

Представление чисел в памяти компьютера

10 класс. Профильный уровень

-
- Сегодня вы познакомитесь с алгоритмом представления целых чисел в компьютере;
 - *Развивающая:* способствовать развитию у учащихся интереса к теоретическим основам фундаментальных наук – теории информатики и математических основ информационных технологий.
 - *Воспитательная:* обеспечить условия для формированию у учащихся таких качеств как точность и внимательность.



Перевести числа:

1. $156_{10} = ?_2$

2. $1110_2 = ?_{10}$

На выполнение задания 2
МИН.

Проверка

1. $156_{10} = 10011100_2$

2. $1110_2 = 14_{10}$

Выполнить действия в двоичной системе счисления

□ $10011100 - 110000 =$

□ $1101100 + 11100 =$

□ $111101 - 1001011 =$

На выполнение задания 2 мин.



Проверка

□ 1101100

□ +11100

□ 10001000

□ 10011100

□ -110000

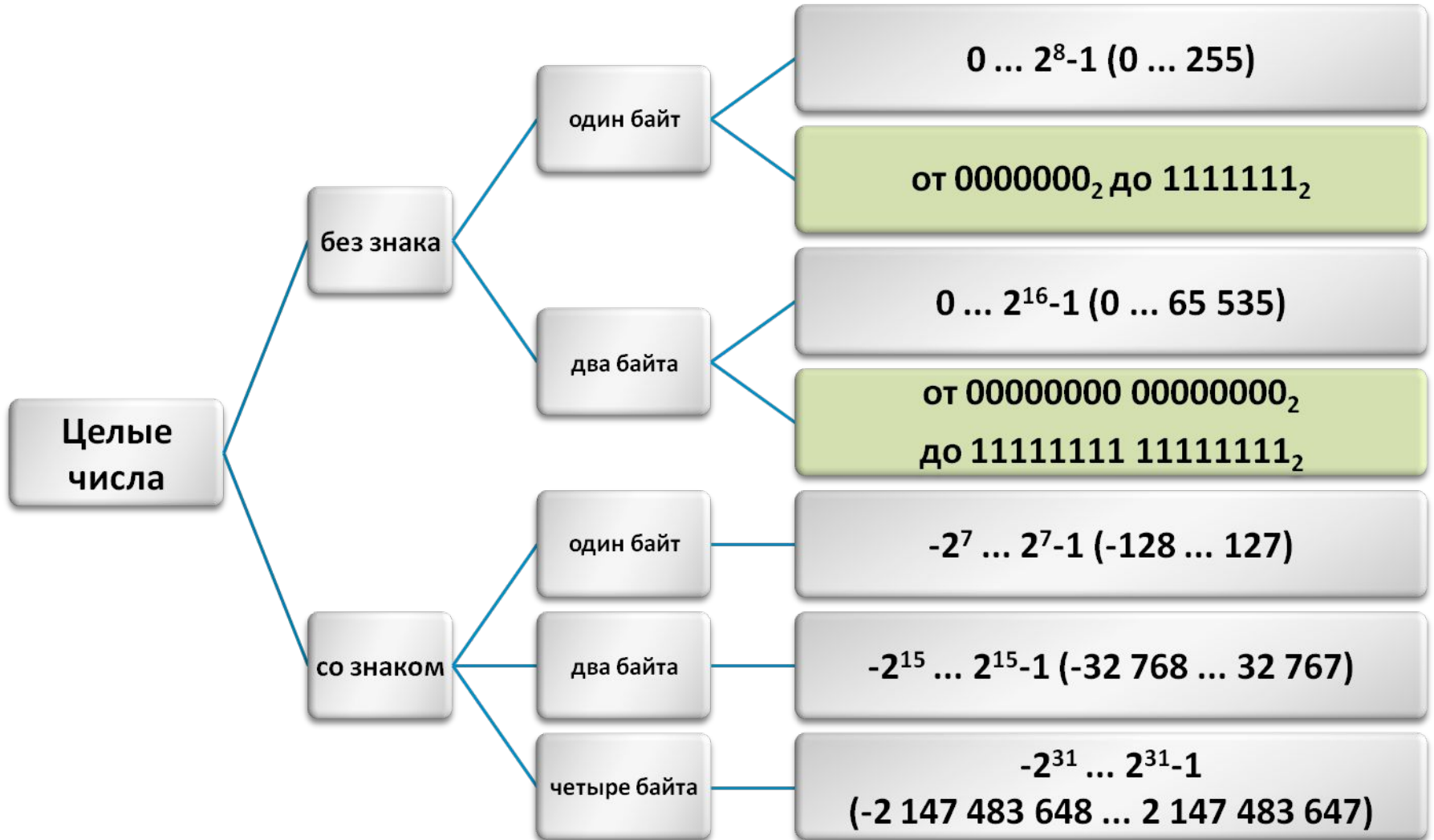
□ 1101100



▣ **Учебная задача:**

- научиться записывать целые отрицательные числа в машинных кодах,
- использовать полученные знания при сложении чисел с разными знаками.

Целые числа в памяти компьютера



Представление чисел в формате с фиксированной запятой

- Для хранения целых неотрицательных чисел отводится одна ячейка памяти (8 бит).
- Минимальное число 00000000
- Число в n-разрядном представлении
 $2^n - 1$
- Максимальное число $2^8 - 1 = 255_{10}$ или 11111111



Числа без знака

□ Число $39_{10} = 100111_2$ в **однобайтовом** формате:

Номера разрядов	7	6	5	4	3	2	1	0
Биты числа	0	0	1	0	0	1	1	1

□ Число $39_{10} = 100111_2$ в **двубайтовом** формате:

Номера разрядов	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Биты числа	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1

□ Число $65\,535_{10} = 11111111\,11111111_2$ в **двубайтовом** формате:

Номера разрядов	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Биты числа	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1



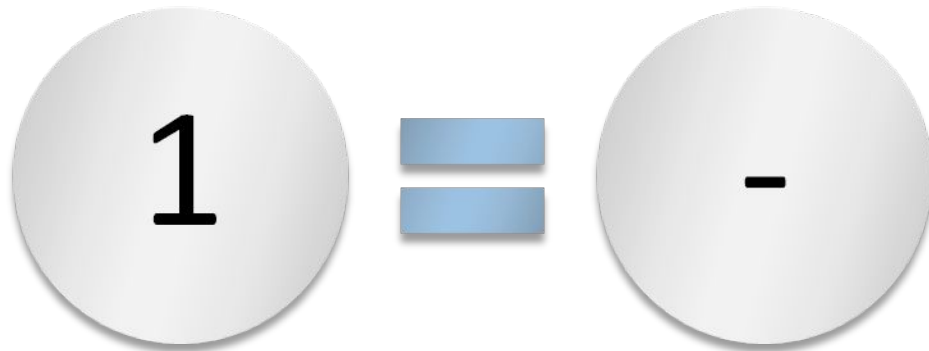
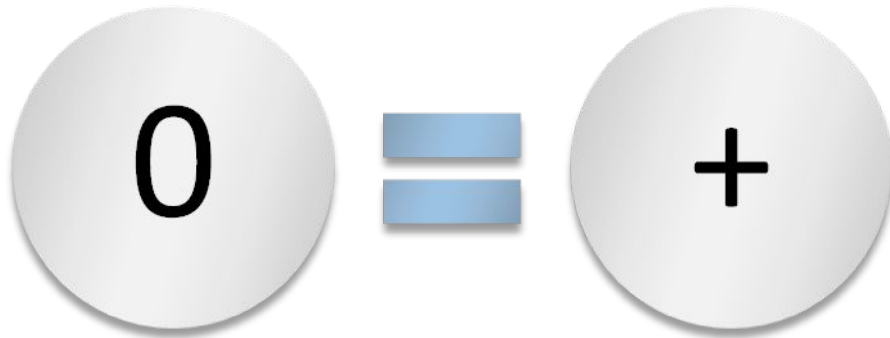
Хранение целых чисел со знаком

- ▣ Для хранения отводится 1 или 2 ячейки памяти (8 или 16 бит)
- ▣ Старший (левый) разряд отводится под знак.



Числа со знаком

Если самый левый (старший) разряд содержит информацию о знаке числа



Формы записи целых чисел со знаком

Прямой код

Обратный код

Дополнительный код



Прямой код числа

**Это -представление в компьютере
положительных чисел с
использованием формата «знак –
величина».**

Пример: $2002_{10} = 11111010010_2$

В 16-ти разрядном представлении

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



**При представлении целых чисел в
n-разрядном представлении со
знаком максимальное
положительное число**

$$A = 2^{n-1} - 1$$



Формы записи чисел целых чисел со знаком

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

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

- 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

Число -127:

Код модуля числа: 0 1111111

Обратный код числа: 1 0000000

- 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

Дополнительный код

- ▣ Используется для представления отрицательных чисел
- ▣ Позволяет заменить арифметическую операцию вычитания операцией сложения, что упрощает работу процессора и увеличивает его быстродействие.
- ▣ Дополнительный код отрицательного числа A , хранящегося в n – ячейках, равен $2^n - |A|$



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.



Арифметические действия

□ 1) *A и B положительные:*

Десятичная

запись:

+

5

7

12

$A_{\text{пк}}$

$B_{\text{пк}}$

$C_{\text{пк}}$

Двоичные коды:

0 0 0 0 0 1 0 1

0 0 0 0 0 1 1 1

0 0 0 0 1 1 0 0



Арифметические действия

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

Десятичная

запись:

+

5

-12

-7

$A_{\text{пк}}$

$B_{\text{дк}}$

$C_{\text{дк}}$

Двоичные коды:

0 0 0 0 0 1 0 1

1 1 1 1 0 1 0 0

1 1 1 1 1 0 0 1

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

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



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

0 0 0 0 0 1 1 1



Перенос
отбрасывается

Единицу переноса из
отбрасывает.

ьютер



Арифметические действия

4) *A и B отрицательные*

Десятичная

запись:

-5
+
-7
<hr/>
-12


$A_{\text{дк}}$

$B_{\text{дк}}$

$C_{\text{дк}}$

Двоичные коды:

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

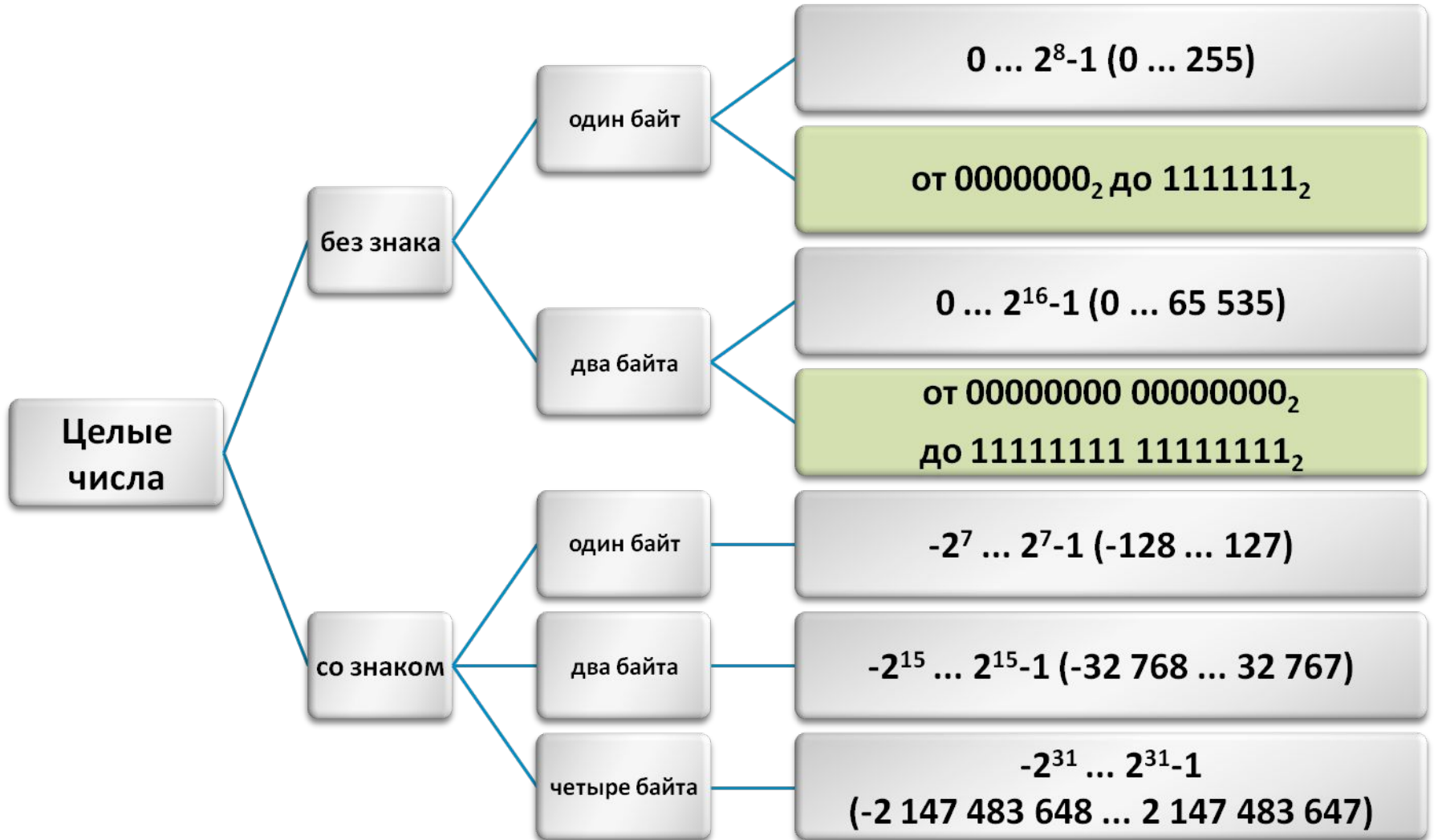


**Перенос
отбрасывается**

При переводе в прямой код старшие цифровые части результата инвертируются и к младшему разряду прибавляется единица:

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

Целые числа в памяти компьютера



Вернемся к заданию №2.

□ Решите данный пример

□ $111101 - 1001011$



Проверка

□ 111101

□ -1001011

	61	$A_{\text{пк}}$	0	0	1	1	1	1	0	1
+	-75	$B_{\text{дк}}$	1	0	1	1	0	1	0	1
	-14	$C_{\text{дк}}$	1	1	1	1	0	0	1	0

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

$$10001101 + 1 = 10001110 = -14_{10}$$



Упражнение 1

- ▣ **Определить максимальное положительное число, которое может храниться в оперативной памяти в формате *целое число со знаком в двухбайтном представлении*.**



Решение

$$A = 2^{n-1} - 1$$

$$A_{10} = 2^{15} - 1 = 32767_{10}$$



Упражнение 2.

- ▣ **Получить 8-разрядный дополнительный код числа -52:**



Решение.

- ▣ **00110100 - число $|-52|=52$ в прямом коде**
- ▣ **11001011 - число -52 в обратном коде**
- ▣ **11001100 - число -52 в дополнительном коде**



Практическая работа

- Задание.
- Получить прямой, обратный и дополнительный коды для числа -536 в формате «Знак» – «величина» в 16-разрядном представлении
- Провести проверку решения на калькуляторе.



Практическая работа

- Решение.
- модуль числа -536 будет равен
0000001000011000,
- обратный код – 111110111100111,
- дополнительный код –
111110111101000.



Практическая работа

- Проверка на калькуляторе.
- Ввести значение модуля числа -536, т. е. число 536 в строку ввода
- с помощью опционной кнопки **Bin** преобразуем это число, представленное в десятичной системе счисления, в двоичную систему, предварительно установив опционную кнопку **2 байта**.
- Нажав кнопку **Not** калькулятора, получим обратный код числа.
- прибавив к обратному коду двоичную единицу, – дополнительный код.
- Получился окончательный результат



Калькулятор

Правка Вид Справка

1111110111101000

☐ Hex ☐ Dec ☐ Oct ☒ Bin

☐ 8 байт ☐ 4 байта ☒ 2 байта ☐ 1 байт

☐ Inv

☐ Нур

Backspace

CE

C

Sta

F-E

(

)

MC

7

8

9

/

Mod

And

Ave

dms

Exp

ln

MR

4

5

6

*

Or

Xor

Sum

sin

x^y

log

MS

1

2

3

-

Lsh

Not

s

cos

x^3

n!

M+

0

+/-

,

+

=

Int

Dat

tg

x^2

1/x

pi

A

B

C

D

E

F

Практическая работа

- Проверка на калькуляторе.
- Можно поступить еще проще: набрав на калькуляторе число -536 и активизировав кнопку **Bin**, получить дополнительный код этого числа в двоичной системе счисления.

