

# Представление числовой информации в компьютере

Информатика  
10 класс

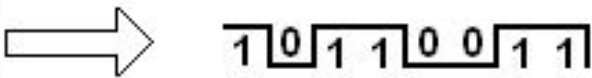
Человек использует десятичную систему счисления, а компьютер – двоичную систему счисления. Поэтому возникает необходимость перевода чисел из десятичной системы в двоичную и наоборот.

- 1. Как перевести целое десятичное число в двоичное?
- 2. Как перевести целое двоичное число в десятичное?  
Двоичная система удобна для компьютера, но неудобна для человека. Для внешнего представления данных и для работы с памятью компьютера используются еще две системы счисления – восьмеричная и шестнадцатеричная.
- 3. Как называются системы счисления с основаниями 2 и 8 или 2 и 16?
- 4. Как переводятся числа в смешанных системах счисления?
- 5. Как мы можем использовать эти замечательные свойства смешанных систем счисления?

## Фронтальный опрос

**Информация** в компьютере представлена в **двоичном коде**, алфавит которого состоит из двух цифр (0 и 1)

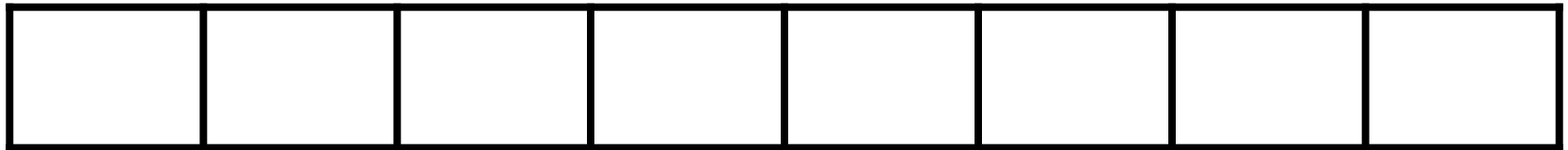
**Двоичное кодирование информации  
в компьютере**

Вид информации	Двоичный код
Числовая	
Текстовая	
Графическая	
Звуковая	
Видео	

**Ячейка** – это часть памяти компьютера, вмещающая в себя информацию, доступную для обработки **отдельной командой** процессора.

**n - 1** разряд

**0** разряд



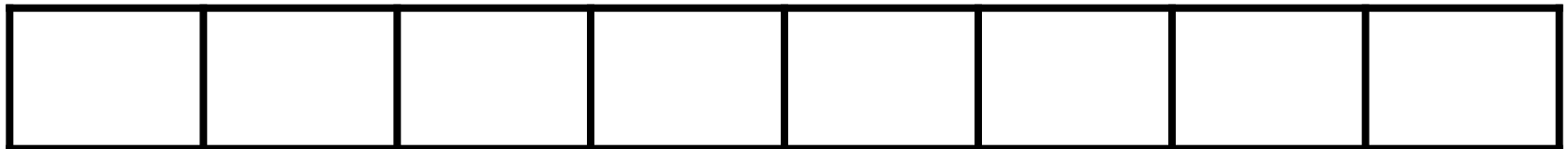
ячейка из **n** разрядов

Содержимое ячейки памяти называется **МАШИНЫМ СЛОВОМ**.

Ячейка памяти разделяется на **разряды**, в каждом из которых хранится разряд числа.

**n - 1** разряд

**0** разряд



ячейка из **n** разрядов

## Единицы измерения объема информации

*Количество информации*, хранящейся в ЭВМ, измеряется ее «объемом», который выражается в **битах** (от английского **binary digit** — двоичная цифра).

**Битом** также называют **разряд** ячейки памяти ЭВМ.

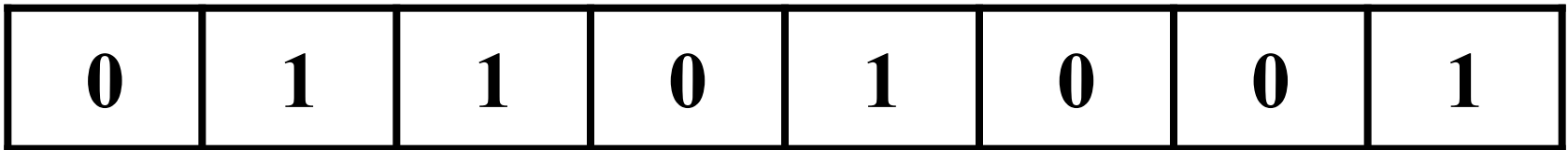
8 бит = 1 байт

*Байт* - основная единица представления данных.

*Байт* (от английского *byte* - слог) – часть машинного слова, состоящая из *8 бит*, обрабатываемая в ЭВМ как одно целое.

7 разряд

0 разряд



ячейка из 8 разрядов

# Форматы данных

7 0



**Байт = 8 бит**

15 8 7 0



**Полуслово = 2 байта = 16 бит**

**Слово = 4 байта = 32 бита**

31 24 23 16 15 8 7 0



**Двойное слово = 8 байт = 64 бита**

63 56 55 8 7 0





## Производные единицы измерения объема информации

**1 Килобайт (*Кбайт*) = 1024 байт =  $2^{10}$  байт;**

**1 Мегабайт (*Мбайт*) = 1024 Кбайт =  $2^{20}$  байт;**

**1 Гигабайт (*Гбайт*) = 1024 Мбайт =  $2^{30}$  байт;**

**1 Терабайт (*Тбайт*) = 1024 Гбайт =  $2^{40}$  байт;**

**1 Петабайт (*Пбайт*) = 1024 Тбайт =  $2^{50}$  байт.**

$$2^{10} = 1024$$

## Специальные типы для **целых чисел** вводятся для:

- *эффективного расходования памяти;*
- *повышения быстродействия;*
- *введения операции деления нацело с остатком;*
- *решения задач экономического характера;*
- *обозначения даты и времени;*
- *нумерации различных объектов.*

# Форматы представления чисел

целочисленный

целые положительные  
числа

целые числа со  
знаком

вещественные

с фиксированной  
точкой

с плавающей точкой

# Целочисленный формат

(Формат с фиксированной запятой) используется для представления в компьютере целых положительных и отрицательных чисел.

Для этого, как правило, используются форматы, кратные байту: 1, 2, 4 байта.

Однобайтовое представление применяется только для положительных целых чисел. Наибольшее положительное целое число, которое может быть записано в 1 байтовом формате 255 или 11111111.

Для положительных и отрицательных целых чисел обычно используется 2 и 4 байта, при этом старший бит выделяется под знак числа: 0 – плюс, 1 – минус.

Самое большое число, которое может поместиться в 2 – байтовом формате это 0 11111111 11111111 или 32 767

## Формат с плавающей точкой

Используется в компьютере для представления действительных чисел.

Числа с плавающей точкой размещаются, как правило, в 4 или 8 байтах.

# Алгоритм представления целого положительного числа в компьютере в двух байтовом формате

1. Число переводится в двоичную систему счисления;
2. Результат дополняется нулями слева в пределах выбранного формата;
3. Последний разряд является знаковым, в положительном числе он равен 0, а в отрицательном 1.

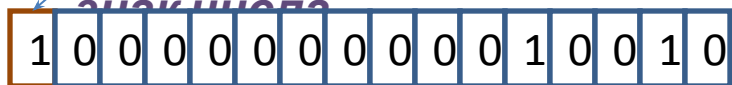
Пример:

*представить десятичное число – 18 в двухбайтовом формате.*

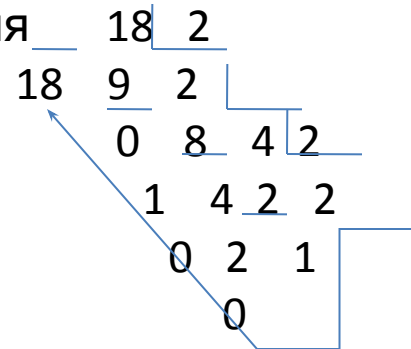
1. Переведем – 18 в двоичную систему счисления

$$18 = 10010$$

2. Запишем в старший разряд знак числа



2 – й байт (8 бит)    1 – й байт (8 бит)



Максимальное положительное число (с учетом выделения одного разряда на знак) для целых чисел со знаком в  $n$ -разрядном представлении равно:

$$2^{n-1} - 1.$$

Для представления отрицательных чисел используется *дополнительный код*.

Дополнительный код позволяет заменить арифметическую операцию вычитания операцией сложения, что существенно упрощает работу процессора и увеличивает его быстродействие.

Дополнительный код отрицательного числа  $A$ , хранящегося в  $n$  ячейках, равен

$$2^n - |A|.$$

## **Дополнительный код**

# Алгоритм

Для получения дополнительного кода отрицательного числа можно использовать довольно простой :

1. Модуль числа записать в прямом коде в  $n$  двоичных разрядах.
2. Получить обратный код числа, для этого значения всех битов инвертировать (все единицы заменить на нули и все нули заменить на единицы).
3. К полученному коду прибавить единицу.

Запишем дополнительный код отрицательного числа  $-2002$  для 16-разрядного компьютерного представления:

Прямой код модуля	$ -2002_{10} $	$0000011111010010_2$
Обратный код	Инвертирование	$1111100000101101_2$
	Прибавление единицы	$1111100000101101_2$ $+ 0000000000000001_2$
Дополнительный код		$1111100000101110_2$

# Представление вещественного числа в компьютере

С фиксированной точкой  $10^2 25.43$

С плавающей точкой –  $0.2543 * 10^2$  ;  $2.543 * 10^1$  ;  $254.3 * 10^{-1}$

С плавающей запятой удобно представлять числа очень близкие к нулю.

Любое число с плавающей запятой можно представить в виде  $N = m * g^p$ , где  $m$  – мантисса, представляющая собой правильную дробь ( $0,1 < m < 1$ )

$g$  -основание системы счисления;

$p$  – порядок числа, указывающий порядок запятой в числе.

Например :  $25.324 = 0.25324 * 10^2$

$m = 0.25324$

$g = 10$

$P = 2$



В двух байтовом формате представление вещественного числа первые байт и три разряда второго байта выделяются для размещения мантиссы, в остальных разрядах второго байта размещаются порядок числа, знаки числа и порядка.

1-й байт							0-й байт							
Знак числа	Знак порядка	Порядок			Мантисса									

В 4-байтовом формате представления вещественного числа первые три байта выделяются для размещения мантиссы, в четвертом байте размещаются порядок числа, знаки числа и порядка.

3-й байт				2-й байт				1-й байт				0-й байт				
Знак числа	Знак порядка	Порядок			Мантисса											

# Пример

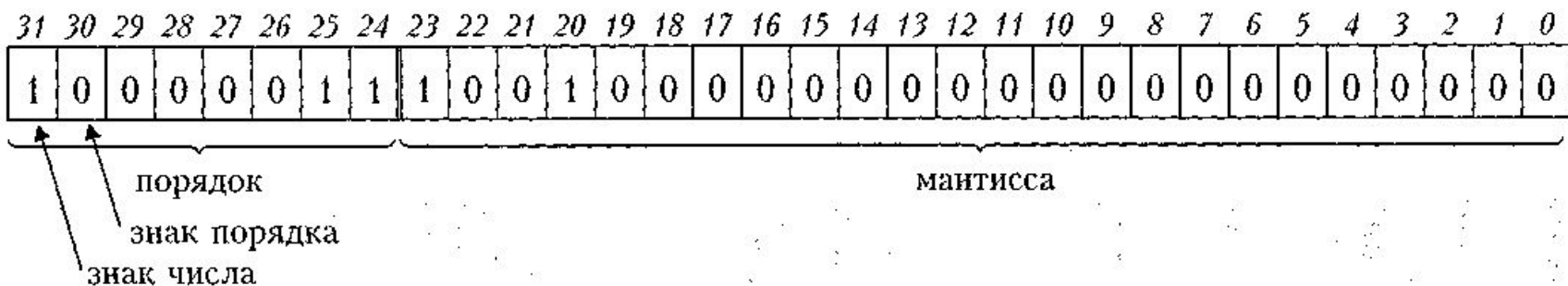
$$\begin{aligned}n &= -100,1_2 = -0,1001 \cdot 2^{11} \\m &= -0.1001 \\q &= 2 \\p &= 11 (3_{10} = 11_2).\end{aligned}$$

Данное число может быть представлено в четырехбайтовом формате (32 бита) следующим образом (см. рис.).

На мантиссу отводится 23 бита, поэтому максимальная величина мантиссы равна  $2^{23} - 1 = 8\,388\,607$ , т.е. 7 десятичных цифр.

Компьютер при вычислениях отбрасывает лишние цифры в мантиссе, поэтому все вычисления с вещественными числами всегда выполняются приближенно (с ошибкой). Вещественные числа обрабатываются в компьютере медленнее, чем целые.

Представление числа в четырехбайтовом формате



## Выполни задание

1. Представьте десятичное число 29 в двухбайтовом формате.
2. Представьте десятичное число 40 в двухбайтовом формате.
3. Представьте двоичное число  $11011,11$  в четырехбайтовом формате. Представьте число сначала в форме с плавающей запятой.
4. Представьте двоичное число  $1110,101$  в четырехбайтовом формате. Представьте число сначала в форме с плавающей запятой.