

## Номер 2

№ 21888 Открытый вариант 2025 (Уровень: Базовый)

Миша заполнял таблицу истинности логической функции  $F = (x \wedge \neg y) \vee (y \equiv z) \vee w$ , но успел заполнить лишь фрагмент из трёх различных её строк, даже не указав, какому столбцу таблицы соответствует каждая из переменных  $w, x, y, z$ .

				F
		1		0
0	0	0	1	0
1	0		1	0

Определите, какому столбцу таблицы соответствует каждая из переменных  $w, x, y, z$ .

В ответе напишите буквы  $w, x, y, z$  в том порядке, в котором идут соответствующие им столбцы (сначала буква, соответствующая первому столбцу; затем буква, соответствующая второму столбцу, и т.д.). Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.

[Показать ответ](#)

xwzy

Смотри, есть 2 стула...ну ладно, вообще их 3 Давай рассмотрим самый простой, нудный, и долгий

## Самый простой, нудный, и долгий способ

Сначала пронумеруем каждое действие, 1 всегда инверсия, потом все что в скобках и в конце слева направо

$$F = (x^{(2)} \wedge \neg^{(1)} y) \vee^{(4)} (y \equiv^{(3)} z) \vee^{(5)} w,$$

`` Теперь нам нужно построить таблицу истинности для этого сначала запишем переменные и поставим единицы и нули

x|y|z|w

0|0|0|0

0|0|0|1

0|0|1|0

0|0|1|1

0|1|0|0

0|1|0|1

0|1|1|0

0|1|1|1

1|0|0|0

1|0|0|1

1|0|1|0

1|0|1|1

1|1|0|0

1|1|0|1

запомнить это можно следующим образом:

1) 1й столбик 8 нулей, потом 8 единиц

2й столбик чередуешь 4ку нулей и 4ку единиц

3й столбик чередуешь 2ку нулей и 2ку единиц

4й столбик чередуешь нули и единицы

2) сначала записываешь 0 0 0 0 в 1 строку,  
а далее прибавляешь 1 к концу для каждой  
следующей

```

1|1|1|0
1|1|1|1

```

После этого применим операции

A	B	$A \wedge B$	A	B	$A \vee B$	A	B	$A \rightarrow B$
0	0	0	0	0	0	0	0	1
0	1	0	0	1	1	0	1	1
1	0	0	1	0	1	1	0	0
1	1	1	1	1	1	1	1	1

A	B	$A \equiv B$	A	$\neg A$
0	0	1	0	1
0	1	0	1	0
1	0	0		
1	1	1		

Где:

- $A \wedge B$  – конъюнкция (AND)
- $A \vee B$  – дизъюнкция (OR)
- $A \rightarrow B$  – импликация
- $\neg A$  – инверсия A
- $A \equiv B$  – тождество (эквивалентность)

и после этого соответственно табличкам с операциями сопоставляем 0 и 1 Так получим табличку:

x	y	z	w	1	2	3	4
0	0	0	0	1	0	1	1
0	0	0	1	1	0	1	1
0	0	1	0	1	0	0	0
0	0	1	1	1	0	0	1
0	1	0	0	0	0	0	0
0	1	0	1	0	0	0	1
0	1	1	0	0	0	1	1
0	1	1	1	0	0	1	1
1	0	0	0	1	1	1	1
1	0	0	1	1	1	1	1
1	0	1	0	1	1	0	1
1	0	1	1	1	1	0	1
1	1	0	0	0	0	0	0
1	1	0	1	0	0	0	1

	1		1		1		0		0		0		1		1	
	1		1		1		1		0		0		1		1	

Последний столбик, в данном случае (4) будет итогом нашей функции  
 Теперь смотрим на столбик таблицы из задания. Там записан результат, в нашем случае 0

Теперь отберем строки с значениями, нам нужны те, где в результате 0

	x		y		z		w		1		2		3		4	
	0		0		0		0		1		0		1		1	
	0		0		0		1		1		0		1		1	
	0		0		1		0		1		0		0		0	
	0		0		1		1		1		0		0		1	
	0		1		0		0		0		0		0		0	
	0		1		0		1		0		0		0		1	
	0		1		1		0		0		0		1		1	
	0		1		1		1		0		0		1		1	
	1		0		0		0		1		1		1		1	
	1		0		0		1		1		1		1		1	
	1		0		1		0		1		1		0		1	
	1		0		1		1		1		1		0		1	
	1		1		0		0		0		0		0		0	
	1		1		0		1		0		0		0		1	
	1		1		1		0		0		0		1		1	
	1		1		1		1		0		0		1		1	

Наши строки:

	x		y		z		w	
	0		0		1		0	
	0		1		0		0	
	1		1		0		0	

Их теперь нужно соотнести с таблицей

# Сопоставление

Наша таблица	Таблица по условию
x   y   z   w	?   ?   ?   ?
0   0   1   0	1

0	1	0	0	0	0	0	1
1	1	0	0	1	0	1	

Первым делом попробуем заполнить таблицу в исходной таблице есть 2 строки с одной единицей и строка с двумя единицами в нашей таблице все точно так же, значит можем сразу сказать, что все пустые ячейки с "0", тогда:

Наша таблица	Таблица по условию
x   y   z   w	?   ?   ?   ?
0   0   1   0	0   0   1   0
0   1   0   0	0   0   0   1
1   1   0   0	1   0   0   1

так могло и не получиться, в таком случае нужно было смотреть по столбикам  
Если бы не получилось и так, то пропускаем пункт

Теперь же попробуем найти уникальную строку В нашем случае такая есть: | 1 | 1 | 0 | 0 | <--- единственная с 2мя "1", тогда на местах "1" либо x либо y

Наша таблица	Таблица по условию
x   y   z   w	x/y   ?   ?   x/y
0   0   1   0	0   0   1   0
0   1   0   0	0   0   0   1
1   1   0   0	1   0   0   1

Теперь же заметим, что у нас во второй строке | 0 | 0 | 0 | 1 | на местах где может быть x и y стоят 0 и 1 и при этом в строке только одна "1" Значит определив что под "1" сможем сразу сказать где x, а где y

x	y	z	w
0	1	0	0

у нас только одна строка где одна 1 и y = 1, значит таблица теперь выглядит следующим образом:

Наша таблица	Таблица по условию
x   y   z   w	x   ?   ?   y
0   0   1   0	0   0   1   0
0   1   0   0	0   0   0   1
1   1   0   0	1   0   0   1

теперь давай посмотрим на 2ю строку с одной "1"

	x		y		z		w	
	0		0		1		0	

Тут под "1" стоит переменная z. Значит:

Наша таблица	Таблица по условию
x   y   z   w	x   ?   z   y
0   0   1   0	0   0   1   0
0   1   0   0	0   0   0   1
1   1   0   0	1   0   0   1

И теперь давай определим место w. Но теперь посмотрим не по строке, а по столбику, нам нужен столбик с 3мя нулями

	w	
	0	
	0	
	0	

Тогда получается:

Наша таблица	Таблица по условию
x   y   z   w	x   w   z   y
0   0   1   0	0   0   1   0
0   1   0   0	0   0   0   1
1   1   0   0	1   0   0   1

Ответ: xwzy



Каждая задача решается уникальным образом, но можно вывести несколько путей решения:

1. Для начала нужно попробовать заполнить таблицу недостающими цифрами. Это может не получиться, но, ничего страшного, скорее всего, хоть где-то по одной строке удастся распознать. Но даже если не получится, это особо не критично.
2. После этого ищем уникальную строку, которую сразу можно сопоставить.
3. Если таковой нет, то смотрим уникальный столбец. Иногда нужно сочетать и 2 и 3, чтобы не ошибиться, иначе может не получиться.

#### 4. далее пытаемся сопоставить наши буквы со столбиками

Обращая внимание на такие комбинации как:

Такие строки могут сразу помочь сопоставить переменные

```
| 1 | 0 | 0 | 0 |      | 1 | 0 | 1 | 1 |
```

Причем сам порядок не важен, само наличие одного 0|1 дает возможность сразу сказать что это за переменная

А так же такие столбики тоже помогут опознать переменную

```
| 0 |      | 1 |  
| 0 |      | 1 |  
| 0 |      | 1 |
```

При этом не забывай про уникальность столбиков и строк, например

```
|   |   |   |   |  
| 0 | 1 | 1 | 1 |  
| 0 | 1 | 0 | 1 |  
| 0 | 1 | 0 | 0 |
```

Можем сразу заметить, что у нас все столбцы разные, значит их можно сразу верно сопоставить

или же такой вариант, с различными строками

```
|   |   |   |   |  
| 1 | 1 | 1 | 1 |  
| 1 | 1 | 0 | 1 |  
| 0 | 1 | 1 | 0 |
```



Но при этом есть одна особенность у этого способа и решения через циклы могут появляться лишние строки, они не участвуют в таблице, но при этом тоже могут быть равны 0 В таком случае, нужно аргументировать, что эта строка не подходит Это можно сделать следующим образом: 1. Посмотреть столбики, иногда например нужно чтобы было 3 нуля или 3 единицы 2. Посмотреть на строки, иногда может так случиться что некуда поставить 3 нуля или 3 единицы В таком случае можно сразу выяснить, что строка

лишняя 3. Если не помогли 1 и 2, то просто перебирать строки, с лишней табличку составить не получится, так, методом исключения сможешь выяснить нужные строки

## Перебор таблицы циклами

Тут самое главное понимание что такое цикл и оператор условия Шаблон кода:

```
1  print('x y z w')
2  for x in range(2):
3      for y in range(2):
4          for z in range(2):
5              for w in range(2):
6                  if твое условие функции == значение функции(0|1):
7                      print(x, y, z, w)
```

А теперь давай решим нашу задачу

№ 21888 Открытый вариант 2025 (Уровень: Базовый)

Миша заполнял таблицу истинности логической функции  $F = (x \wedge \neg y) \vee (y \equiv z) \vee w$ , но успел заполнить лишь фрагмент из трёх различных её строк, даже не указав, какому столбцу таблицы соответствует каждая из переменных  $w, x, y, z$ .

				F
		1		0
0	0	0	1	0
1	0		1	0

Определите, какому столбцу таблицы соответствует каждая из переменных  $w, x, y, z$ .

В ответе напишите буквы  $w, x, y, z$  в том порядке, в котором идут соответствующие им столбцы (сначала буква, соответствующая первому столбцу; затем буква, соответствующая второму столбцу, и т.д.). Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.

[Показать ответ](#)

**xwzy**

Нам нужно лишь переписать функцию и значение функции

функция:  $(x \wedge \neg y) \vee (y \equiv z) \vee w$

значение функции:

- $A \wedge B$  — конъюнкция (AND)
- $A \vee B$  — дизъюнкция (OR)
- $A \rightarrow B$  — импликация
- $\neg A$  — инверсия A
- $A \equiv B$  — тождество (эквивалентность)

Единственное при переписывании нужно учесть один нюанс:

как операции выглядят в коде

$\wedge$  — and

$\vee$  — or

$\rightarrow$  — <=

```
¬ -- not
≡ -- ==
```

Тогда наше решение выглядит следующим образом:

```
1 print('x y z w')
2 for x in range(2):
3     for y in range(2):
4         for z in range(2):
5             for w in range(2):
6                 if ((x and (not y)) or (y == z) or w) == 0:
7                     print(x, y, z, w)
```

```
x y z w
0 0 1 0
0 1 0 0
1 1 0 0
```

## Теперь давай посмотрим как работает код:

`print('x y z w')` – выводим символы для удобства  
просто чтобы было понятно какая переменная за что отвечает

`for x in range(2):`  
цикл для перебора  
напомню что `range(2)`: просто подставляет под `x` `[0, 1]`

Когда же у нас несколько циклов, то мы переберем все случаи

```
for x in range(2):
    for y in range(2):
        for z in range(2):
            for w in range(2):
```

таким образом мы получим ту же таблицу

```
  x y z w
1. 0 0 0 0
2. 0 0 0 1
3. 0 0 1 0
```

и так далее

`if ((x and (not y)) or (y == z) or w) == 0:`  
тут же мы смотрим наше условие, где под `x y z w` подставляются наши единицы и нули  
в соответствии с шагом



```
И когда так получится что наше условие выполнилось, то мы выведем наши x y z w
print(x, y, z, w)
```

так получим наши x y z w, при которых (( x and (not y)) or (y == z) or w) равно 0

```
x y z w
0 0 1 0
0 1 0 0
1 1 0 0
```

А сопоставлять мы уже умеем)

## Способ перебора x y z w и таблицы

Самый прекрасный способ, для которой нужно запомнить и понять всего одну строчку Но который окупит наше умение запоминать, потому как он выводит сразу ответ

Шаблон кода:

```
1  from itertools import permutations, product
2
3
4  def f(x, y, z, w):
5      return наша функция
6
7  for x1, x2, x3, x4 in product([0,1], repeat = 4):
8      t = (
9          (x1, x2, 1, x3, 0),
10         (0, 0, 0, 1, 0),
11         (1, 0, x4, 1, 0)
12         переменная в которую мы заносим наши строки таблицы
13     )
14     if len(t) == len(set(t)):
15         for p in permutations('xyzw'):
16             if all(f(**dict(zip(p, m))) == m[-1] for m in t):
17                 print(*p)
```

Давай решим нашу задачу:

```
1  from itertools import permutations, product
2
3
4  def f(x, y, z, w):
5      return( x and (not y)) or (y == z) or w
6
```

```
7  for x1, x2, x3, x4 in product([0,1], repeat = 4):
8      t = (
9          (x1, x2, 1, x3, 0),
10         (0, 0, 0, 1, 0),
11         (1, 0, x4, 1, 0)
12     )
13     if len(t) == len(set(t)):
14         for p in permutations('xyzw'):
15             if all(f(**dict(zip(p, m))) == m[-1] for m in t):
16                 print(*p)
```

И теперь давай посмотрим что для чего нам нужно