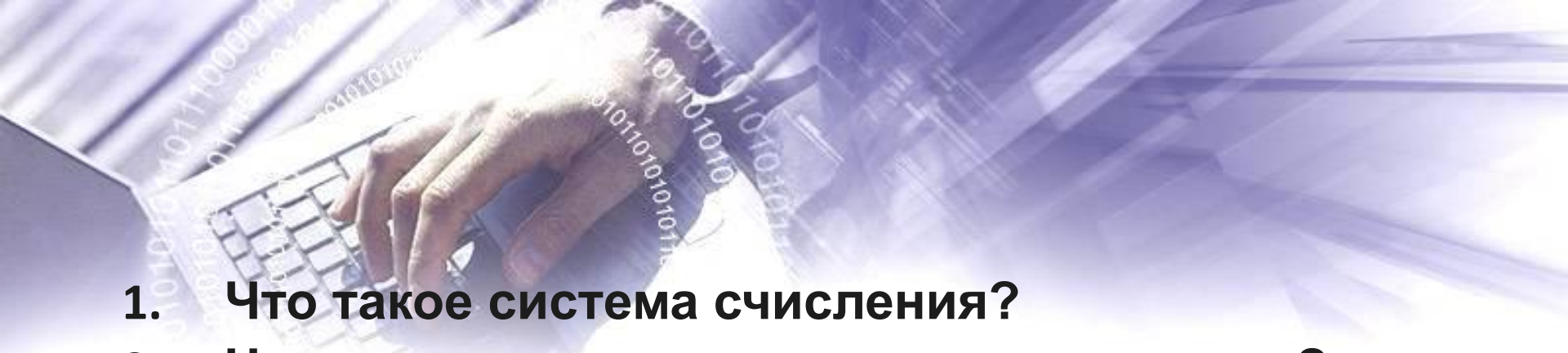


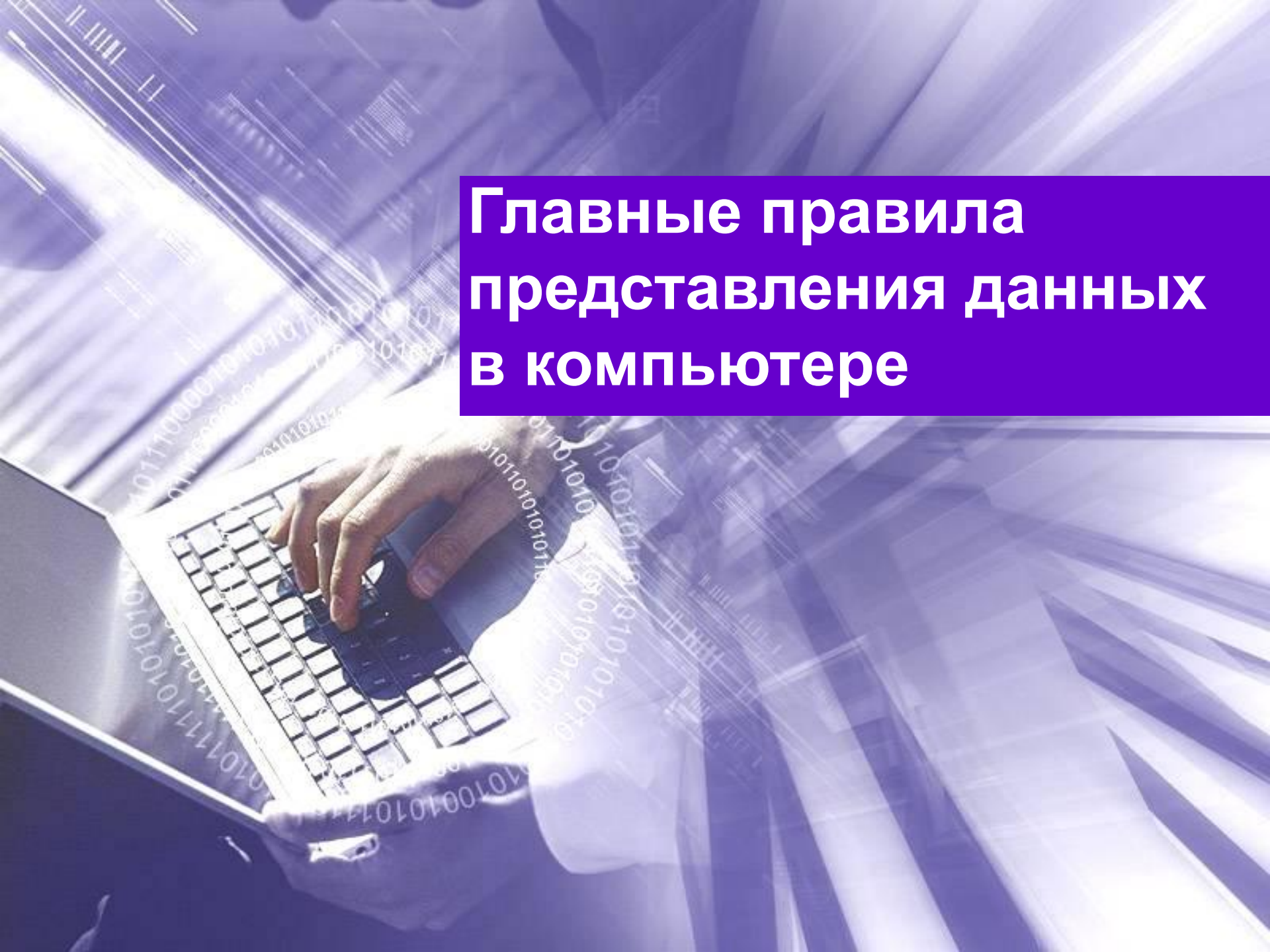
# Представление чисел в памяти компьютера

- 
- 1. Что такое система счисления?**
  - 2. Что такое основание системы счисления?**
  - 3. Какие системы счисления используются в ПК?**
  - 4. Какой алфавит и основание имеет двоичная система счисления?**
  - 5. Какой алфавит и основание имеет десятичная система счисления?**
  - 6. Как перевести число из двоичной системы счисления в десятичную?**
  - 7. Как перевести число из десятичной системы счисления в двоичную?**
  - 8. Каковы правила сложения двоичных чисел.**



# 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

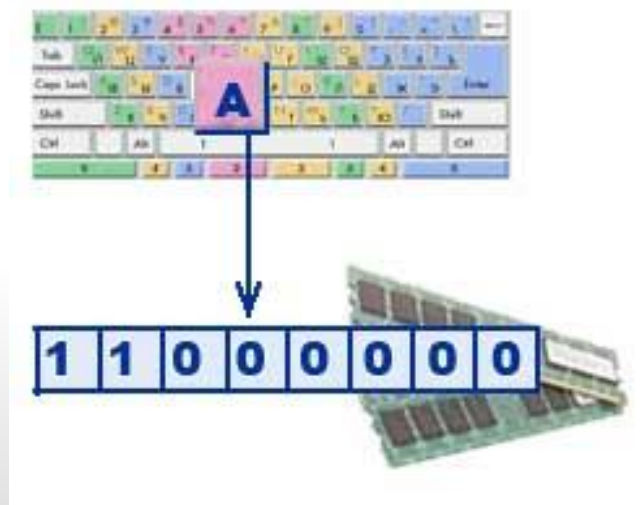


# Главные правила представления данных в компьютере



# Правило 1

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





## Правило 2


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

Дискретное множество состоит из отделенных друг от друга элементов.



## 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**



## Правило 4

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







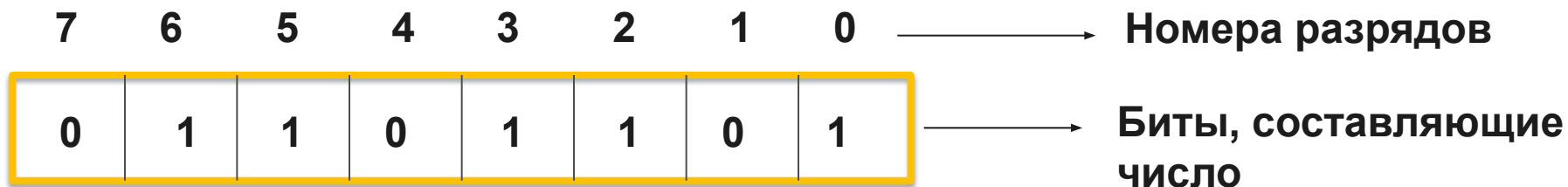
# Числовые величины

**Целые**  
(формат с  
фиксированной  
запятой)

**Вещественные**  
(формат с  
плавающей запятой)

# 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  **$2^n - 1$** .



# 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
---	---	---	---	---	---	---	---

# 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**.



## 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.**

# 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.



# 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

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



## 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  $n$ .



# Вещественные числа

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

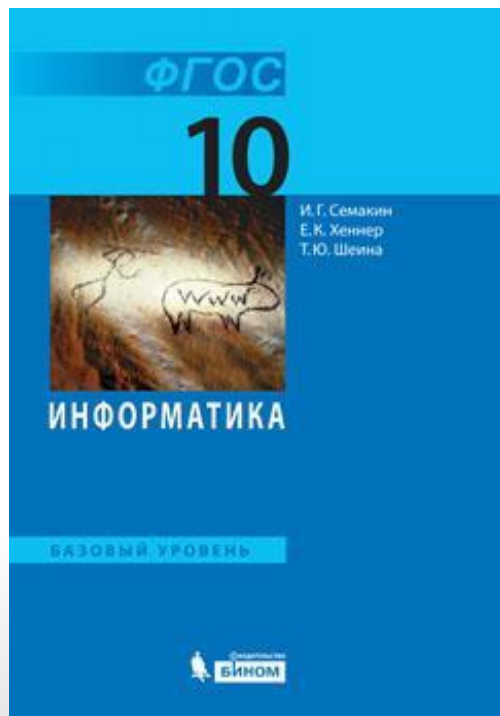
Порядок указывает, на какое количество позиций и в каком направлении должна сместиться десятичная запятая в мантиссе.

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.

# Домашнее задание



- § 5
- Задания № 3, 4 стр. 43