

[Главная](#) / [Калькуляторы](#) / Симплекс метод

## Калькулятор симплекс-метода

Количество переменных: Количество ограничений: 

Целевая функция:

  $x_1$  +   $x_2$  +   $x_3 \rightarrow$  max ▾

Ограничения:

  $x_1$  +   $x_2$  +   $x_3 \leq$  ▾   $x_1$  +   $x_2$  +   $x_3 \leq$  ▾   $x_1$  +   $x_2$  +   $x_3 \leq$  ▾   $x_1$  +   $x_2$  +   $x_3 \leq$  ▾   $x_1$  +   $x_2$  +   $x_3 \leq$  ▾   $x_1$  +   $x_2$  +   $x_3 \geq$  ▾  $x_1, x_2, x_3 \geq 0$ 

В виде дробей

С решением

Базовый симплекс-метод ▾

Очистить

Решить

В двойственную

### Введённые данные

$$24 \cdot x_1 + 22 \cdot x_2 + 10 \cdot x_3 \rightarrow \max$$

$$2 \cdot x_1 + x_2 + x_3 \leq 2$$

$$x_1 + 2 \cdot x_2 - x_3 \leq -3$$

$$-x_1 + 2 \cdot x_2 + x_3 \leq 3$$

$$x_1 \leq 1$$

$$x_2 \leq 0$$

$$x_3 \geq 0$$

### Ответ

Решение задачи не существует.

### Решение базовым симплекс-методом

Меняем знаки у ограничений с  $\geq$ , путём умножения на -1:

$$2 \cdot x_1 + x_2 + x_3 \leq 2$$

$$x_1 + 2 \cdot x_2 - x_3 \leq -3$$

$$-x_1 + 2 \cdot x_2 + x_3 \leq 3$$

$$x_1 \leq 1$$

$$x_2 \leq 0$$

$$-x_3 \leq 0$$

Для каждого ограничения с неравенством добавляем дополнительные

$$\begin{aligned}x_1 + 2 \cdot x_2 - x_3 + x_5 &= -3 \\-x_1 + 2 \cdot x_2 + x_3 + x_6 &= 3 \\x_1 + x_7 &= 1 \\x_2 + x_8 &= 0 \\-x_3 + x_9 &= 0\end{aligned}$$

**Ищем начальное базисное решение:**

Ограничение 1 содержит неравенство, базисной будет добавленная дополнительная переменная  $x_4$

Ограничение 2 содержит неравенство, базисной будет добавленная дополнительная переменная  $x_5$

Ограничение 3 содержит неравенство, базисной будет добавленная дополнительная переменная  $x_6$

Ограничение 4 содержит неравенство, базисной будет добавленная дополнительная переменная  $x_7$

Ограничение 5 содержит неравенство, базисной будет добавленная дополнительная переменная  $x_8$

Ограничение 6 содержит неравенство, базисной будет добавленная дополнительная переменная  $x_9$

**Начальная симплекс-таблица**

С	24	22	10	0	0	0	0	0	0	0
базис	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	b
$x_4$	2	1	1	1	0	0	0	0	0	2
$x_5$	1	2	-1	0	1	0	0	0	0	-3
$x_6$	-1	2	1	0	0	1	0	0	0	3
$x_7$	1	0	0	0	0	0	1	0	0	1
$x_8$	0	1	0	0	0	0	0	1	0	0
$x_9$	0	0	-1	0	0	0	0	0	1	0

В столбце b присутствуют отрицательные значения.

Максимальное по модулю  $|b|_{\max} = |-3|$  находится в строке 2.

Максимальный по модулю элемент в строке 2 = -1 находится в столбце 3.

В качестве базисной переменной  $x_5$  берём  $x_3$ .

Делим строку 2 на -1. Из строк 1, 3, 4, 5, 6 вычитаем строку 2, умноженную на соответствующий элемент в столбце 3.

**Обновлённая таблица:**

С	24	22	10	0	0	0	0	0	0	0
базис	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	b
$x_4$	3	3	0	1	1	0	0	0	0	-1
$x_3$	-1	-2	1	0	-1	0	0	0	0	3
$x_6$	0	4	0	0	1	1	0	0	0	0
$x_7$	1	0	0	0	0	0	1	0	0	1
$x_8$	0	1	0	0	0	0	0	1	0	0
$x_9$	-1	-2	0	0	-1	0	0	0	1	3

В столбце b присутствуют отрицательные значения.

Максимальное по модулю  $|b|_{\max} = |-1|$  находится в строке 1.

## Как пользоваться калькулятором

- Задайте количество переменных и ограничений
- Введите коэффициенты целевой функции
- Введите коэффициенты ограничений и выберите условия ( $\leq$ ,  $=$  или  $\geq$ )
- Выберите тип решения
- Нажмите кнопку "Решить"

## Что умеет калькулятор симплекс-метода

- Решает основную задачу линейного программирования
- Позволяет получить решение с помощью основного симплекс-метода и метода искусственного базиса
- Работает с произвольным количеством переменных и ограничений

## Что такое симплекс-метод

**Задача линейного программирования** — это задача поиска неотрицательных значений параметров, на которых заданная линейная функция достигает своего максимума или минимума при заданных линейных ограничениях.

**Симплекс-метод** — алгоритм решения оптимизационной [задачи линейного программирования](#) путём перебора вершин выпуклого многогранника в многомерном пространстве. Алгоритм является универсальным методом, которым можно решить любую задачу линейного программирования.

Если вам тоже ничего не понятно из этого определения, то вы на верном пути. Чаще всего статьи про симплекс-метод очень сильно углубляются в дебри теории задачи линейного программирования, из-за чего очень легко потерять суть и так ничего и не понять. Мы постараемся описать алгоритм симплекс-метода так, чтобы показать, что в нём нет ничего страшного и на самом деле он весьма простой. Но сначала нам всё-таки потребуется ввести несколько определений.

**Целевая функция** — функция, максимум (или минимум) которой нужно найти. Представляет собой сумму произведений коэффициентов на значения переменных:  $F = c_1 \cdot x_1 + c_2 \cdot x_2 + \dots + c_n \cdot x_n$

**Ограничение** — условие вида  $a_1 \cdot x_1 + a_2 \cdot x_2 + \dots + a_n \cdot x_n \leq b$ , где вместо  $\leq$  ставится один из знаков:  $\leq$ ,  $=$  или  $\geq$

**План** — произвольный набор значений переменных  $x_1 \dots x_n$ .

## Алгоритм решения основной задачи ЛП симплекс-методом

Пусть в задаче есть  $m$  ограничений, а целевая функция зависит от  $n$  основных переменных. Первым делом необходимо привести все ограничения к **каноническому виду** — виду, в котором все условия задаются равенствами. Для этого предварительно все неравенства с  $\geq$  умножаются на  $-1$ , для получения неравенств с  $\leq$ .

Чтобы привести ограничения с неравенствами к каноническому виду, для каждого ограничения вводят переменную, называемую **дополнительной** с коэффициентом 1. В ответе эти переменные учитываться не будут, однако сильно упростят начальные вычисления. При этом

## Формирование начального базиса

После того как задача приведена к каноническому виду, необходимо найти начальный базис для формирования первого опорного решения. Если в процессе приведения были добавлены дополнительные переменные, то они становятся базисными.

Иначе необходимо выделить среди коэффициентов ограничений столбец, который участвует в формировании единичной матрицы в заданной строке (например, если требуется определить вторую базисную переменную, то необходимо искать столбец, в котором второе число равно 1, а остальные равны нулю). Если такой столбец найден, то переменная, соответствующая этому столбцу, становится базисной.

В противном случае можно поискать столбец, в котором все значения кроме числа в заданной строке равны нулю, и, если он будет найден, то разделить все значения строки на число, стоящее на пересечении этих строки и столбца, тем самым образовав столбец, участвующий в формировании единичной матрицы.

### ► Пример 2

Если такой столбец отсутствует, то для формирования базиса необходимо применить исключение Гаусса для первого ненулевого столбца, который ещё не является базисным. Для этого вся строка делится на элемент в найденном столбце, а из остальных строк вычитается полученная строка, разделённая на значение, стоящее в этом же столбце. После этой операции все значения вне данной строки будут обнулены, и столбец можно будет считать базисным.

### ► Пример 3

После того как базис сформирован, нужно построить начальную симплекс-таблицу. Она строится следующим образом:

- Для удобства в первой строке можно записать коэффициенты  $C_j$  целевой функции (для дополнительных переменных эти коэффициенты равны нулю)
- Вторая строка формирует шапку таблицы. В ней первый столбец называется базис, а остальные перечисляют основные переменные  $x_1 \dots x_n$  и дополнительные  $x_{n+1} \dots x_{n+k}$
- Затем построчно перечисляются базисные переменные и коэффициенты ограничений

Схематично начальная таблица будет выглядеть примерно так:

<b>С</b>	$c_1$	$c_2$	...	$c_n$	0	0	...	0	0
<b>базис</b>	$x_1$	$x_2$	...	$x_n$	$x_{n+1}$	$x_{n+2}$	...	$x_{n+k}$	<b>b</b>
$x_{e_1}$	$a_{11}$	$a_{12}$	...	$a_{1n}$	$a_{1n+1}$	$a_{1n+2}$	...	$a_{1n+k}$	$b_1$
$x_{e_2}$	$a_{21}$	$a_{22}$	...	$a_{2n}$	$a_{2n+1}$	$a_{2n+2}$	...	$a_{2n+k}$	$b_2$
...	...	...	...	...	...	...	...	...	...
$x_{e_m}$	$a_{m1}$	$a_{m2}$	...	$a_{mn}$	$a_{mn+1}$	$a_{mn+2}$	...	$a_{mn+k}$	$b_m$

## Избавляемся от отрицательных свободных коэффициентов

перейти к дальнейшим вычислениям. Чтобы избавиться от отрицательных значений  $b$  необходимо:

- Найти строку, в которой находится максимальное по модулю значение  $b$ . Пусть это будет строка  $i$ ;
- Найти максимальный по модулю элемент в этой строке. Пусть он находится в столбце  $j$ ;
- Строку  $i$  разделить на элемент, стоящий на пересечении  $i$ -ой строки и  $j$ -го столбца;
- Из каждой оставшейся строки  $k$  вычесть строку  $i$ , умноженную на элемент строки  $k$  и столбца  $j$ ;
- Переменную, соответствующую найденному столбцу  $j$ , сделать базисной (добавить в базис вместо переменной, находящейся в строке  $i$ ).

Этот шаг необходимо повторять до тех пор, пока все отрицательные  $b$  не станут положительными или в строке не останется отрицательных элементов. Если строка с максимальным по модулю  $b_i$  не содержит отрицательных элементов, то такая задача не имеет решений и на этом алгоритм заканчивает свою работу. В противном случае все  $b_i$  положительны и алгоритм переходит к следующему этапу — расчёту дельт.

#### [▶ Пример 4](#)

### Расчёт дельт

Дельты — это параметры, на основании которых проверяется оптимальность текущего решения и улучшается функция. Они рассчитываются для каждой из переменных ограничений и записываются последней строкой таблицы.

Для расчёта дельт используется следующая формула:  $\Delta_i = c_{e_1} \cdot a_{1i} + c_{e_2} \cdot a_{2i} + \dots + c_{e_m} \cdot a_{mi} - c_i$ . Проще говоря, чтобы вычислить дельту по заданной  $i$ -ой переменной, нужно перемножить коэффициенты условий в  $i$ -ом столбце на коэффициенты целевой функции при соответствующих базисных переменных, сложить эти произведения и вычесть из полученной суммы коэффициент целевой функции столбца  $i$ .

#### [▶ Пример 5](#)

### Проверка плана на оптимальность

После того как дельты рассчитаны, необходимо проверить оптимальность текущего плана. Критерий оптимальности формулируется следующим образом:

**При максимизации функции:** *текущее решение считается оптимальным, если в таблице отсутствуют отрицательные дельты.*

**При минимизации функции:** *текущее решение считается оптимальным, если в таблице отсутствуют положительные дельты.*

#### [▶ Пример 6](#)

Если текущий план оптимален, то алгоритм завершает свою работу. Значения переменных соответствуют значения столбца свободных коэффициентов  $b$ . Если свободной переменной нет в базисе, то её значение считается нулевым. Значение целевой функции, принимаемой на данном наборе, находится в строке с дельтами в том же столбце. Если

Если текущий план оказался не оптимальным, то алгоритм ищет столбец с наименьшей (с наибольшей, если ищется минимум) дельтой. После чего вычисляются симплекс-отношения  $Q$ . Для этого значения свободных коэффициентов делятся на ненулевые коэффициенты из найденного столбца. Если результат деления получается отрицательным, то такие отношения игнорируются.

Среди найденных симплекс-отношений ищется строка, в которой находится симплекс-отношение с наименьшим значением. Если таких отношений нет, то алгоритм останавливает свою работу, так как *целевая функция не ограничена и решения не существует*.

#### ► [Пример 7](#)

В противном случае строка с наименьшим отношением считается разрешающей и, аналогично избавлению от отрицательных свободных коэффициентов, делится на разрешающий элемент, расположенный в найденных столбце и строке, и из остальных строк вычитается найденная строка, разделённая на значения, стоящие в этом же столбце соответствующей строки. Переменная, стоящая в разрешающем столбце заменяет базисную переменную, находящуюся в найденной строке.

После этого вычисляются новые дельты и проверяется новый план. Так продолжается до тех пор пока не будет выполнен критерий оптимальности плана или не будет установлено, что решение не существует.

#### ► [Пример 8](#)

## Метод искусственного базиса

Очень часто при решении задачи линейной оптимизации бывает довольно сложно выполнять алгебраические преобразования над коэффициентами ограничений для поиска начального базиса. Для упрощения вычислений существует альтернативный метод решения, называемый **методом искусственного базиса**. Его суть заключается в том, что вместо того, чтобы искать базис среди имеющихся основных и дополнительных переменных, ввести так называемые **искусственные переменные**, которые сформируют начальный базис. Возможно, звучит сложно и непонятно, но сейчас мы всё объясним.

### Подготовительный этап

Аналогично базовому симплекс-методу для всех ограничений с неравенством вводятся дополнительные переменные, причём для ограничений с  $\geq$  они берутся с коэффициентом  $-1$ , а для ограничений с  $\leq$  с коэффициентом  $1$ . Ограничения с равенством остаются без изменений. Если свободный коэффициент какого-либо из ограничений меньше нуля, то такое ограничение умножается на  $-1$  (знак неравенства при этом меняется на противоположный). После этого приступают к поиску базиса.

#### ► [Пример 9](#)

### Формирование начального базиса

Для того, чтобы сформировать начальный базис в первую очередь можно поискать столбец, у которого одно значение равно единице, а все значения остальные значения равны нулю, и сделать соответствующую переменную базисной для этой строки. Однако такое случается довольно редко, поэтому проще сразу перейти к следующему пункту. Для всех ограничений, не имеющих базисной переменной, добавляем искусственную переменную с коэффициентом  $1$ . В целевую функцию добавляем эту же переменную с коэффициентов  $-M$ , если ищется

## Расчёт дельт и проверка плана на оптимальность

После того, как начальный базис сформирован необходимо вычислить дельты. Дельты вычисляются полностью аналогично базовому методу:  $\Delta_i = c_{e_1} \cdot a_{1i} + c_{e_2} \cdot a_{2i} + \dots + c_{e_m} \cdot a_{mi} - c_i$ . Единственным отличием будет тот факт, что результат может содержать значения с М. Когда дельты будут получены необходимо проверить текущий опорный план на оптимальность (см. проверку плана на оптимальность в базовом симплекс-методе). Если план оптимален, то алгоритм завершает свою работу, иначе формирует более оптимальное решение и повторяет процесс.

[▶ Пример 11](#)

### Programforyou

Programforyou — это сообщество, в котором Вы можете подтянуть свои знания по программированию, узнать, как эффективно решать те или иные задачи, а также воспользоваться нашими онлайн сервисами.

Полезное	Редактор	VK programforyou
Проекты	блок-схем	VK Андрей
Программы	Редактор графов	VK Светлана
	Калькуляторы	