# Двоично-десятичные коды

*Числа в кодах такого типа представляются двоичными тетрадами соответствующих десятичных цифр.*

**Двоично-десятичный код (2/10) – код прямого замещения; код 8421**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Разрешенные комбинации** | | | | **Запрещенные комбинации** | | | |
| **10 код** | **2/10 код** | **10 код** | **2/10 код** | **10 код** | **2/10 код** | **10 код** | **2/10 код** |
| 0 | 0000 | 5 | 0101 | 10 | 1010 | 15 | 1111 |
| 1 | 0001 | 6 | 0110 | 11 | 1011 |  |  |
| 2 | 0010 | 7 | 0111 | 12 | 1100 |  |  |
| 3 | 0011 | 8 | 1000 | 13 | 1101 |  |  |
| 4 | 0100 | 9 | 1001 | 14 | 1110 |  |  |

# Преобразование числа в обратный код

1. Запись отрицательного числа в прямом коде
2. Добавление тетрады + 0110 во все тетрады числа из п.1 - Сложение
3. Инверсия полученной в п.2. суммы - Это и есть результат – число в обратном коде.

# Преобразование числа в дополнительный код

1. Выполнить операции 1-3 из преобразования в обратный код
2. В младшую тетраду добавить + 0001 - Результат сложения – число в дополнительном коде.

# Преобразование в прямой код из обратного или дополнительного кода происходит аналогично

Например:

|  |  |
| --- | --- |
| Прямое преобразование | Обратное преобразование |
| 1. **1925 - прямой код**   0 0001 1001 0010 0101 (1925пр) | не требуется |
| 1. **-5689 – в обратный код:**   Преобразование в обратный код:  1 0101 0110 1000 1001 (-5689пр)  + 0110 0110 0110 0110 (поправка)  1 1011 1100 1110 1111 (далее инверсия)  **1 0100 0011 0001 0000 (-5689обр)** | **1 0100 0011 0001 0000 (-5689обр)**  + 0110 0110 0110 0110 (поправка)  1 1010 1001 0111 0110  (далее инверсия)  **1 0101 0110 1000 1001 (-5689пр)** |
| 1. **-1542 – в обратный код:**   Преобразование в обратный код:  1 0001 0101 0100 0010 (-1542пр)  + 0110 0110 0110 0110 (поправка)  1 0111 1011 1010 1000 (далее инверсия)  **1 1000 0100 0101 0111 (-1542обр)** | **1 1000 0100 0101 0111 (-1542обр)**  + 0110 0110 0110 0110 (поправка)  1 1110 1010 1011 1101  (далее инверсия)  **1 0001 0101 0100 0010 (-1542пр)** |
| 1. **-3567 – в дополнительный код:**   Преобразование в дополнительный код:  1 0011 0101 0110 0111 (-3567пр)  + 0110 0110 0110 0110 (поправка)  1 1001 1011 1100 1101 (далее инверсия)  1 0110 0100 0011 0010 (-3567обр)  + 1  **1 0110 0100 0011 0011 (-3567доп)** | **1 0110 0100 0011 0011 (-3567доп)**  + 0110 0110 0110 0110 (поправка)  1 1100 1010 1001 1001  (далее инверсия)  1 0011 0101 0110 0110  + 1  **1 0011 0101 0110 0111 (-3567пр)** |

# Правила выполнения арифметических операций

Вычитание заменяется сложением с отрицательным числом в дополнительном или обратном коде.

Сначала вычисляется сумма, а потом, если требуется – выполняется коррекция.

Коррекция результата потетрадного сложения путем добавления поправки + 0110 требуется в случае, если:

1. был перенос в старшую тетраду;
2. возникают запрещенные комбинации.

При коррекции разрешен межтетрадный перенос.

В примерах, приведенных ниже, запрещенные комбинации подчеркнуты, а межтетрадные переносы указаны знаком «<». При сложении удобно использовать модифицированные знаковые разряды: 00 – положительное число; 11 – отрицательное.

Например:

|  |  |
| --- | --- |
| Решение | Проверка |
| 1. **279+581=860**   00 0010 0111 1001 (279)  + 00 0101 1000 0001 (581)  00 0111 1111 1010  + 0110 0110 (корр –запрещ.комбин)  **00 1000 0110 0000 (860)** | не требуется |
| 1. **689+579=1268**   00 0000 0110 1000 1001 (689)  +00 0000 0101 0111 1001 (579)  +00 0000 1100<0000<0010 (корр-переносы+  0110 0110 0110 запрещ. комбинац.)  **00 0001 0010 0110 1000 (1268)** | не требуется |
| 1. **934 – 1239 = –305 сложение в обратном коде**   Перевод числа -1239 в обратный код:  11 0001 0010 0011 1001 (-1239пр)  + 0110 0110 0110 0110 (поправка)  11 0111 1000 1001 1111 (затем инверсия)  11 1000 0111 0110 0000 (-1239обр)  Сложение:  00 0000 1001 0011 0100 (934)  +11 1000 0111 0110 0000 (-1239обр)  11 1001<0000 1001 0100  + 0110 (корр- по переносу)  **11 1001 0110 1001 0100 (-305обр)** | Перевод числа в прямой код:  **1 1001 0110 1001 0100 (-305обр)**  0110 0110 0110 0110 (поправка)  1 1111 1100 1111 1010 (далее  инверсия)  **1 0000 0011 0000 0101 (-305пр)** |
| 1. **1239-934 = 305 сложение в обратном коде**   Перевод числа -934 в обратный код:  11 0000 1001 0011 1000 (­­–934пр)  + 0110 0110 0110 0110 (поправка)  11 0110 1111 1001 1010  11 1001 0000 0110 0101 (–934обр)  Сложение  00 0001 0010 0011 1001 (1239)  +11 1001 0000 0110 0101 (–934обр)  11 1010 0010 1001 1110  + 0110 0110 (коррекция запр.ком)  (корр-перенос в  00<0000 0010 1010 0100 знак.разряд (+1)+  00 0110 1 запрещ комбин.)  **00 0000 0011 0000 0101 (305)** | не требуется |
| 1. **-568-329=-897 сложение в дополнительном коде**   Перевод числа -568 в дополнительный код:  1 0101 0110 1000 (-568пр)  + 0110 0110 0110 (поправка)  1 1011 1100 1110 (далее инверсия)  1 0100 0011 0001  + 1  1 0100 0011 0010 (-568доп)  Перевод числа -329 в дополнительный код:  1 0011 0010 1001 (-329пр)  + 0110 0110 0110 (поправка)  1 1001 1000 1111 (далее инверсия)  1 0110 0111 0000  + 1  1 0110 0111 0001 (-329 доп)  Сложение:  11 0100 0011 0010 (-568доп)  11 0110 0111 0001 (-329 доп)  10 1010 1010 0011  + 0110 0110 (корр-запрещ.комб )  **11 0001 0000 0011 (-897доп)** | Перевод числа в прямой код:  1 0001 0000 0011 (-897доп)  + 0110 0110 0110 (поправка)  1 0111 0110 1001  (далее инверсия)  1 1000 1001 0110  + 1  **1 1000 1001 0111 (-897пр)** |
| 1. **636+985=1621**   00 0110 0011 0110 (636)  +00 1001 1000 0101 (985)  00 1111 1011 1011  + 0110 0110 0110 (корр-запрещ.комб.)  **01** 0110 0010 0001  |\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **(переполнение)** | не требуется |

##### Двоично-десятичный код 8421 с избытком 3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Разрешенные комбинации** | | | | **Запрещенные комбинации** | |
| **10 код** | **2/10 код** | **10 код** | **2/10 код** | **2/10 код** | **2/10 код** |
| 0 | 0011 | 5 | 1000 | 0000 | 1111 |
| 1 | 0100 | 6 | 1001 | 0001 |  |
| 2 | 0101 | 7 | 1010 | 0010 |  |
| 3 | 0110 | 8 | 1011 | 1101 |  |
| 4 | 0111 | 9 | 1100 | 1110 |  |

# Преобразование числа в обратный код

1. Запись отрицательного числа в прямом коде
2. Инверсия полученного в п.1. числа – результат – число в обратном коде.

# Преобразование числа в дополнительный код

1. Выполнить операции 1-2 из преобразования в обратный код
2. В младшую тетраду добавить + 0001 - Результат сложения – число в дополнительном коде.

# Преобразование в прямой код из обратного или дополнительного кода происходит аналогично

Например:

|  |  |
| --- | --- |
| Прямое преобразование | Обратное преобразование |
| 1. **1925 - прямой код**   0 0100 1100 0101 1000 (1925пр)  (**1**+3)(**9**+3)(**2**+3)(**5**+3) |  |
| 1. **-5689 – в обратный код:**   Преобразование в обратный код:  1 1000 1001 1011 1100 (-5689пр)  (далее инверсия)  **1 0111 0110 0100 0011 (-5689обр)** | 1 0111 0110 0100 0011 (-5689обр)  (далее инверсия)  **1 1000 1001 1011 1100 (-5689пр)**  (**5**+3)(**6**+3) (**8**+3)(**9**+3) |
| 1. **-1542 – в обратный код:**   Преобразование в обратный код:  1 0100 1000 0111 0101 (-1542пр)  (далее инверсия)  **1 1011 0111 1000 1010 (-1542обр)** | 1 1011 0111 1000 1010 (-1542обр)  (далее инверсия)  **1 0100 1000 0111 0101 (-1542пр)**  (**1**+3)(**5**+3)(**4**+3)(**2**+3) |
| 1. **-3567 – в дополнительный код:**   Преобразование в дополнительный код:  1 0110 1000 1001 1010 (-3567пр)  (далее инверсия)  1 1001 0111 0110 0101 (-3567обр)  + 1  **1 1001 0111 0110 0110 (-3567доп)** | 1 1001 0111 0110 0110 (-3567доп)  (далее инверсия)  1 0110 1000 1001 1001  + 1  **1 0110 1000 1001 1010 (-3567пр)**  (**3**+3)(**5**+3)(**6**+3)(**7**+3) |

# Правила выполнения арифметических операций

Вычитание заменяется сложением с отрицательным числом в дополнительном или обратном коде.

Сначала вычисляется сумма, а потом – коррекция.

Коррекция результата:

1. если при сложении не было переноса из анализируемой тетрады, то в нее надо добавить + 1101;

## если был перенос в старшую тетраду, то в нее надо добавить + 0011;

1. если получена неправильная тетрада, то в нее надо добавить + 0110.

При коррекции межтетрадный перенос блокирован.

В примерах, приведенных ниже, запрещенные комбинации подчеркнуты, а межтетрадные переносы указаны знаком «<». При сложении будем использовать модифицированные знаковые разряды: 00 – положительное число; 11 – отрицательное.

Например:

|  |  |
| --- | --- |
| Решение | Проверка |
| 1. **279+581=860**   00 0101 1010 1100 (279)  +00 1000 1011 0100 (581)  00 1110<0110<0000  + 1101 0011 0011 (корр–по переносам)  **00 1011 1001 0011 (860)**  (8+3)(6+3)(0+3) | не требуется |
| 1. **689+579=1268**   00 0011 1001 1011 1100 (0689)  +00 0011 1000 1010 1100 (0579)  +00 0111<0010<0110<1000  1101 0011 0011 0011 (корр-переносы)  00 0100 0101 1001 1011  (1+3)(2+3)(6+3)(8+3) | не требуется |
| 1. **934 – 1239 = –305 сложение в обратном коде**   Перевод числа -1239 в обратный код:  11 0100 0101 0110 1100 (-1239пр)  (затем инверсия)  11 1011 1010 1001 0011 (-1239обр)  Сложение:  00 0011 1100 0110 0111 (0934)  +11 1011 1010 1001 0011 (-1239обр)  11 1111<0110 1111 1010  + 1101 0011 1101 1101 (корр-переносы)  **11 1100 1001 1100 0111 (-305обр)** | **1 1100 1001 1100 0111 (-305обр)**  (далее инверсия)  **1 0011 0110 0011 1000 (-305пр)**  **(0+3)(3+3)(0+3)(5+3)** |
| 1. **1239-934 = 305 сложение в обратном коде**   Перевод числа -934 в обратный код:  11 0011 1100 0110 0111 (­­–934пр)  (затем инверсия)  11 1100 0011 1001 1000 (–934обр)  Сложение  00 0100 0101 0110 1100 (1239)  +11 1100 0011 1001 1000 (–934обр)  00<0000 1001<0000<0100  + 0011 1101 0011 0011 (корр-переносы)  1 (корр-перенос  в знак.разряд)  **00 0011 0110 0011 1000 (0305)**  (**0**+3)(**3**+3)(**0**+3)(**5**+3) | не требуется |
| 1. **-568-329=-897 сложение в дополнительном коде**   Перевод числа -568 в дополнительный код:  11 1000 1001 1011 (-568пр)  (далее инверсия)  11 0111 0110 0100  + 1  11 0111 0110 0101 (-568доп)  Перевод числа -329 в дополнительный код:  11 0110 0101 1100 (-329пр)  (далее инверсия)  11 1001 1010 0011 (-329 обр)  + 1  11 1001 1010 0100 (-329 доп)  Сложение:  11 0111 0110 0101 (-568доп)  +11 1001 1010 0100 (-329 доп)  11<0001<0000 1001  + 0011 0011 1101 (корр-переносы)  **11 0100 0011 0110 (-897доп)** | **1 0100 0011 0110 (-897доп)**  (далее инверсия)  1 1011 1100 1001  + 1  **1 1011 1100 1010 (-897пр)**  **(8+3)(9+3)(7+3)** |
| 1. **636+985**   00 1001 0110 1001 (636)  +00 1100 1011 1000 (985)  01<0110<0010<0001  + 0011 0011 0011 (корр-переносы)  **01** 1001 0101 0100  |\_\_\_\_\_\_\_\_\_\_\_\_ **(положительное переполнение)** | не требуется |

##### Система счисления по основанию 3 с весами +1, 0, -1 ()

Существуют позиционные системы счисления, специально созданные для упрощения или ускорения вычислений на ЭВМ. Их достоинством является простота алгоритмов выполнения арифметических операций. Недостаток – необходимость перевода из классических позиционных систем счисления в специальные. Поэтому их применяют в вычислениях, в которых не требуется изменять систему счисления при вводе и выводе данных, либо это преобразование выполняется достаточно просто.

Рассмотрим, в качестве примера, троичную уравновешенную систему счисления с цифрами +1, 0, -1. Для обозначения -1 далее используется символ .

Примеры записи чисел в этой системе:

10 = 32⋅1+30⋅= 9–1=8

1 = 33⋅1+32⋅+31⋅= 27–9–3+1/3–1/9 = 15

110, = 33⋅+32⋅1+31 = -27+9+3-1/3+1/9 = –15

Достоинства данной уравновешенной троичной системы:

1. знак числа задается его старшей ненулевой цифрой
2. переход к числу с противоположным знаком выполняется заменой 1 на и наоборот
3. операция округления сводится к отбрасыванию дробной части

**Суммирование** в этой системе счисления выполняется по правилам:

|  |  |  |  |
| --- | --- | --- | --- |
| + |  | 0 | 1  **Вычитание** сводится к изменению знака на противоположный у вычитаемого и последующему суммированию.  **Умножение** на 1 выполняется обычным образом, при умножении на знак частного произведения меняется на противоположный |
|  | 1 |  | 0 |
| 0 |  | 0 | 1 |
| 1 | 0 | 1 | 1 |

Перевод в уравновешенную троичную систему осуществляется по известным правилам перевода дробной и целой частей числа умножением и делением этого числа на основание системы счисления 3.

**Пример.** Перевести число 230,53 в уравновешенную троичную систему счисления:

Переводим целую часть (учтите, что остатки 2 не должны получаться):

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 230 |  | 3 |  |  |  |  |  |  |  |  |
| -1 |  | 77 |  | 3 |  |  |  |  |  |  |
|  |  | -1 |  | 26 |  | 3 |  |  |  |  |
|  |  |  |  | -1 |  | 9 |  | 3 |  |  |
|  |  |  |  |  |  | 0 |  | 3 |  | 3 |
|  |  |  |  |  |  |  |  | 0 |  | 1 |
|  |  |  |  |  |  |  |  |  |  |  |

23010 = 100 (-1,0,1)

Можно перевести в обычную троичную систему счисления (с цифрами 0, 1, 2), а потом сделать поправки, заменив 2 на 1:

Поправки (заменяем 2 на 1, при этом 1 суммируется в старшем разряде):

2 2 1 1 ~~2~~

+ 1\_\_

2 2 1 ~~2~~

+ 1\_\_\_

2 2 ~~2~~

+ 1\_\_\_\_\_

1 0 0

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 230 |  | 3 |  |  |  |  |  |  |
| 2 |  | 76 |  | 3 |  |  |  |  |
|  |  | 1 |  | 25 |  | 3 |  |  |
|  |  |  |  | 1 |  | 8 |  | 3 |
|  |  |  |  |  |  | 2 |  | 2 |
|  |  |  |  |  |  |  |  |  |

Результат тот же - 23010 = 100 (-1,0,1)

Переводим дробную часть, сначала переведем в обычную троичную систему счисления, затем сделаем поправки:

Поправки (заменяем 2 на 1, при этом 1 суммируется в старшем разряде):

0, 1 1 ~~2~~

+ 1 \_\_

0, 1 ~~2~~

+ 1 \_\_\_\_

0, ~~2~~

+ 1 \_\_\_\_\_\_

1, 0,5310 = 0,1123=1, (-1,0,1)

3

|  |  |
| --- | --- |
| 0, | 53  3 |
| 1 | 59  3 |
| 1 | 77  3 |
| 2 | 31  3 |
| 0 | 93 |

Чтобы получить окончательный ответ, просуммируем целую и дробную части:

100,000

+ 1,

100,

Ответ: 230,5310 = 100,(-1,0,1)

##### Система остаточных классов (СОК)

Органическим недостатком любой позиционной системы счисления является наличие межразрядных связей. Действительно, результат сложения в i -м разряде зависит не только от значений i -х разрядов слагаемых, но и от переноса из i-1 разряда и, в конечном итоге – от значений всех младших разрядов слагаемых: i-1, i-2, ..., 1, 0. Поэтому вычисление разрядов суммы может проходить только последовательно (с учетом формирования переноса из предыдущего (младшего) разряда). Это обстоятельство препятствует распараллеливанию процесса вычисления и, естественно, снижает быстродействие процессора.

В рамках позиционных систем счисления известно несколько способов логического и схемотехнического ускорения арифметических операций — параллельный перенос, матричная и табличная арифметика и др., однако все они требуют весьма значительных аппаратных затрат.

Поиск новых путей построения арифметических устройств ЭВМ, позволяющий исключить зависимость между разрядами при выполнении арифметических операций, привел к применению в машинной арифметике аппарата теории вычетов – одного из разделов теории чисел. В рамках этого аппарата разработана *непозиционная* система счисления – система счисления в остаточных классах (СОК).

**Система остаточных классов (СОК)** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) residue number system) – [система счисления](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%81%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F), основанная на [*модулярной арифметике*](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D1%83%D0%BB%D1%8F%D1%80%D0%BD%D0%B0%D1%8F_%D0%B0%D1%80%D0%B8%D1%84%D0%BC%D0%B5%D1%82%D0%B8%D0%BA%D0%B0).

Представление числа в системе остаточных классов основано на понятии [вычета](https://ru.wikipedia.org/wiki/%D0%A1%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5_%D0%BF%D0%BE_%D0%BC%D0%BE%D0%B4%D1%83%D0%BB%D1%8E) и [китайской теореме об остатках](https://ru.wikipedia.org/wiki/%D0%9A%D0%B8%D1%82%D0%B0%D0%B9%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B5%D0%BC%D0%B0_%D0%BE%D0%B1_%D0%BE%D1%81%D1%82%D0%B0%D1%82%D0%BA%D0%B0%D1%85). СОК определяется

а) набором *попарно* [*взаимно простых*](https://ru.wikipedia.org/wiki/%D0%92%D0%B7%D0%B0%D0%B8%D0%BC%D0%BD%D0%BE_%D0%BF%D1%80%D0%BE%D1%81%D1%82%D1%8B%D0%B5_%D1%87%D0%B8%D1%81%D0%BB%D0%B0) *модулей* ( m1 , m2 , … , mn ) таких, что наибольший общий делитель НОД (mi, mj ) = 1 (i, j = 0 , 1 , … , n ;   i ≠ j ), называемых базисом, и

b) произведением M = m1 ⋅ m2 ⋅ … ⋅ mn, опреляющим диапазон чисел.

Каждому целому числу X∈[ 0 , M − 1 ] ставится в соответствие набор *вычетов* (x1 , x2 , …, xn), где:

x1 ≡ (X mod m1) - остаток от деления X на m1;

x2 ≡ (X mod m2) - остаток от деления X на m2;

. . .

xn ≡ (X (mod mn) - остаток от деления X на mn

[Китайская теорема об остатках](https://ru.wikipedia.org/wiki/%D0%9A%D0%B8%D1%82%D0%B0%D0%B9%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B5%D0%BC%D0%B0_%D0%BE%D0%B1_%D0%BE%D1%81%D1%82%D0%B0%D1%82%D0%BA%D0%B0%D1%85) гарантирует однозначность (единственность) представления целых неотрицательных чисел из отрезка [ 0 ,   M − 1 ] .

## **Преимущества системы остаточных классов**

В СОК арифметические операции (сложение, вычитание, умножение, деление) выполняются покомпонентно, если про результат известно, что он является целочисленным и также лежит в   
[ 0 ,   M − 1 ] .

(z1, z2, …, zn) = (x1, x2, …, xn) ⊗ (y1, y2, …, yn), где

⊗ - обозначение операции умножения, сложения, вычитания и деления;

z1 = (x1 ⊗ y1) mod m1;

z2 = (x2 ⊗ y2) mod m2;

. . .

zn = (xn ⊗ yn) mod mn;

На деление накладываются дополнительные ограничения. Деление должно быть целочисленным, то есть делитель должен нацело делить делимое. Делитель должен быть взаимопростым со всеми модулями базиса.

## **Недостатки системы остаточных классов**

* отсутствие эффективных алгоритмов для сравнения чисел; обычно, сравнение осуществляется через перевод аргументов из СОК в систему счисления (полиадическую) со смешанными основаниями: m 1 , m 1 m 2 , … , m 1 m 2 … m n − 1 {\displaystyle m\_{1},m\_{1}m\_{2},\dots ,m\_{1}m\_{2}\dots m\_{n-1}} m1, m1m2, …, m1m2mn;
* медленные алгоритмы взаимного преобразования представления чисел из позиционной системы счисления в СОК и обратно;
* сложные алгоритмы деления (когда результат не является целым);
* трудность в обнаружении переполнения.

## **Применение системы остаточных классов**

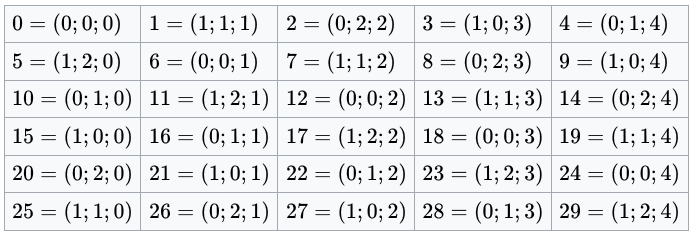
СОК широко используется в теории чисел, криптографии и других дисциплинах. а также в микроэлектронике в специализированных устройствах [ЦОС](https://ru.wikipedia.org/wiki/%D0%A6%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%B0%D1%8F_%D0%BE%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0_%D1%81%D0%B8%D0%B3%D0%BD%D0%B0%D0%BB%D0%BE%D0%B2), где требуется:

* контроль за ошибками за счет введения дополнительных избыточных модулей;
* высокая скорость работы, которую обеспечивает параллельная реализация базовых арифметических операций;
* [информационная безопасность](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B5%D0%B7%D0%BE%D0%BF%D0%B0%D1%81%D0%BD%D0%BE%D1%81%D1%82%D1%8C).

Подробности см. в <https://ru.wikipedia.org/wiki/%D0%9A%D0%B8%D1%82%D0%B0%D0%B9%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B5%D0%BC%D0%B0_%D0%BE%D0%B1_%D0%BE%D1%81%D1%82%D0%B0%D1%82%D0%BA%D0%B0%D1%85>

## **Пример**

Рассмотрим СОК с базисом {2, 3, 5}( 2 ; 3 ; 5 ) {\displaystyle (2;3;5)} . В этом базисе можно взаимно-однозначно представить числа из промежутка от 0 {\displaystyle 0} 0 до 29 {\displaystyle 29} 29, так как M = 2 × 3 × 5 = 30 {\displaystyle M=2\times 3\times 5=30} M=2\*3\*5. Таблица соответствия чисел из позиционной системы счисления и системы остаточных классов:



### **Пример сложения**

Сложим два числа 9 и 14 в базисе ( 2 ; 3 ; 5 ) {\displaystyle (2;3;5)} {2,3,5}. Их представление в заданном базисе: 9=(1;0;4) 9 = ( 1 ; 0 ; 4 ) {\displaystyle 9=(1;0;4)} и 14 = ( 0 ; 2 ; 4 ) {\displaystyle 14=(0;2;4)} 14=(0;2;4) (см. табличку выше): ( z 1 , z 2 , z 3 ) = ( 1 , 0 , 4 ) + ( 0 , 2 , 4 ) {\displaystyle (z\_{1},z\_{2},z\_{3})=(1,0,4)+(0,2,4)}

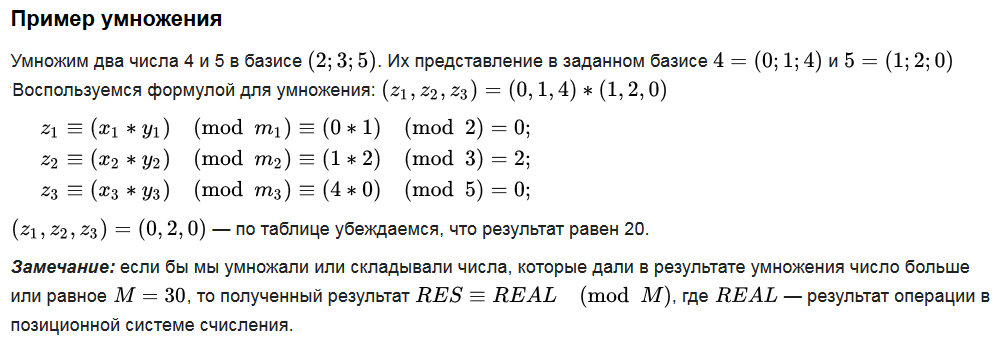
(z1, z2, z3) = (1,0,4) + (0,2,4) = (1, 2, 3), где

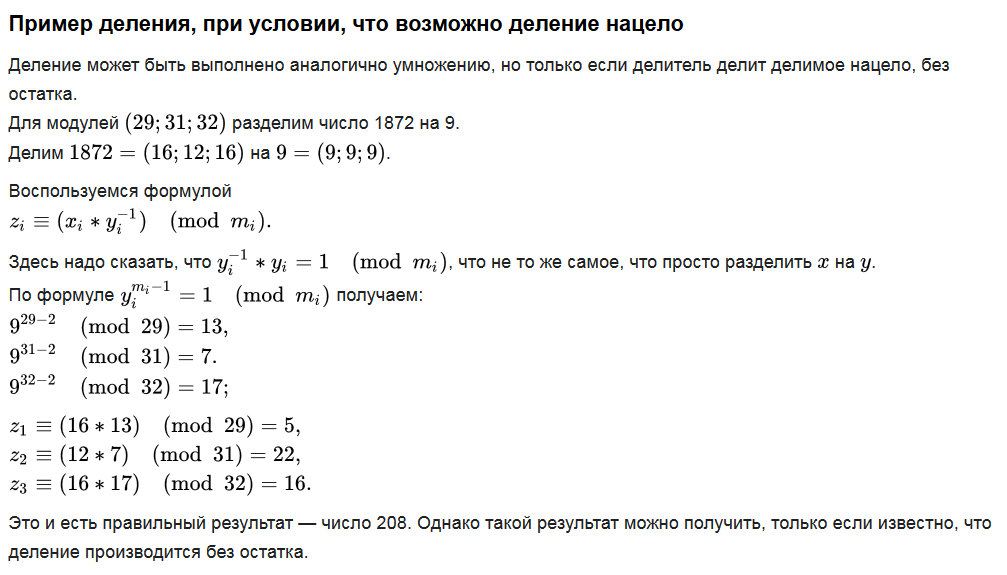
z1 = (x1 + y1) mod m1 = (1+0) mod 2 = 1;

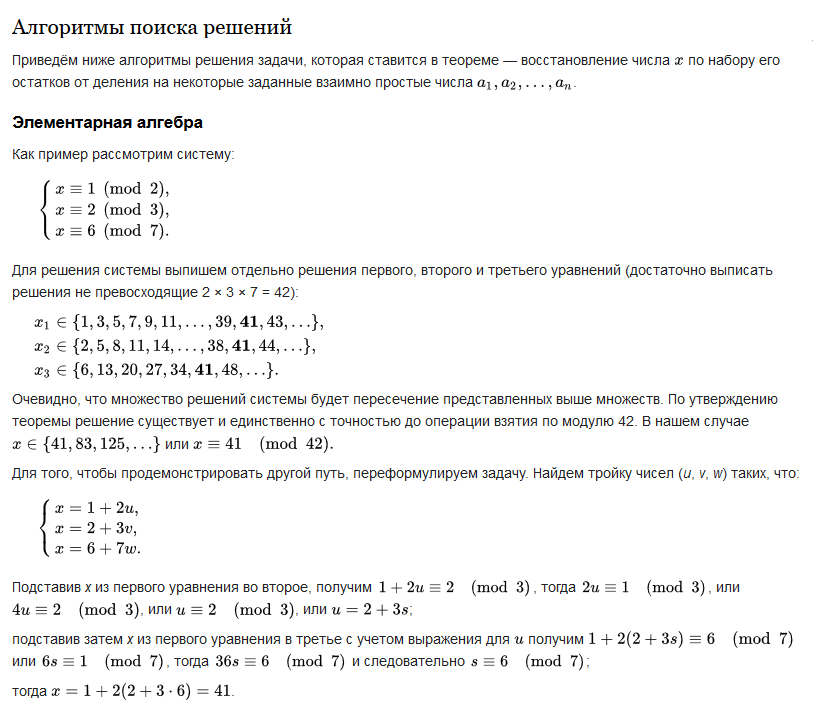
z2 = (x2 + y2) mod m2 = (0+2) mod 3 = 2;

z3 = (x3 + y3) mod m3 = (4+4) mod 5 = 3.

(1, 2, 3) по таблице убеждаемся, что результат равен 23.







**Применение СОК.**

Модулярная арифметика широко применяется в теории чисел, криптографии и других дисциплинах. Подробности см. в <https://ru.wikipedia.org/wiki/%D0%9A%D0%B8%D1%82%D0%B0%D0%B9%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B5%D0%BC%D0%B0_%D0%BE%D0%B1_%D0%BE%D1%81%D1%82%D0%B0%D1%82%D0%BA%D0%B0%D1%85>

**Домашнее задание.**

1. Выполнить алгебраическое сложение и вычитание двоично-десятичных чисел в кодировке:
2. 8421 и
3. 8421+3.

Для четных вариантов операнды представлять в *обратном коде*, для нечетных – *в дополнительном*.

1. Выполнить эти же операции в *уравновешенной троичной системе*, используя для представления чисел столько троичных разрядов, сколько необходимо, чтобы не было переполнения.
2. Выполнить эти же вычисления в СОК {7,13,17,19}, для проверки результата восстановить X по алгоритму поиска решений

Вариант выбирается по формуле K=(N-1) mod S +1, где N-ваш номер в списке группы, S- число вариантов, K-номер варианта

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Вариант** | **Задание** | **Вариант** | **Задание** | **Вариант** | **Задание** | **Вариант** | **Задание** |
| **1** | **563, -759** | **8** | **-287, -278** | **15** | **544, -691** | **22** | **-441, 643** |
| **2** | **346, -287** | **9** | **344, -635** | **16** | **-599, 143** | **23** | **-378, 289** |
| **3** | **-60, -678** | **10** | **-234, 556** | **17** | **487, -406** | **24** | **-627, 450** |
| **4** | **6754, -123** | **11** | **463, -528** | **18** | **-1745, 176** | **25** | **2427, -1831** |
| **5** | **675, -1223** | **12** | **491, -518** | **19** | **-754, 116** | **26** | **929, -1701** |
| **6** | **754, -1230** | **13** | **583, -528** | **20** | **-2745, 176** | **27** | **427, -1831** |
| **7** | **9876, -4656** | **14** | **-607, +929** | **21** | **701, -354** | **28** | **386, -127** |

Контрольные вопросы

Литература