

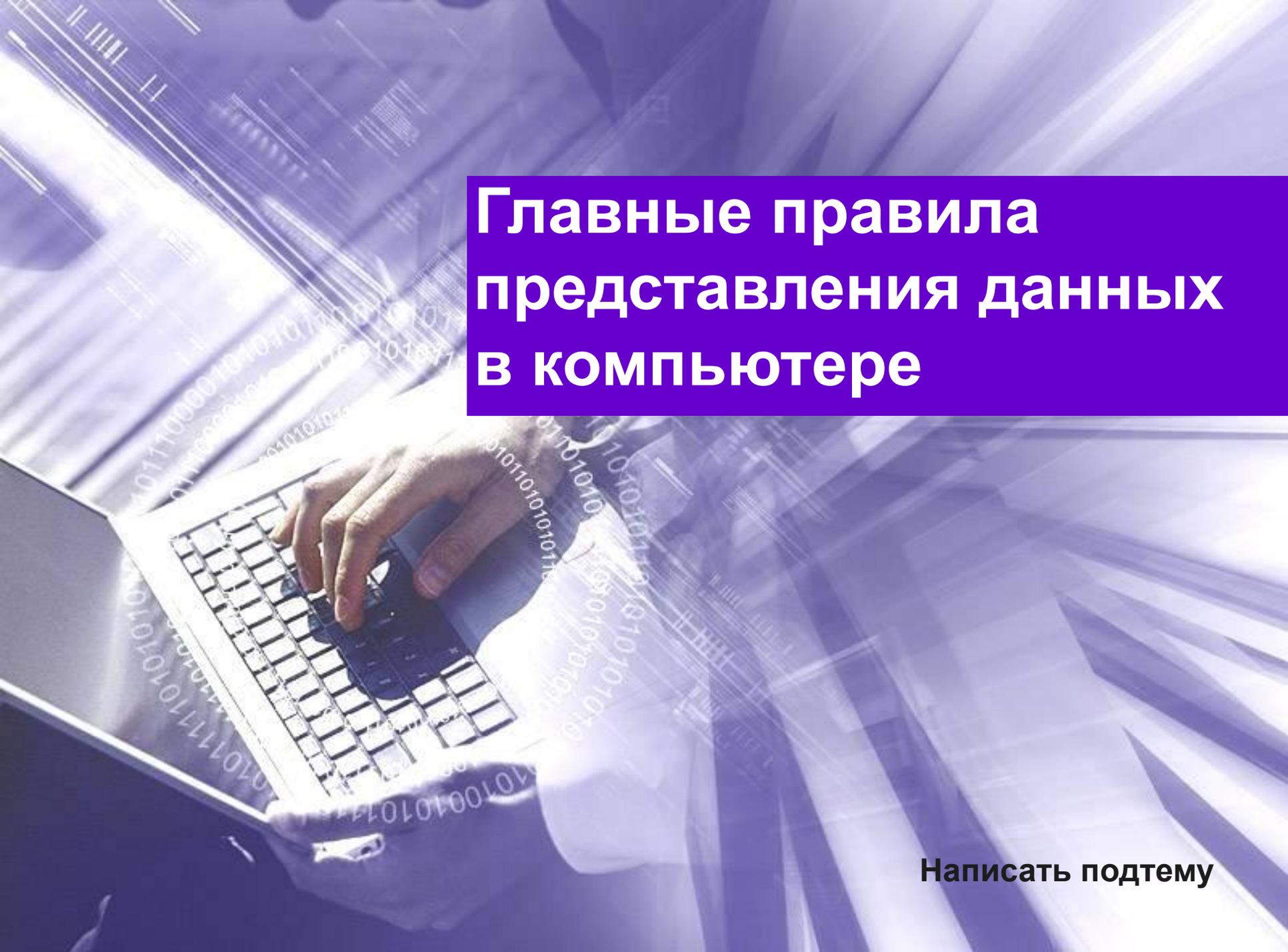
# Представление чисел в памяти компьютера

Написать тему

Не писать

# Образ компьютерной памяти

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



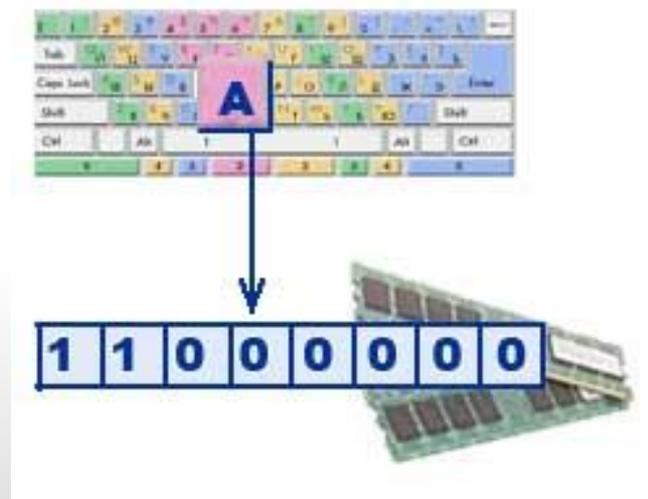
# Главные правила представления данных в компьютере

Написать подтему



# Правило 1

*Данные (и программы) в памяти компьютера хранятся в двоичном виде, т. е. в виде цепочек единиц и нулей.*



Написать всё



## Правило 2

*Представление данных в компьютере  
**дискретно.***

Дискретное множество состоит из отделенных друг от друга элементов.

Написать всё



## Правило 3

*Множество представимых в памяти компьютера величин **ограничено** и **конечно**.*

**МАТЕМАТИКА:**  
множество целых чисел **дискретно, бесконечно, не ограничено**

**ИНФОРМАТИКА:**  
множество целых чисел **дискретно, конечно, ограничено**

Написать всё



## Правило 4

*В памяти компьютера числа хранятся в двоичной системе счисления.*



**Написать всё**

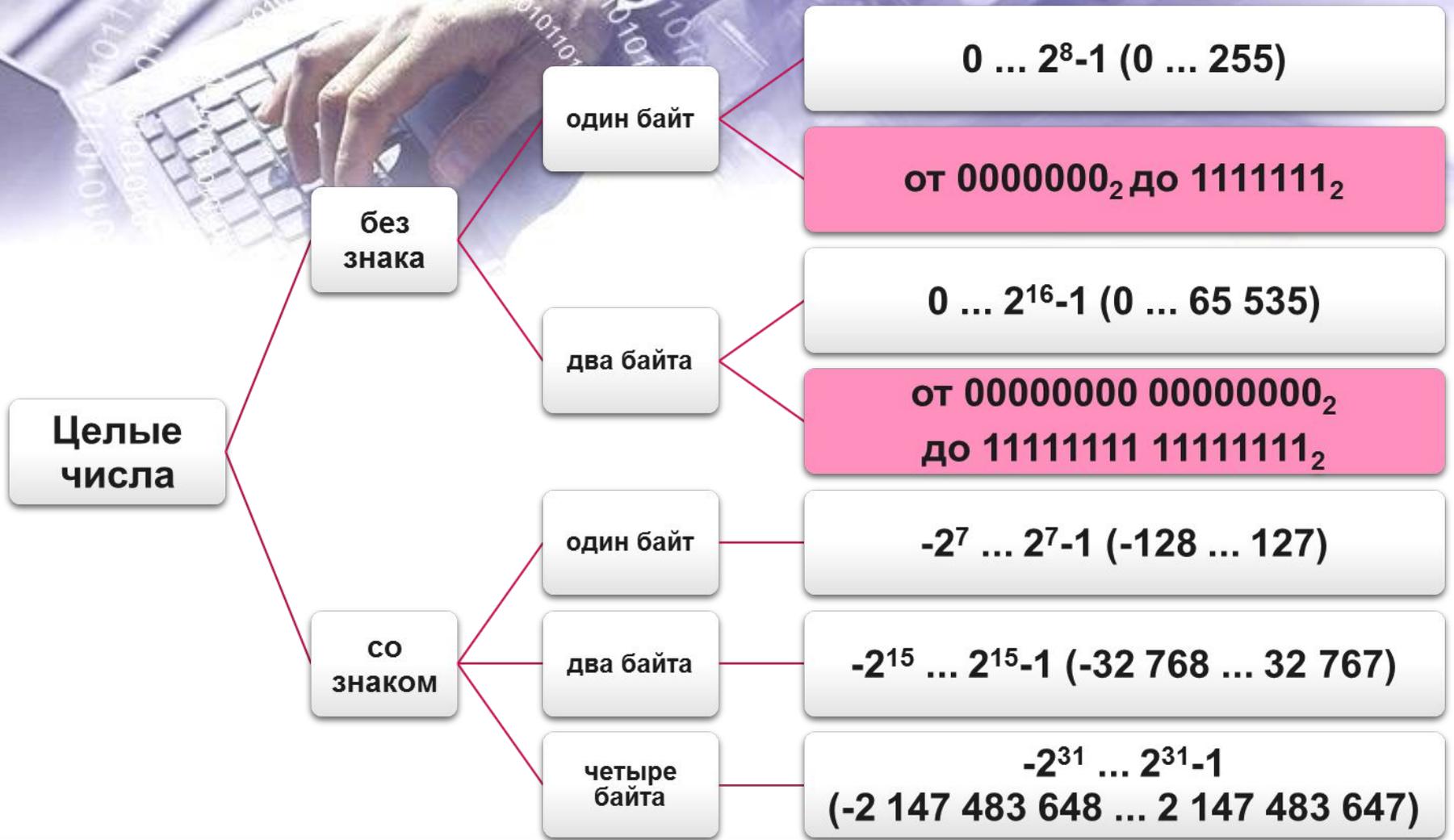


# Числовые величины

**Целые**  
(формат с  
фиксированной  
запятой)

**Вещественные**  
(формат с  
плавающей запятой)

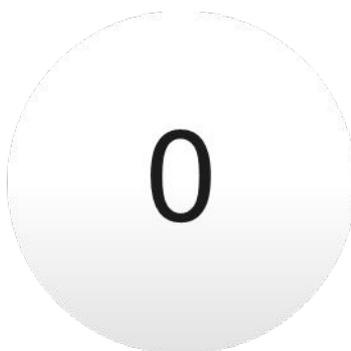
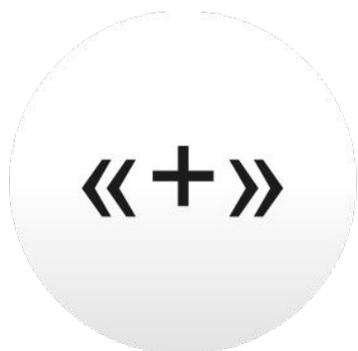
Написать всё



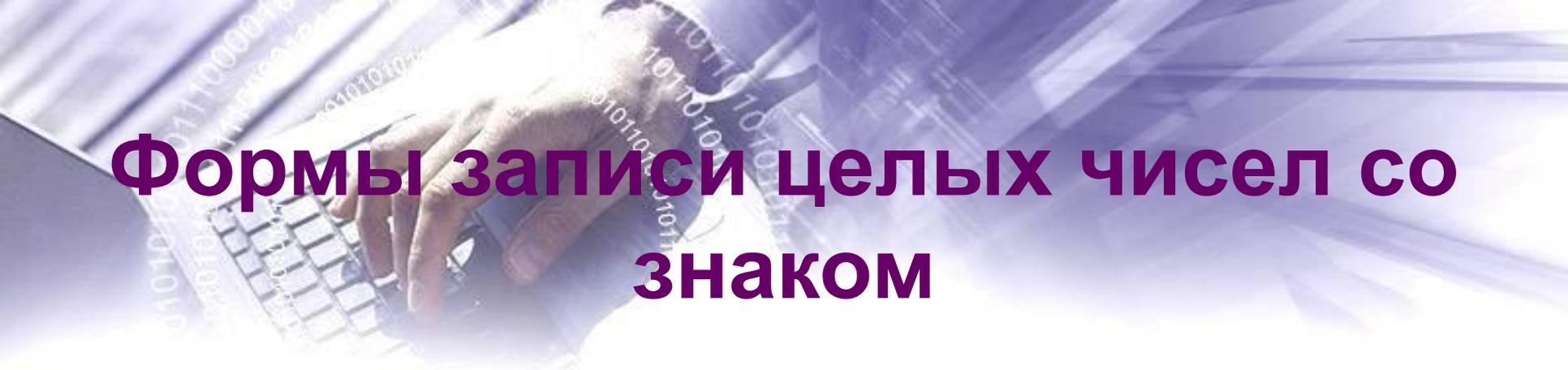
Написать всё



**Самый левый (старший) разряд содержит информацию о знаке числа**



**Написать всё**



# Формы записи целых чисел со знаком

Прямой код

Обратный код

Дополнительный код

Написать всё

# Формы записи чисел целых чисел со знаком

Написать всё

## Положительное число

прямой  
код

обратный  
код

дополнительный  
код

имеют одинаковое  
представление

Число  $19_{10} = 10011_2$

прямой, обратный и дополнительный  
код

0	0	0	1	0	0	1	1
---	---	---	---	---	---	---	---

«+»

Число  $127_{10} = 1111111_2$

прямой, обратный и дополнительный  
код

0	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

«+»

# Формы записи чисел целых чисел со знаком

Написать всё

## Отрицательное число

прямой  
код

обратный  
код

дополнительный  
код

имеют разное  
представление

Прямой код числа  
-19:

1	0	0	1	0	0	1	1
«-»							

Прямой код числа  
-127:

1	1	1	1	1	1	1	1
«-»							

# Формы записи чисел целых чисел со знаком

Написать всё

- Обратный код получается инвертированием всех цифр двоичного кода абсолютной величины числа, включая разряд знака: нули заменяются единицами, а единицы – нулями.

**Число -19:**

Код модуля числа: 0 0010011

Обратный код числа: 1 1101100

1	1	1	0	1	1	0	0
---	---	---	---	---	---	---	---

**Число -127:**

Код модуля числа: 0 1111111

Обратный код числа: 1 0000000

1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

- «Дополнительный код» получается образованием обратного кода с последующим прибавлением единицы к его младшему разряду.

**Дополн. код числа -19:**

1	1	1	0	1	1	0	1
---	---	---	---	---	---	---	---

«-»

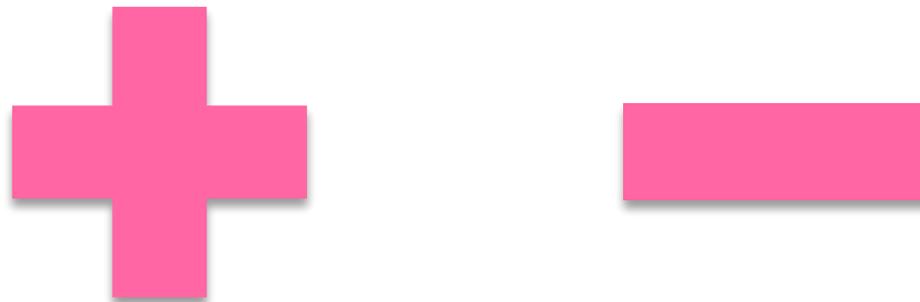
**Дополн. код числа -127:**

1	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

«-»

# Арифметические действия

В большинстве компьютеров операция вычитания не выполняется. Вместо неё производится сложение уменьшаемого с обратным или дополнительным кодом вычитаемого.



При сложении **дополнительных кодов** чисел А и В имеют место четыре случая.

Написать всё

# Арифметические действия

- 1) *A и B положительные:*

Десятичная  
запись:

Двоичные коды:

	5	$A_{\text{пк}}$	0	0	0	0	0	1	0	1
+	7	$B_{\text{пк}}$	0	0	0	0	0	1	1	1
	12	$C_{\text{пк}}$	0	0	0	0	1	1	0	0

Написать всё

# Арифметические действия

- 2)  $A$  – положительное,  $B$  – отрицательное,  $|B| > |A|$

Десятичная  
запись:

Двоичные коды:

	5	$A_{\text{пк}}$	0	0	0	0	0	1	0	1
+	-12	$B_{\text{дк}}$	1	1	1	1	0	1	0	0
	-7	$C_{\text{дк}}$	1	1	1	1	1	0	0	1

При переводе в прямой код биты цифровой части результата инвертируются и к младшему разряду прибавляется единица:

$$10000110 + 1 = 10000111 = -7_{10}$$

Написать всё

# Арифметические действия

- 3)  $A$  – положительное,  $B$  – отрицательное,  $|B| < |A|$

Десятичная  
запись:

+ 12  
-5  
7

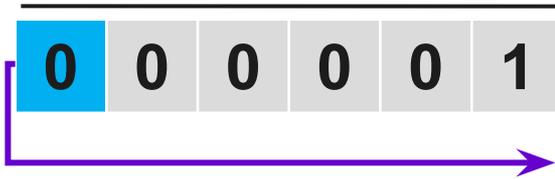
$A_{\text{пк}}$

$B_{\text{дк}}$

$C_{\text{пк}}$

Двоичные коды:

0	0	0	0	1	1	0	0
1	1	1	1	1	0	1	1
<hr/>							
0	0	0	0	0	1	1	1



Перенос отбрасывается

Единицу переноса из знакового разряда компьютер отбрасывает.

Написать всё

# Арифметические действия

- 4) *A и B отрицательные*

Десятичная  
запись:

**-5**  
+

**-7**

**-12**

**A**<sub>дк</sub>

**B**<sub>дк</sub>

**C**<sub>дк</sub>

Двоичные коды:

**1** 1 1 1 1 0 1 1

**1** 1 1 1 1 0 0 1

**1** 1 1 1 0 1 0 0

Перенос отбрасывается

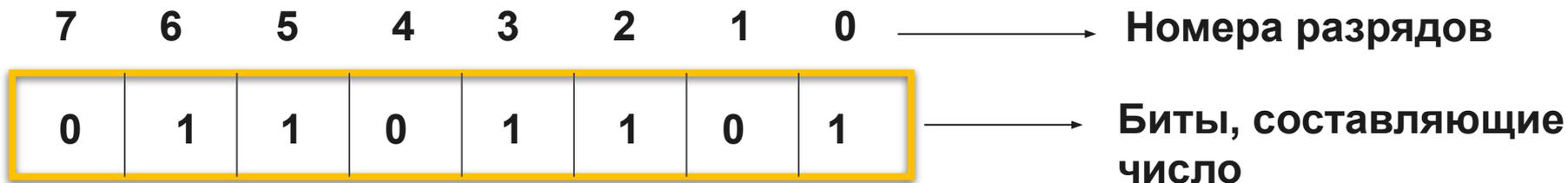
При переводе в прямой код биты цифровой части результата инвертируются и к младшему разряду прибавляется единица:

$$10001011 + 1 = 10001100 = -12_{10}$$

Написать всё

# Целые числа без знака

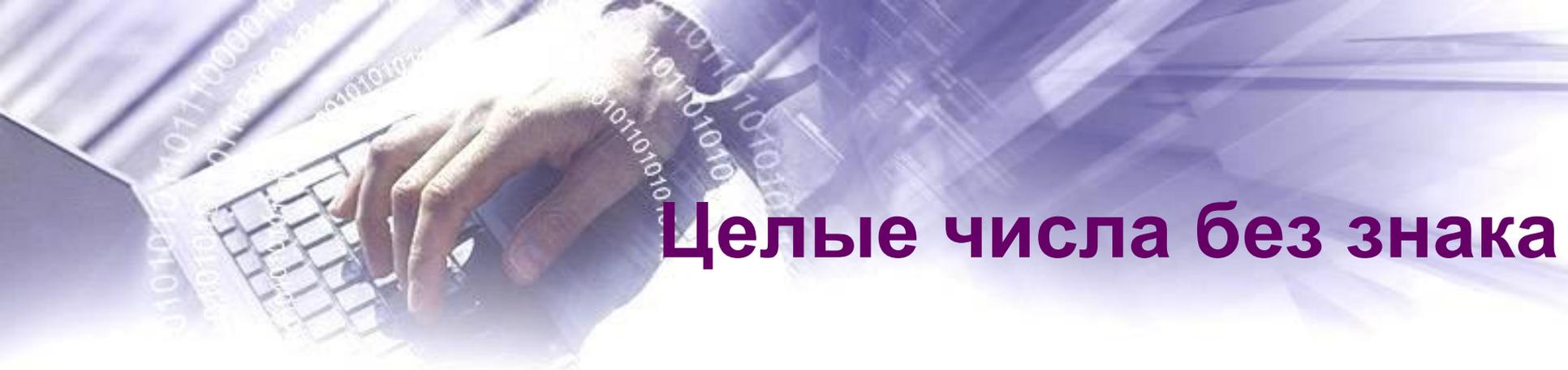
Для хранения *целых неотрицательных чисел без знака* отводится **одна ячейка памяти (8 битов)**.



$$11111111_2 = 100000000_2 - 1 = 2^8 - 1 = 255_{10}$$

Для **n-разрядного** представления максимальное целое неотрицательное число равно  **$2^n - 1$** .

Написать всё



# Целые числа без знака

**Пример.** Представить число  $51_{10}$  в двоичном виде в восьмибитовом представлении в формате целого без знака.

Решение.

$$51_{10} = 110011_2$$

0	0	1	1	0	0	1	1
---	---	---	---	---	---	---	---

Написать всё

# Целые числа со знаком

Для хранения **целых чисел со знаком** отводится две ячейки памяти (16 битов).

**Старший разряд** числа определяет его знак. Если он равен 0, число положительное, если 1, то отрицательное.

$$51_{10} = 110011_2$$



$$-51_{10} = -110011_2$$



Такое представление чисел в компьютере называется **прямым кодом**.

Написать всё



# Целые числа со знаком

Для **n-разрядного** представления со знаком (с учетом выделения одного разряда на знак):

- минимальное отрицательное число равно  $-2^{n-1}$
- максимальное положительное число равно  $2^{n-1} - 1$ ,

Целые числа в памяти компьютера —  
это **дискретное, ограниченное и конечное**  
множество.

Написать всё

Написать всё

# Целые числа со знаком

Для представления отрицательных целых чисел используется **дополнительный код**.

**Алгоритм получения дополнительного кода отрицательного числа:**

1. Число записать **прямым кодом** в  $n$  двоичных разрядах.
2. Получить **обратный код** числа, для этого значения всех битов инвертировать, кроме старшего разряда.
3. К полученному обратному коду **прибавить единицу**.

Представить число  $-2014_{10}$  в двоичном виде в шестнадцатибитном представлении в формате целого со знаком.

Прямой код	$-2014_{10}$	10000111 11011110 <sub>2</sub>
Обратный код	Инвертирование	11111000 00100001 <sub>2</sub>
	Прибавление единицы	11111000 00100001 <sub>2</sub> 00000000 00000001 <sub>2</sub>
Дополнительный код		11111000 00100010 <sub>2</sub>

# Целые числа со знаком

## Алгебраическое сложение двоичных чисел

1. Положительные слагаемые представить в прямом коде.
2. Отрицательные слагаемые – в дополнительном.
3. Найти сумму кодов, включая знаковые разряды, которые при этом рассматриваются как старшие разряды. При переносе из знакового разряда единицу переноса отбрасывают.
4. В результате получают алгебраическую сумму в прямом коде, если эта сумма положительная, и в дополнительном, если сумма отрицательная.

# Целые числа со знаком

**Пример 1.** Найти разность  $13_{10} - 12_{10}$  в восьмибитном представлении.

	$13_{10}$	$-12_{10}$
Прямой код	00001101	10001100
Обратный код	-	11110011
Дополнительный код	-	11110100

$$\begin{array}{r} + \quad 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \\ \quad \underline{1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0} \\ 10 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \end{array}$$

Так как произошел перенос из знакового разряда, первую единицу отбрасываем, и в результате получаем 00000001.

**Написать всё**

# Целые числа со знаком

**Пример 2.** Найти разность  $8_{10} - 13_{10}$  в восьмибитном представлении.

	$8_{10}$	$-13_{10}$
Прямой код	00001000	10001101
Обратный код	-	11110010
Дополнительный код	-	11110011

$$\begin{array}{r} + \quad 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \\ \quad 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \\ \hline 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 1 \end{array}$$

Написать всё

Написать всё

## Целые числа со знаком

**Пример 2.** Найти разность  $8_{10} - 13_{10}$  в восьмибитном представлении.

$$\begin{array}{r} 00001000 \\ + 11110011 \\ \hline 11111011 \end{array}$$

В знаковом разряде стоит 1, значит результат получен в дополнительном коде. Прейдем от дополнительного кода к обратному, вычтя единицу:

$$\begin{array}{r} 11110011 \\ - 00000001 \\ \hline 11111010 \end{array}$$

Прейдем от обратного кода к прямому, инвертируя все цифры, за исключением знакового (старшего) разряда:  $10000101_2 = 5_{10}$ .



# Вещественные числа

Вещественные числа хранятся и обрабатываются в компьютере в формате *с плавающей запятой*, использующем экспоненциальную форму записи чисел.

$$A = M \cdot q^n$$

$M$  – мантисса числа (правильная отличная от нуля дробь),

$q$  – основание системы счисления,

$n$  – порядок числа.

Диапазон ограничен максимальными значениями  $M$  и  $n$ .

Написать всё



Написать всё

## Вещественные числа

Например,  $123,45 = 0,12345 \cdot 10^3$

Порядок указывает, на какое количество позиций и в каком направлении должна сместиться десятичная запятая в мантиссе.

Число в формате с плавающей запятой может занимать в памяти 4 байта (*обычная точность*) или 8 байтов (*двойная точность*).

При записи числа выделяются разряды для хранения знака мантиссы, знака порядка, порядка и мантиссы.

Мантисса **M** и порядок **n** определяют диапазон изменения чисел и их точность.