закон «исключенного третьего»:* $\quad f_1\vee\overline{f_1} = 1$

На основе этих законов выводятся свойства, которые используют, упрощая формулы:

10. Поглощение:

Доказательства свойств:

1) $f_1\vee f_1\&f_2 = f_1$

$f_1\vee f_1\&f_2 \overset{[6]}{=} f_1\&1\vee f_1\&f_2 \overset{[3]}{=} f_1\&(1\vee f_2) \overset{[1,6]}{=} f_1\&1 \overset{[6]}{=} f_1.$

2) $f_1\&(f_1\vee f_2) = f_1$

$f_1\&(f_1\vee f_2) \overset{[3]}{=} f_1\&f_1\vee f_1\&f_2 \overset{[4]}{=} f_1\vee f_1\&f_2 \overset{[10(1)]}{=} f_1.$

11. Склеивание:

1) $f_1\&f_2\vee f_1\&\overline{f_2} = f_1$

$f_1\&f_2\vee f_1\&\overline{f_2} \overset{[3]}{=} f_1\&(f_2\vee\overline{f_2}) \overset{[9]}{=} f_1\&1 \overset{[6]}{=} f_1.$

2) $f_1\vee\overline{f_1}\&f_2 = f_1\vee f_2$

$f_1\vee\overline{f_1}\&f_2 \overset{[3]}{=} (f_1\vee\overline{f_1})\&(f_1\vee f_2) \overset{[9]}{=} 1\&(f_1\vee f_2) \overset{[6]}{=} f_1\vee f_2.$

3) $f_1\&f_2\vee f_1\&f_3\vee f_2\&\overline{f_3} = f_1\&f_3\vee f_2\&\overline{f_3}$

$f_1\&f_2\vee f_1\&f_3\vee f_2\&\overline{f_3} \overset{[3]}{=}$

$\overset{[3]}{=} f_1\&f_2\,(f_3\vee\overline{f_3})\vee f_1\&f_3\vee f_2\&\overline{f_3} \overset{[3]}{=}$

$\overset{[3]}{=} f_1\&f_2\&f_3\vee f_1\&f_2\&\overline{f_3}\vee f_1\&f_3\vee f_2\&\overline{f_3} \overset{[3]}{=}$

$\overset{[3]}{=} f_1\&f_3\&f_2\vee f_1\&f_3\vee f_1\&f_2\&\overline{f_3}\vee f_2\&\overline{f_3} \overset{[3]}{=}$

$\overset{[3]}{=} f_1\&f_3\vee f_2\&\overline{f_3}.$

Приведенные законы и свойства используются при упрощении формул, т. е. для получения эквивалентных формул, содержащих меньшее количество символов.

Порядок действий определяется скобками и следующим **приоритетом операций**: отрицание; конъюнкция; дизъюнкция.

Пример. Максимально упростим выражение

$$(b\&\neg d)\vee((c\vee a)\&(a\vee\neg c)\&(a\&b)\&(\neg b\vee a))\vee(b\&d).$$

а) Выделим в данном выражении подвыражения, соединенные одинаковым знаком (знаком дизъюнкции):

$$(\overbrace{b\&\neg d}^{\text{①}})\vee\overbrace{((c\vee a)\&(a\vee\neg c)\&(a\&b)\&(\neg b\vee a))}^{\text{②}}\vee(\overbrace{b\&d}^{\text{③}}).$$

По закону 2 коммутативности дизъюнкции, выражения ① и ② можно поменять местами:

$$\overbrace{((c\vee a)\&(a\vee\neg c)\&(a\&b)\&(\neg b\vee a))}^{②}\vee\overbrace{(b\&\neg d)}^{①}\vee\overbrace{(b\&d)}^{③}.$$

По закону (1) ассоциативности дизъюнкции, порядок выполнения операций $\vee$ над выражениями ①, ②, ③ отрегулируем скобками:

$$\overbrace{((c\vee a)\&(a\vee\neg c)\&(a\&b)\&(\neg b\vee a))}^{②}\vee((\overbrace{b\&\neg d}^{①})\vee(\overbrace{b\&d}^{③})).$$

Упростим сначала подвыражение $((b\&\neg d)\vee(b\&d))$. По закону (3) дистрибутивности конъюнкции относительно дизъюнкции, вынесем b за скобки:

$$((b\&\neg d)\vee(b\&d)) = b\&(\neg d\vee d).$$

По закону «исключенного третьего» (9), $(\neg d\vee d) = 1$. Тогда

$$((b\&\neg d)\vee(b\&d)) = b\&(\neg d\vee d) = b\&1 = (\text{по свойствам констант (6)}) = b.$$

Этот же результат можно получить проще – по свойству склеивания (11):

$$((b\&\neg d)\vee(b\&d)) = b.$$

Упростим теперь подвыражение ②. Так как выражения $(c\vee a)$, $(a\vee\neg c)$, $(a\&b)$, $(\neg b\vee a)$ соединены знаком конъюнкции, по закону (1) ассоциативности, расставим скобки, которые определят дальнейший порядок действий:

$$((c\vee a)\&(a\vee\neg c))\ \&\ ((a\&b)\&(\neg b\vee a)).$$

$(c\vee a)\&(a\vee\neg c) \overset{(2)}{=} (a\vee c)\&(a\vee\neg c) \overset{(3)}{=} a\vee(c\,\&\,\neg c) \overset{(8)}{=} a\vee 0 \overset{(6)}{=} a.$

$(a\&b)\&(\neg b\vee a) \overset{(2)}{=} (a\&b)\&(a\vee\neg b) \overset{(3)}{=} a\&b\&a \vee a\&b\&\neg b \overset{(2),(1)}{=} a\&a\&b \vee a\&(b\&\neg b) \overset{(1),(8)}{=}$ $(a\&a)\&b \vee a\&0 \overset{(4),(6)}{=} a\&b \vee 0 \overset{(6)}{=} a\&b.$

Или, по-другому, используя свойство поглощения (10):

$$(a\&b)\&(\neg b\vee a) \overset{(2)}{=} (b\&a)\&(a\vee\neg b) \overset{(1)}{=} b\&(a\&(a\vee\neg b)) \overset{(10)}{=} b\&a \overset{(2)}{=} a\&b.$$

Итак, подвыражение ② запишется в виде: $(a)\,\&\,(a\&b) \overset{(1)}{=} (a\&a)\,\&\,b \overset{(4)}{=} a\&b.$

Тогда исходное выражение сведется к такому представлению:

$$a\&b\vee b \overset{(2)}{=} b\vee(a\&b) \overset{(6)}{=} (b\&1)\vee(b\&a) \overset{(3)}{=} b\&(1\vee a) \overset{(6)}{=} b\&1 \overset{(6)}{=} b.$$

Этот же результат проще получить, используя свойство поглощения:

$$a\&b\vee b \overset{(2)}{=} b\vee(a\&b) \overset{(10)}{=} b.$$

Пример. Максимально упростим выражение

$$\neg(\neg c\vee a)\vee\neg((c\&\neg a)\&(\neg a\&d)\vee d\vee(d\&c))\,\&(\neg c\vee\neg d)\vee d.$$

а) $\neg(\neg c\vee a)\vee\neg((c\&\neg a)\&(\neg a\&d)\vee d\vee(d\&c))\,\&(\neg c\vee\neg d)\vee d \overset{(9)}{=} (\neg\neg c\&\neg a)\vee$

$\vee\neg((c\&\neg a)\&(\neg a\&d)\vee d\vee(d\&c))\,\&(\neg c\vee\neg d)\vee d \overset{(5)}{=} c\&\neg a\vee\neg((c\&\neg a)\&(\neg a\&d)\vee d\vee(d\&c))\,\&$

$\&(\neg c\vee\neg d)\vee d \overset{(1),(2)}{=} c\&\neg a\vee\neg((\neg a\&\neg a)\&(c\&d)\vee d\vee(d\&c))\,\&(\neg c\vee\neg d)\vee d \overset{(4)}{=} c\&\neg a\vee$

$\vee\neg(\neg a\&c\&d\vee d\vee d\&c)\,\&(\neg c\vee\neg d)\vee d \overset{(2)}{=} c\&\neg a\vee\neg((d\vee d\&\neg a\&c)\vee(d\&c))\,\&(\neg c\vee\neg d)\vee d=$

$\overset{(11)}{=} c\&\neg a\vee\neg(d\vee(d\&c))\,\&(\neg c\vee\neg d)\vee d \overset{(11)}{=} c\&\neg a\vee\neg(d)\&(\neg c\vee\neg d)\vee d \overset{(11)}{=} {=}c\&\neg a\vee$

$\vee\neg d\vee d \overset{(1)}{=} c\&\neg a\vee(\neg d\vee d) \overset{(9)}{=} c\&\neg a\vee 1 \overset{(6)}{=} 1.$

2. Построение таблицы значений булевой функции

Будем рассматривать множество B = {0,1}, элементы которого являются формальными символами, не имеющими арифметического смысла (0 – ЛОЖЬ, 1 – ИСТИНА).

Булевой (логической) функцией от n переменных называется функция $f(x_1,x_2,\ldots,x_n)$, которая вместе со своими аргументами $x_1, x_2,\ldots, x_n$ принимает значения из множества B.

Булевы функции двух переменных:

x	y	ϕ_0	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7	ϕ_8	ϕ_9	ϕ_{10}	ϕ_{11}	ϕ_{12}	ϕ_{13}	ϕ_{14}	ϕ_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
		0	&	$\not\rightarrow$	x	$\not\leftarrow$	y	$\oplus$	$\vee$	$\downarrow$	~	$\bar{y}$	$\leftarrow$	$\bar{x}$	$\rightarrow$	|	1

функция	название	функция	название
$\phi_0(x, y) \equiv 0$	константа нуля	$\phi_{13}(x, y) = x \rightarrow y$	левая импликация
$\phi_{12}(x, y) = \bar{x} = \neg x$	отрицание x	$\phi_8(x, y) = x \downarrow y$	стрелка Пирса
$\phi_1(x, y) = x \,\&\, y = x \wedge y$	конъюнкция	$\phi_{14}(x, y) = x \mid y$	штрих Шеффера
		$\phi_{15}(x, y) \equiv 1$	константа единицы
$\phi_7(x, y) = x \vee y$	дизъюнкция	$\phi_2(x, y) = x \not\rightarrow y$	левая коимпликация
$\phi_6(x, y) = x \oplus y$	сложение по модулю 2	$\phi_4(x, y) = x \leftarrow y$	правая импликация
$\phi_9(x, y) = x \sim y$	эквиваленция	$\phi_{11}(x, y) = x \not\leftarrow y$	правая коимпликация

Пример. Построим таблицу значений для $f(x,y,z) = \neg(x\rightarrow\neg y) \downarrow (z \oplus (z\&y \sim \neg x))$.

Знание особенностей логических функций 2х переменных позволит ускорить процесс построения таблицы значений заданной функции и проверку результата. Приведем ряд рассуждений (их легко получить, анализируя таблицы значений ф-ций 2х переменных):

- если первый аргумент функции $\phi_{13}(x,y) = x\rightarrow y$ равен 0, то, независимо от значения второго аргумента, значение функции $x\rightarrow y$ равно 1;
- если хотя бы один из аргументов функции $\phi_1(x, y)= x\&y$ равен 0, то, независимо от значения другого аргумента, значение функции x&y равно 0;
- если хотя бы один из аргументов функции $\phi_8(x,y)= x\downarrow y$ равен 1, то, независимо от значения другого аргумента, значение функции $x\downarrow y$ равно 0.

Строим таблицу значений данной функции (здесь указан порядок выполнения действий):

В таких таблицах значения аргументов (x, y, z) (наборы из нулей и единиц) всегда расположены в определенном порядке – <u>*лексикографическом*</u>, совпадающим с порядком возрастания наборов, рассматриваемых как двоичные числа. Этот порядок не изменяется.

десятичные эквиваленты двоичных наборов:

				③	②	①	⑧	⑦	⑤	⑥	④
	x	y	z	¬	(x→	¬y)	↓	(z ⊕	(z&y	~	¬x))
0	0	0	0	0	1	1	1	0	0	0	1
1	0	0	1	0	1	1	0	1	0	0	1
2	0	1	0	0	1	0	1	0	0	0	1
3	0	1	1	0	1	0	1	0	1	1	1
4	1	0	0	0	1	1	0	1	0	1	0
5	1	0	1	0	1	1	1	0	0	1	0
6	1	1	0	1	0	0	0	1	0	1	0
7	1	1	1	1	0	0	0	1	1	0	0

Т.к. порядок расположения наборов в таблице всегда одинаков, при задании функции ее значениями для сокращения записи столбец со значениями функции транспонируется и записывается в виде строки.

Например, рассмотренная выше функция задается так: f(x,y,z) = (1011 0100).

Другие варианты построения таблиц функций трех и четырех переменных:

f(x,y,z):

xy \ z	0	1
00	f(0,0,0)	f(0,0,1)
01	f(0,1,0)	f(0,1,1)
10	f(1,0,0)	f(1,0,1)
11	f(1,1,0)	f(1,1,1)

f(x,y,z,t):

xy \ zt	00	01	10	11
00	f(0,0,0,0)	f(0,0,0,1)	f(0,0,1,0)	f(0,0,1,1)
01	f(0,1,0,0)	f(0,1,0,1)	f(0,1,1,0)	f(0,1,1,1)
10	f(1,0,0,0)	f(1,0,0,1)	f(1,0,1,0)	f(1,0,1,1)
11	f(1,1,0,0)	f(1,1,0,1)	f(1,1,1,0)	f(1,1,1,1)

Если выражение, задающее функцию, представить в нормальной форме (дизъюнктивной или конъюнктивной), то построение таблицы значений функции упростится.

Пример. Построим таблицу значений заданной функции f(a,b,c,d) с учетом ее представления в виде минимальной ДНФ.

$f(a,b,c,d)=((\neg c\vee d)\&(d\vee a))\vee((a\vee\neg b)\&(\neg c\vee a)\&\neg(c\vee d)\&(\neg d\vee a))\vee a\&(b\vee\neg a)=d\vee\neg b\&\neg c\vee a\&b.$

	a	b	c	d	f(a,b,c,d)
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
1	1	1	1	1	1

Эта таблица заполняется следующим образом. По выражению, задающему функцию f(a,b,c,d), определяем, на каких наборах ее значение равно 1.

Так как функция представлена в виде ДНФ, то, по свойству констант ($f_1\vee 1 = 1$), для того чтобы значение всего выражения, задающего функцию, равнялось 1, достаточно, чтобы хотя бы одно из его дизъюнктивных слагаемых имело значение, равное 1. Выражение $d\vee\neg b\&\neg c\vee a\&b$ имеет 3 дизъюнктивных слагаемых: d, $\neg b\&\neg c$ и a&b. Значит, f(a,b,c,d) =1 только в следующих случаях:

- либо при d = 1, т.к. тогда $f(a,b,c,d) = 1\vee\underbrace{\neg b\&\neg c\vee a\&b}_{f_1} = 1$,
- либо при $\neg b\&\neg c$ =1, т.к. тогда $f(a,b,c,d) = 1\vee\underbrace{d\vee a\&b}_{f_1} = 1$,
- либо при a&b =1, т.к. тогда $f(a,b,c,d) = 1\vee\underbrace{d\vee\neg b\&\neg c}_{f_1} = 1$.

Определим, на каких наборах эти слагаемые принимают единичные значения.

a&b =1 тогда и только тогда, когда a=1 и b =1 одновременно, т.к. 1&1 =1. При этом переменные c и d (от которых также зависит функция), отсутствующие в этой конъюнкции, могут принимать любые значения. Значит, a&b =1 на наборах **11**00, **11**01, **11**10, **11**11. Аналогично, $\neg b\&\neg c$ =1 на наборах 0**00**0, 0**00**1, 1**00**0, 1**00**1. Для d = 1 наборов будет больше, т.к. для трех остальных переменных a, b и c имеется 8 разных наборов: 000**1**, 001**1**, 010**1**, 011**1**, 100**1**, 101**1**, 110**1**, 111**1**.

На остальных наборах значение заданной функции равно нулю. При заполнении таблицы не обязательно полностью определять наборы, на которых функция принимает единичные значения. Например, в нашем случае достаточно проставить единичные значения функции в тех строках таблицы, где a=1 и b =1 одновременно (12-я, 13-я, 14-я и 15-я строки), где b=0 и c =0 одновременно (0-я, 1-я, 8-я и 9-я строки) и где d = 1 (строки с номерами 1,3,5,7,9,11,13,15).

Другой вариант таблицы:

ab \ cd	00	01	10	11
00	1	1	0	1
01	0	1	0	1
10	1	1	0	1
11	1	1	1	1

3. Получение СДНФ и СКНФ булевых функций

Элементарными конъюнкциями называются конъюнкции переменных или их отрицаний, в которых каждая переменная встречается не более одного раза. **Дизъюнктивной нормальной формой (ДНФ)** формулы называется формула, имеющая вид дизъюнкции элементарных конъюнкций. ДНФ формулы называется совершенной **дизъюнктивной нормальной формой (СДНФ)**, если каждая элементарная конъюнкция ДНФ формулы содержит символы всех переменных, от которых данная функция зависит.

Элементарными дизъюнкциями называются дизъюнкции переменных или их отрицаний, в которых каждая переменная встречается не более одного раза. **Конъюнктивной нормальной формой (КНФ)** формулы называется формула, имеющая вид конъюнкции элементарных дизъюнкций. КНФ формулы называется **совершенной конъюнктивной нормальной формой (СКНФ)**, если каждая элементарная дизъюнкция КНФ формулы содержит символы всех переменных, от которых данная функция зависит.

Пример. Получить СДНФ и СКНФ для функции f(x,y,z), заданной таблицей значений.

x	y	z	f(x,y,z)
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Для построения <u>СДНФ</u> по таблице значений рассмотрим только все те наборы, на которых функция принимает <u>единичные</u> значения. Каждому нулю из набора поставим в соответствие отрицание соответствующей переменной, а единице – саму переменную; таким образом, по каждому набору построим конъюнкцию переменных и их отрицаний:

$\boxed{000\to \bar{x}\ \bar{y}\ \bar{z}}$;$\boxed{010\to \bar{x}\,y\,\bar{z}}$;$\boxed{011\to \bar{x}\,y\,z}$;$\boxed{100\to x\ \bar{y}\ \bar{z}}$;$\boxed{101\to x\,\bar{y}\,z}$;$\boxed{110\to x\,y\,\bar{z}}$

Соединив полученные конъюнкции знаком дизъюнкции, получим СДНФ функции: $f(x,y,z) = \bar{x}\ \bar{y}\ \bar{z} \vee \bar{x}\,y\,\bar{z} \vee \bar{x}\,y\,z \vee x\ \bar{y}\ \bar{z} \vee x\,\bar{y}\,z \vee x\,y\,\bar{z}$.

Далее по таблице значений построим <u>СКНФ</u>. Для этого рассмотрим только все те наборы, на которых функция принимает <u>нулевые</u> значения (у нас это 001 и 111).Каждой единице из набора поставим в соответствие отрицание соответствующей переменной, а нулю – саму переменную; таким образом, по каждому набору построим дизъюнкцию переменных и их отрицаний: $(0,0,1) \to x \vee y \vee \bar{z}$; $(1,1,1) \to \bar{x} \vee \bar{y} \vee \bar{z}$.

Соединив полученные дизъюнкции знаком конъюнкции, получим СКНФ для нашей функции: $f(x,y,z) = (x \vee y \vee \bar{z})\&(\bar{x} \vee \bar{y} \vee \bar{z})$.

Пример. Получим СДНФ и СКНФ для функции $f(x,y,z) = \bar{z} \vee \bar{x}\,y \vee x\,\bar{y}$ по ее формуле, не выполняя построения таблицы значений.

$f(x,y,z) = \bar{z} \vee \bar{x}\,y \vee x\,\bar{y} = (x \vee \bar{x})(y \vee \bar{y})\,\bar{z} \vee \bar{x}\,y\,(z \vee \bar{z}) \vee x\,\bar{y}\,(z \vee \bar{z}) =$

$= x\,y\,\bar{z} \vee \underline{x\,\bar{y}\,\bar{z}} \vee \underline{\bar{x}\,y\,\bar{z}} \vee \bar{x}\,\bar{y}\,\bar{z} \vee \bar{x}\,y\,z \vee \underline{\bar{x}\,y\,\bar{z}} \vee x\,\bar{y}\,z \vee \underline{x\,\bar{y}\,\bar{z}}.$

$= \mathbf{x\,y\,\bar{z} \vee x\,\bar{y}\,\bar{z} \vee \bar{x}\,y\,\bar{z} \vee \bar{x}\,\bar{y}\,\bar{z} \vee \bar{x}\,y\,z \vee x\,\bar{y}\,z}$ – СДНФ.

$f(x,y,z) = \bar{z} \vee \bar{x}\,y \vee x\,\bar{y} = (\bar{z} \vee \bar{x}\,y) \vee x\,\bar{y} = (\bar{z} \vee \bar{x})(\bar{z} \vee y) \vee x\,\bar{y} =$

$= ((\bar{z} \vee \bar{x})(\bar{z} \vee y) \vee x)((\bar{z} \vee \bar{x})(\bar{z} \vee y) \vee \bar{y}) =$

$= (\bar{z} \vee \bar{x} \vee x)(\bar{z} \vee y \vee x)(\bar{z} \vee \bar{x} \vee \bar{y})(\bar{z} \vee y \vee \bar{y}) =$

$= (\bar{z} \vee 1)(\bar{z} \vee y \vee x) = (\bar{z} \vee \bar{x} \vee \bar{y})(\bar{z} \vee 1) = \mathbf{(\bar{z} \vee y \vee x)(\bar{z} \vee \bar{x} \vee \bar{y})}$ – СКНФ.

Пример. Получим СДНФ и СКНФ функции f(x,y,z), используя предельные разложения Шеннона.

Формула предельного дизъюнктивного разложения Шеннона, дающая <u>СДНФ функции</u>:

$\boxed{f(x,y,z) = \vee\ x^{\sigma_1}\&\ y^{\sigma_2}\&\ z^{\sigma_3}}$, где дизъюнкция берется по всем наборам $(\sigma_1,\sigma_2,\sigma_3)$ таким, что $f(\sigma_1,\sigma_2,\sigma_3)= 1$.

Формула предельного конъюнктивного разложения Шеннона, дающая <u>СКНФ функции</u>:

$\boxed{f(x,y,z) = \&(x^{\neg\sigma_1} \vee y^{\neg\sigma_2} \vee z^{\neg\sigma_3})}$, где конъюнкция берется по всем наборам $(\sigma_1,\sigma_2,\sigma_3)$ таким, что $f(\sigma_1,\sigma_2,\sigma_3)=0$.

Здесь
$$a^{\sigma} = \begin{cases} \neg a, & \text{åñëè } \sigma = 0, \\ a, & \text{åñëè } \sigma = 1. \end{cases}$$

Запишем СДНФ для заданной функции $f(x,y,z)$, принимающей значение 1 на наборах (0,0,0), (0,1,0), (0,1,1), (1,0,0), (1,0,1), (1,1,0):

$$f(x,y,z) = x^0\&y^0\&z^0 \vee x^0\&y^1\&z^0 \vee x^0\&y^1\&z^1 \vee x^1\&y^0\&z^0 \vee x^1\&y^0\&z^1 \vee x^1\&y^1\&z^0 =$$
$$= \neg x\&\neg y\&\neg z \vee \neg x\&y\&\neg z \vee \neg x\&y\&z \vee x\&\neg y\&\neg z \vee x\&\neg y\&z \vee x\&y\&\neg z.$$

Запишем СКНФ для заданной функции $f(x,y,z)$, принимающей значение 0 на наборах (0,0,1), (1,1,1):

$$f(x,y,z) = (x^{\neg 0} \vee y^{\neg 0} \vee z^{\neg 1})\&(x^{\neg 1} \vee y^{\neg 1} \vee z^{\neg 1}) = (x^1 \vee y^1\& \vee z^0)\&(x^0 \vee y^0 \vee z^0) = (x \vee y \vee \neg z)\ \&\ (\neg x \vee \neg y \vee \neg z).$$

4. Построение минимальной ДНФ функции с помощью карт Карно

	a	**¬a**		**a**	
b	a&b&c&d	¬a&b&c&d	¬a&b&¬c&d	a&b&¬c&d	**d**
	a&b&c&¬d	¬a&b&c&¬d	¬a&b&¬c&¬d	a&b&¬c&¬d	**¬d**
¬b	a&¬b&c&¬d	¬a&¬b&c&¬d	¬a&¬b&¬c&¬d	a&¬b&¬c&¬d	
	a&¬b&c&d	¬a&¬b&c&d	¬a&¬b&¬c&d	a&¬b&¬c&d	**d**
	c		**¬c**		

Таблица 1

	a	**¬a**		**a**	
b	1111	0111	0101	1101	**d**
	1110	0110	0100	1100	**¬d**
¬b	1010	0010	0000	1000	
	1011	0011	0001	1001	**d**
	c		**¬c**		

Таблица 2

Карта Карно для функции четырех переменных f(a,b,c,d) представляет собой таблицу, где предусмотрены ячейки для всех возможных конъюнкций четырех переменных a,b,c,d и их отрицаний (см. таблицу 1).

Каждой ячейке таблицы 1 взаимно однозначно соответствует набор из нулей и единиц (см. таблицу 2). Таблица устроена таким образом, что любая пара соседних по горизонтали или по вертикали наборов отличается друг от друга только в одном разряде.

Заметим, что карта Карно представляет собой таблицу с пустыми внутренними ячейками. В таблицах 1 и 2 внутренние ячейки заполнены только для того, чтобы показать их соответствие конъюнкциям и наборам из нулей и единиц.

Карту Карно удобно заполнять, пользуясь таблицей 1, если функция представлена в виде СДНФ, и пользуясь таблицей 2, если функция задана таблицей значений.

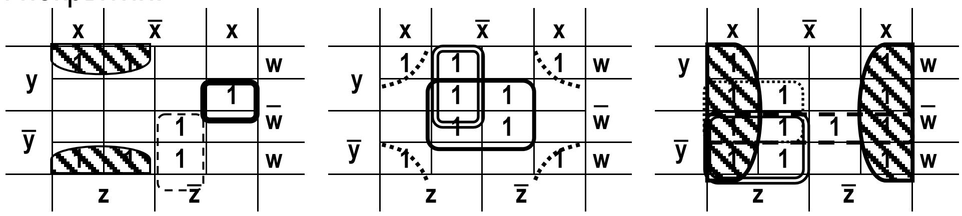
Для получения минимальной ДНФ функции:

1) карты Карно заполняем единицами следующим образом: в ячейку записываем 1, если соответствующая элементарная конъюнкция присутствует в СДНФ функции, или на соответствующем наборе функция принимает единичное значение;

2) строим покрытия всех единиц максимально возможного размера (длина стороны покрытия обязательно является степенью двойки – $2^0=1$, $2^1=2$, $2^2=4$, $2^3=8$ и т.д.); количество покрытий должно быть минимальным; при построении покрытий карты Карно можно мысленно склеивать около горизонтальной оси, около вертикальной оси и около обеих осей сразу; покрытия могут перекрываться; лишних покрытий строить не нужно, иначе получаемая ДНФ не будет минимальной;

3) каждому покрытию ставим в соответствие элементарную конъюнкцию переменных по следующему правилу: если переменная попадает на покрытие и с отрицанием, и без отрицания, то такая переменная в конъюнкцию не входит; остальные переменные попадают в конъюнкцию. Чем больше покрытие, тем меньше переменных входит в конъюнкцию, так для функции 4-х переменных покрытию 1×1 соответствует конъюнкция 4-х переменных, покрытию 1×2 - 3-х переменных, покрытию 2×2 - 2-х переменных и т.д.

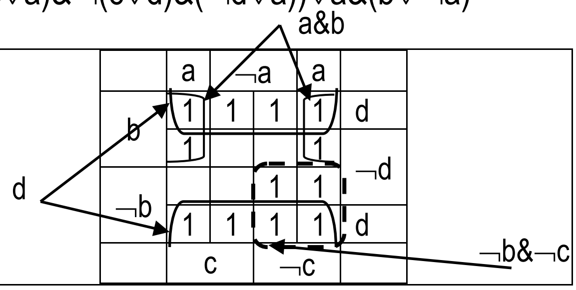
Примеры покрытий:



Пример. Получим минимальную ДНФ функции f(a,b,c,d), используя карты Карно (таблицу значений функции см. выше).

f(a,b,c,d)= ((¬c∨d)&(d∨a))∨((a∨¬b)&(¬c∨a)&¬(c∨d)&(¬d∨a))∨a&(b∨¬a)

	a	¬a		a	
b	1	1	1	1	d
	1			1	¬d
¬b			1	1	
	1	1	1	1	d
	c		¬c		



Результат: минимальная ДНФ функции f(a,b,c,d) = d∨¬b&¬c∨a&b.

5. Получение минимальной ДНФ функции методом Куайна-Мак-Класки

Рассматриваем только те наборы, где значение функции равно 1.

Группируем наборы по количеству единиц в них. Далее объединяем наборы из соседних групп.

Объединению подлежат только те наборы, которые различаются лишь в одном разряде (соседние наборы). В объединяемых наборах разряд, в котором есть различие, заменяется другим символом (например, "+"). Например, 010 и 011 объединяются в 01+.

Если для каких-то строк таблицы такое объединение невозможно, переносим эти строки в следующую таблицу, не производя над ними никаких действий вплоть до шага, где требуется устранить избыточность.

Группируем строки с одинаковой позицией знаков "+" в них. Объединяем наборы в пределах каждой группы, если это возможно (объединяемые наборы должны различаться только в одном разряде). Из нескольких одинаковых строк (если такие есть) оставляем одну. Выполняем операции, описанные в этом абзаце, до тех пор, пока это возможно.

Далее устраняем избыточность: если в некоторой строке все присутствующие в ней номера имеются в других строках, то эта строка является избыточной, и ее удаляем.

Полностью реализация метода показана на примере функции f(a,b,c,d), таблицу значений которой получили выше.

f(a,b,c,d)= ((¬c∨d)&(d∨a))∨((a∨¬b)&(¬c∨a)&¬(c∨d)&(¬d∨a))∨a&(b∨¬a)

1) Выпишем из таблицы значений f(a,b,c,d) только те наборы, на которых значение функции равно единице

	a	b	c	d	f
0	0	0	0	0	1
1	0	0	0	1	1
3	0	0	1	1	1
5	0	1	0	1	1
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	1
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

2) Группируем наборы по количеству единиц в них:

		a	b	c	d
без 1	0	0	0	0	0
с одной 1	1	0	0	0	1
	8	1	0	0	0
с двумя 1	3	0	0	1	1
	5	0	1	0	1
	9	1	0	0	1
	12	1	1	0	0
с тремя 1	7	0	1	1	1
	11	1	0	1	1
	13	1	1	0	1
	14	1	1	1	0
с четырьмя 1	15	1	1	1	1

3) Строим таблицу (строки формируются из соседних наборов)

	a	b	c	d
0,1	0	0	0	+
0,8	+	0	0	0
1,3	0	0	+	1
1,5	0	+	0	1
1,9	+	0	0	1
8,9	1	0	0	+
8,12	1	+	0	0
3,7	0	+	1	1
3,11	+	0	1	1
5,7	0	1	+	1
5,13	+	1	0	1
9,11	1	0	+	1
9,13	1	+	0	1
12,14	1	1	+	0
7,15	+	1	1	1
11,15	1	+	1	1
13,15	1	1	+	1
14,15	1	1	1	+

4) Группируем наборы с одинаковой позицией знаков (+) в них:

	a	b	c	d
0,1	0	0	0	+
8,9	1	0	0	+
14,15	1	1	1	+
1,3	0	0	+	1
5,7	0	1	+	1
9,11	1	0	+	1
12,14	1	1	+	0
13,15	1	1	+	1
1,5	0	+	0	1
8,12	1	+	0	0
3,7	0	+	1	1
9,13	1	+	0	1
11,15	1	+	1	1
0,8	+	0	0	0
1,9	+	0	0	1
3,11	+	0	1	1
5,13	+	1	0	1
7,15	+	1	1	1

5) Составляем новую таблицу, где строки формируются из соседних наборов

	a	b	c	d
0,1,8,9	+	0	0	+
1,3,5,7	0	+	+	1
1,3,9,11	+	0	+	1
5,7,13,15	+	1	+	1
9,11,13,15	1	+	+	1
12,14,13,15	1	1	+	+
1,5,3,7	0	+	+	1
1,5,9,13	+	+	0	1
8,12,9,13	1	+	0	+
3,7,11,15	+	+	1	1
9,13,11,15	1	+	+	1
0,8,1,9	+	0	0	+
1,9,3,11	+	0	+	1
1,9,5,13	+	+	0	1
3,11,7,15	+	+	1	1
5,13,7,15	+	1	+	1
14,15	1	1	1	+

6) В новой таблице из нескольких одинаковых наборов оставим один:

	a	b	c	d
0,1,8,9	+	0	0	+
1,3,5,7	0	+	+	1
1,3,9,11	+	0	+	1
5,7,13,15	+	1	+	1
9,11,13,15	1	+	+	1
1,5,9,13	+	+	0	1
8,12,9,13	1	+	0	+
3,7,11,15	+	+	1	1
12,14,13,15	1	1	+	+
14,15	1	1	1	+

7) Группируем наборы с одинаковой позицией знаков (+) в них:

0,1,8,9	+	0	0	+
1,3,5,7	0	+	+	1
9,11,13,15	1	+	+	1
1,3,9,11	+	0	+	1
5,7,13,15	+	1	+	1
1,5,9,13	+	+	0	1
3,7,11,15	+	+	1	1
8,12,9,13	1	+	0	+
12,14,13,15	1	1	+	+
14,15	1	1	1	+

8) Объединяем строки, наборы в которых различаются между собой только в одном разряде:

	a	b	c	d
0,1,8,9	+	0	0	+
1,3,5,7,9,11,13,15	+	+	+	1
1,3,9,11,5,7,13,15	+	+	+	1
1,5,9,13,3,7,11,15	+	+	+	1
8,12,9,13	1	+	0	+
12,14,13,15	1	1	+	+
14,15	1	1	1	+

9) Из нескольких строк таблицы таких, что записанные в них наборы одинаковы, оставляем одну:

	a	b	c	d
0,1,8,9	+	0	0	+
1,3,5,7,9,11,13,15	+	+	+	1
8,12,9,13	1	+	0	+
12,14,13,15	1	1	+	+
14,15	1	1	1	+

10) Устраняем избыточность: 0,1,8,9 – не является избыточной, т.к. 0 не встречается больше ни в какой группе; 1,3,5,7,9,11,13,15 – не является избыточной, т.к., например, 11 нигде больше не встречается; 14,15 – избыточна, т.к. эти номера есть в 12,14,13,15; 8,12,9,13 – избыточна, т.к. 8 и 9 есть в 0,1,8,9, а 12 и 13 есть в 12,14,13,15. Удалив лишние строки, получаем:

	a	b	c	d	конъюнкции:
0,1,8,9	+	0	0	+	$\neg b \& \neg c$
1,3,5,7,9,11,13,15	+	+	+	1	d
12,14,13,15	1	1	+	+	a&b

Результат: минимальная ДНФ функции $f(a,b,c,d) = d \vee \neg b \& \neg c \vee a \& b$.

6. Привести выражение, задающее минимальную ДНФ функции f(a,b,c,d), к КНФ

Пусть минимальная ДНФ функции $f(a,b,c,d) = d \vee \neg b \& \neg c \vee a \& b$.

Используя законы булевой алгебры, приведем данную ДНФ к КНФ.

$f(a,b,c,d) = d \vee \neg b \& \neg c \vee a \& b = (d \vee \neg b \& \neg c \vee a) \& (d \vee \neg b \& \neg c \vee b) =$

$= (d \vee a \vee \neg b \& \neg c) \& (d \vee b \vee \neg b \& \neg c) = ((d \vee a) \vee (\neg b \& \neg c)) \& ((d \vee b) \vee (\neg b \& \neg c)) =$

$= (d \vee a \vee \neg b) \& (d \vee a \vee \neg c) \& (d \vee b \vee \neg b) \& (d \vee b \vee \neg c) =$

$= (d \vee a \vee \neg b) \& (d \vee a \vee \neg c) \& (d \vee b \vee \neg c)$ – это и есть КНФ функции f(a,b,c,d).

7. Построение минимальной бинарной диаграммы решений (БДР) функции

Бинарная диаграмма решений (БДР) булевой функции $f(x_1,\ldots,x_n)$ – это корневой направленный ациклический граф $G=(V, E)$ с множеством вершин V, содержащим два типа вершин: *нетерминальные* и *терминальные*.

Каждая нетерминальная вершина помечена переменной x_i из множества $\{x_1,\ldots,x_n\}$. Вершины, помеченные одной и той же переменной, располагаются на одном уровне.

Из каждой нетерминальной вершины, помеченной как x_i, исходит ровно две дуги:

– пунктирная линия (левая дуга) соответствует нулевому значению переменной x_i,

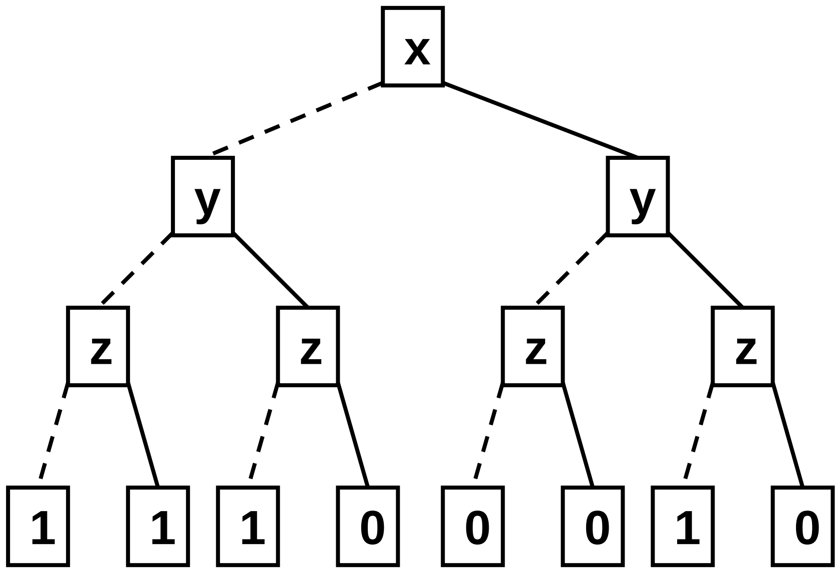
– сплошная линия (правая дуга) соответствует единичному значению переменной x_i.

Каждая терминальная вершина помечена 0 или 1 (значением функции) и не имеет исходящих дуг.

Сокращенная БДР для выбранного порядка расположения переменных по уровням – это БДР, в которой число нетерминальных вершин (промежуточных узлов) минимально. При построении сокращенной БДР пользуются следующими двумя правилами.

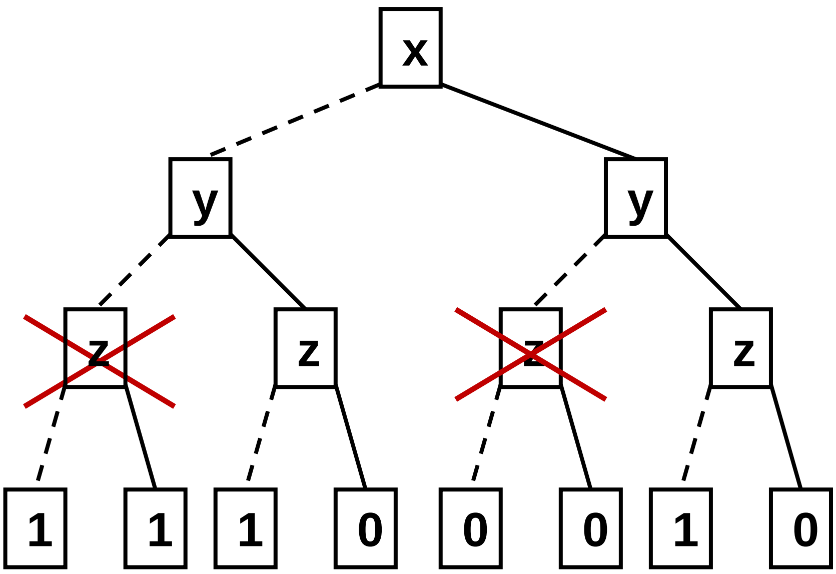
- В сокращенный БДР не должно быть избыточных узлов. Узел *v* считается **избыточным**, если обе исходящие из него дуги ведут в один и тот же узел *w*. Если такой узел найдется, то его удаляем, а входящие в него дуги перенаправляем в узел *w*.
- Из нескольких одинаковых подграфов оставляем только один. При этом дуги, ведущие в каждый из этих подграфов, перенаправляем в оставшийся подграф.

Пример. Построим сокращенную БДР для функции g(x,y,z) = (1110 0010).
Выберем порядок расположения переменных по уровням: **x, y, z**.
Построим полную БДР:

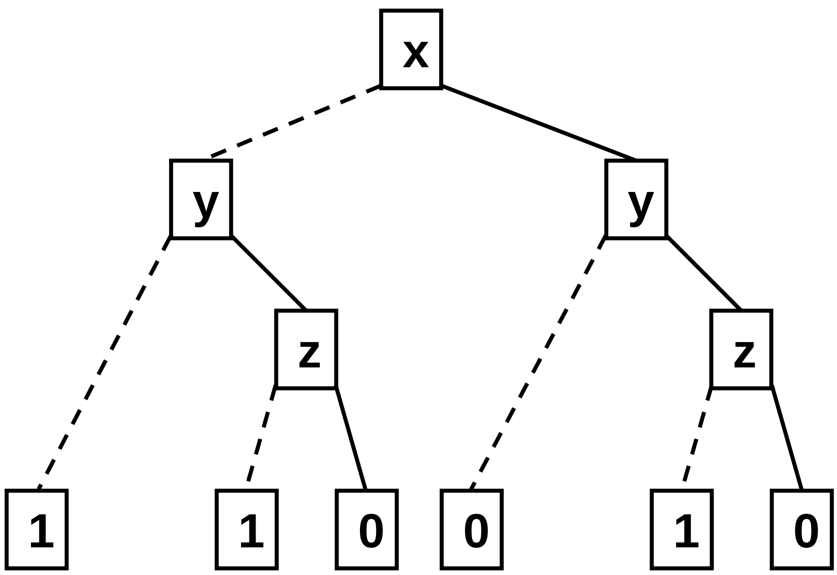


x	y	z	g(x,y,z)
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

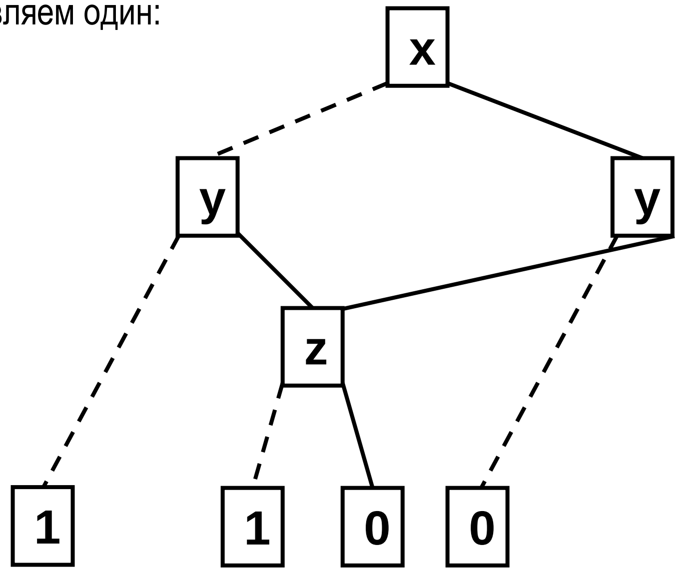
Получим сокращенную БДР для выбранного порядка переменных.
Удаляем избыточные узлы:



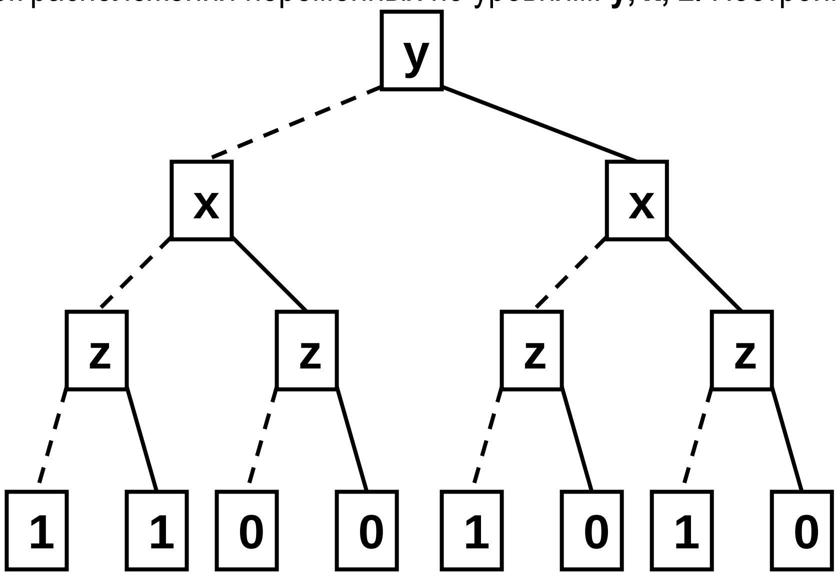
Получим:



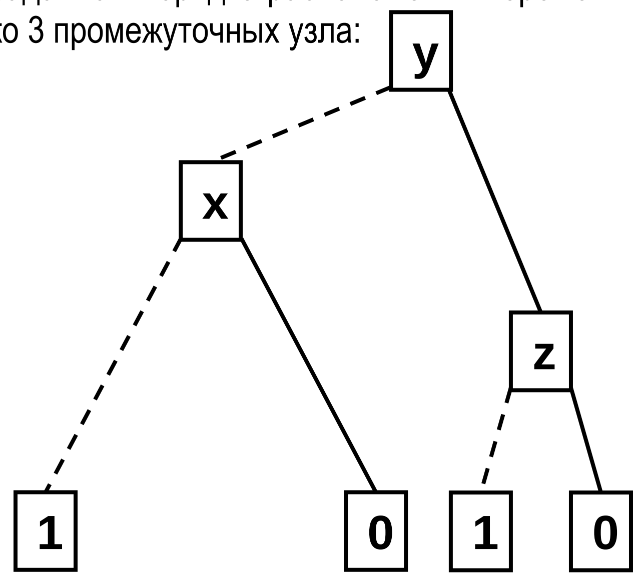
Вместо двух одинаковых подграфов оставляем один:
В полученной сокращенной БДР,
построенной для порядка x, y, z,
имеется 4 промежуточных узла.



Изменим порядок расположения переменных по уровням: **y, x, z**. Построим полную БДР:



При заданном порядке расположения переменных по уровням сокращенная БДР имеет только 3 промежуточных узла:




```
if y then
   if z then
      f:=false
   else
      f:=true
else
   if x then
      f:=false
   else
      f:=true
```

Поскольку у рассматриваемой функции фиктивных переменных нет (все три переменные существенны), то количество узлов в БДР не может быть уменьшено. Следовательно, получена БДР с минимальным числом узлов.
По полученной БДР зададим функцию с использованием оператора IF-THEN-ELSE.
Заметим, что по БДР также можно определить ДНФ и КНФ булевой функции. Для получения ДНФ находим все пути из корневой вершины в терминальные со значением 1. В нашем примере таких пути два: $\neg y \wedge \neg x$ и $y \wedge \neg z$. Записываем минимальную ДНФ: $\neg y \wedge \neg x \vee y \wedge \neg z$.
Найденная ДНФ минимальна, т.к. имеющиеся на минимальной БДР пути не пересекаются.
Аналогично, по всем путям из корня в терминальные вершины со значением 0 запишем минимальную КНФ: $(y \vee \neg x) \wedge (\neg y \vee \neg z)$.

Итак, полная БДР булевой функции n переменных $f(x_1,\ldots,x_n)$ представляет собой дерево со следующими свойствами:

- Вершины соответствуют переменным, от которых зависит функция, и расположены по уровням. Каждому уровню соответствует одна переменная.
- Из каждой вершины выходит две дуги. Одна соответствует нулевому значению переменной (пунктирная линия), а вторая - единичному (сплошная линия).
- БДР имеет 2^n листьев, каждый из которых соответствует одному из значений функции.

Суть задачи о минимизации БДР (получение сокращенной БДР) состоит в том, чтобы минимизировать число вершин в БДР. При этом порядок расположения вершин по уровням может быть произвольный.

Для фиксированного порядка расположения переменных по уровням:

1. Строим полную БДР.
2. Двигаясь по уровням сверху вниз, для каждого уровня выполняем 2 действия:
   а) находим вершины, от прохождения которых не зависит значение функции, и удаляем их;
   б) находим одинаковые поддеревья и из нескольких одинаковых оставляем одно.
3. Приводим БДР к конечному виду (для заданного порядка переменных она будет содержать минимальное число вершин).
4. Меняем порядок переменных и выполняем шаги 1-3 для нового порядка расположения переменных по уровням. Эти действия выполняем для всех возможных перестановок переменных, от которых зависит функция.
5. Выбираем тот порядок расположения переменных по уровням, который будет оптимальным (т.е. при котором БДР функции будет содержать минимальное количество вершин), если же таких вариантов несколько, то выбираем любой из них.

Логика высказываний и предикатов

Высказывание – это грамматически правильное повествовательное предложение, про которое в данный момент времени можно сказать, что оно либо истинно, либо ложно, но не то и другое одновременно.
Различают простые и сложные высказывания.

Простые высказывания обозначаются буквами латинского алфавита.
Сложные высказывания строятся из простых с помощью пяти операторов: $\neg$, $\wedge$, $\vee$, $\rightarrow$, ~.
Алфавит в алгебре высказываний – это множество

$$A = \{P, Q, R, \ldots, Z_1, \ldots, \neg, \wedge, \vee, \rightarrow, \sim, (,), 1, 0\}.$$

Некоторые слова в алфавите A являются *формулами* алгебры высказываний:
а) простые высказывания – формулы;
б) если F_1 и F_2 – формулы, то $(\neg F_1)$, $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$, $(F_1 \rightarrow F_2)$, $(F_1 \sim F_2)$ – тоже формулы;
в) других формул в алгебре высказываний нет.

Предикатом $P(x_1,\ldots,x_n)$ называется функция, переменные которой принимают значения из некоторого множества $M=X_1\times\ldots\times X_n$, а сама функция принимает значения из множества {0,1}, т.е.

$$P(x_1,\ldots,x_n)\colon M \rightarrow \{0,1\}.$$

Множество M называется *предметной областью*, а переменные $x_1,\ldots,x_n$ - *предметными переменными*.
Предикат от n аргументов называется n-местным предикатом.
Приписав конкретные значения всем аргументам предикатной функции, мы получаем нульместный предикат, или высказывание, к которому применимы все законы логики высказываний.
Предикат P называется *тождественно истинным* (*тождественно ложным*), если на всех наборах своих переменных он принимает значение 1(0).
Предикат называется *выполнимым*, если на некотором наборе своих переменных он принимает значение 1.

<u>Пример.</u> R(x) = “x нацело делится на 2” – одноместный предикат. R(4)=1 (истинное высказывание), R(5)=0 (ложное высказывание).
В программировании предикаты возникают в операторах if, until, while и т.п. при задании условий.
<u>Пример.</u>
if **<u>x<y</u>** then …Здесь “x<y” - двухместный предикат.
while **<u>x<0</u>** do … Здесь “x<0” - одноместный предикат.
Над предикатами можно производить обычные логические операции. В результате получаются новые предикаты.
<u>Пример.</u> Q(x) = “x нацело делится на 3”. Тогда выражение R(x)&Q(x) соответствует предикату “x нацело делится на 2 и на 3”.
В логике предикатов кроме операций логики высказываний применяются также операции связывания квантором.
Фразы “все x”, “для любого x”, “для каждого x” обозначаются в логике предикатов как $\forall x$.
Символ $\forall$ называется ***квантором общности*** ($\forall$ - перевернутая первая буква слова All).
Фразы “некоторые x”, “существует хотя бы одно значение x” обозначаются как $\exists x$.
Символ $\exists$ называется ***квантором существования*** ($\exists$ - перевернутся первая буква слова Exist).

Выставляя кванторы перед предикатами, мы как бы усиливаем (квантором общности) или ослабляем (квантором существования) действие этих предикатов.

Выражение $\forall xP(x)$ означает «для всех без исключения значений x P(x) истинно».

Выражение $\exists xP(x)$ означает «существует хотя бы одно значение x такое, что P(x) истинно».

<u>Пример.</u> Для предиката из предыдущего примера имеем: $\forall x(R(x)\&Q(x))$ – ложное высказывание, $\exists x(R(x)\&Q(x))$ – истинное высказывание.

Аргументы предикатной функции, связанные квантором общности или существования (т.е. записанные после знака $\forall$ или $\exists$), называются ***связанными переменными***; остальные аргументы предикатной функции называются ***свободными переменными***.

Очень часто при операции связывания предметных переменных квантором указывается предметная область, из которой предметные переменные берут свои значения. В это случае делается запись:

$(\forall x\in M)P(x)$ – «для любого значения x из предметной области M P(x) истинно»;

$(\exists x\in M)P(x)$ –«существует такое x из предметной области M, для которого P(x) истинно».

<u>Пример.</u> Пусть M={2,4,6}. Тогда для предикатов из предыдущих примеров справедливо следующее:

$(\forall x\in M)R(x) = 1$, $(\exists x\in M)R(x) = 1$, $(\forall x\in M)Q(x) = 0$, $(\exists x\in M)Q(x) = 1$.

Если все предметные переменные в предикатном выражении связаны, то оно превращается в обычное высказывание (нульместный предикат).