**04.12.23.**

**Системы счисления (с/с). Перевод чисел из одной системы счисления в другую систему счисления. Представление информации в оперативной памяти компьютера.**

**Системы счисления (с/с)**

**Система счисления** – это способ представления любого числа посредством заданного набора символов, *называемых цифрами*. Различают ***позиционные*** и***непозиционные*** **с/с**. В позиционных**с/с** значение каждой цифры зависит от её позиции в числе. Пример непозиционной **с/с** – римская **с/с**.

**Основанием системы счисления** называется количество цифр и символов, используемых в ней (обозначим основание **с/с** через **р**).

**Двоичная с/с**.Для представления чисел используют только цифры **0** и **1**. Основание **с/с** записывается как

**210  = 1\*21 + 0\*20 = 10**2

Арифметические операции в **2** **c/c**:

0 + 1 = 1 1 – 0 = 1 0 \* 1 = 0

1 + 0 = 1 1 – 1 = 0 1 \* 0 = 0

1 + 1 =10 10 – 1 = 1 1 \* 1 = 1

Примеры выполнения операций в **2-**ой **с/с:**

00110111 11110

+01101101 -01101

10100100 10001

**Шестнадцатеричная с/с.** Для представления чисел используют шестнадцать цифр: **0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F**. Основание системы счисления записывается как

**1610  = 1\*161 +0\*160 = 1016**

Примеры выполнения операций в **16** с/с:

F 0C3 D8 1000 1A35, F1

+2 +175 **-**19 -21C -1EB, ED

11 238 BF DE4 184A, 04

**Перевод чисел из одной системы счисления в другую**

**Правило 1. Эквивалентный перевод.** Перевод смешанного числа (числа с дробной частью) из системы счисления **p** в систему счисления **q,** если основания систем счисления **р** и **q** связаны соотношением **p = qk** , где **к –** целое положительное число, осуществляется поразрядно – каждая цифра исходной системы **р з**аменяется на её **к**-*разрядный эквивалент* в новой системе **q**:

BC5,AD416 =(1011)(1100)(0101),(1010)(1101)(0100) = 101111000101,1010110101002

***Обратный перевод***. Для перевода из системы счисления **q** в систему счисления **р** запись числа разбивают на группы по **к** цифр, двигаясь от запятой вправо и влево. Если при этом крайние группы окажутся неполными, то их дополняют до **к** цифр незначащими нулями. Затем заменяют каждую группу цифр её эквивалентом в системе счисления **р**. Например:



1010100111,1012 = (0010)(1010)(0111),(1010) = 2A7,A16

**Правило 2. Перевод целой части числа**. Чтобы перевести целую часть числа из системы счисления **р** в систему счисления **q**, нужно выполнить в исходной системы **р** последовательное деление числа и получаемых частных на основание системы **q** до тех пор, пока очередное частное не станет меньше **q.** Число в системе **q** следует записать в виде последнего частного и полученных остатков в обратном направлении, представляя их в системе счисления **q**.

Рассмотрим примеры:

1. перевести число **26** из **10** с/с в **2** с/с

2610 = 110102

1. перевести число **286** из **10** с/с в **16** с/с



286 : 16 –> 17 – частное, 14 – остаток (мл. цифра)

17 : 16 –> 1 – частное 1 (ст. цифра), 1 – остаток

28610 = 11E16

1. перевести число из **2** с/с в **10** с/с. Удобно использовать **16** с/с.

110101102  = D616 = 13 \* 161 + 6 \* 160 = 21410

1. перевести число из **16** с/с в **10** с/с

1CB16 = 1 \* 162 + 12 \* 161 + 11 \* 160 = 45910

**Правило 3. Перевод дробной части числа**. Чтобы перевести правильную дробь из системы счисления **р** в систему счисления **q**, нужно выполнить последовательное умножение в системе **р** исходной дроби и дробных частей получаемых произведений на основание системе **q**. Целые части полученных произведений в прямом направлении, представленные в системе счисления **q**, дадут искомую дробную часть числа. Процесс умножения продолжают до тех пор, пока дробная часть произведения не станет нулевой или пока не получат требуемое количество знаков записи дроби в **q**-ой системе счисления.

**Примеры:**

1. перевести дробное число **0,3** из **10** с/с в **2** с/с

0,310 = 0,0(1001)2

1. перевести дробное число **0,1** из **10** с/с в **16**с/с

0,1 \* 16 –> 1 – целая часть; 0,6 – дробная часть

0,6 \* 16 –> 9 – целая часть; 0,6 – дробная часть

…

0,110 = 0,1(9)16

**Представление информации в оперативной памяти компьютера**

Информация в памяти компьютера хранится и обрабатывается в двоичном виде. Форма записи данных в памяти называется *внутренним представлением информации.* Единицей хранения информации является один **бит**, т. е. ***двоичный разряд***, который может принимать значения **0** или **1**. Восемь битов – это **один** **байт**.

***Адресом любой информации***считается **адрес** (номер) самого первого байта области памяти (*ячейки*), выделенной для её хранения.

**Представление чисел с фиксированной точкой**

Положение точки зафиксировано после младшей цифры числа, дробная часть отсутствует, изображение точки в числе опускается, т. е. с фиксированной точкой могут храниться только целые числа, представленные в **2**-ой **с/с**.

**Представление положительных целых чисел*.*** Число может занимать в памяти компьютера **8, 16** или **32** двоичных разрядов – т**.е. 1, 2, 4** байта соответственно. Для **64**-разрядных [операционных систем](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) целое число может занимать и **64** бита (**8 байтов**).

Рассмотрим, как хранится целое положительное число в **4**-х байтах памяти:

* знак числа записывается в старшем бите **1**-го байта;
* младший двоичный разряд числа записывается в **0** бите, т.е. число заполняет байты справа налево.

Например, чтобы **137** представить в **4-х** байтах, необходимо выполнить:

* перевести число в **2**-ю систему счисления

13710 = 8916 = 100010012

* записать двоичное число без знака в **4** байта:

00000000 00000000 00000000 10001001

1-й байт 2-й байт 3-й байт 4-й байт

**Представление отрицательных целых чисел.**Дополнительный код числа получается из прямого кода путем инверсии каждого бита (**0** на **1**, **1** на **0**), а затем прибавления **1** к младшему биту числа. Например, для представления числа **-95** в 2-х байтах необходимо выполнить:

* перевести число в 2-ую систему счисления

9510 = 5F16 = 10111112

* записать двоичное число без знака в два байта:

00000000 01011111

* инвертировать: 11111111 10100000
* прибавить **1** к младшему биту:

11111111 10100001 – дополнительный код числа -95 в **2-х байтах**

В результате выполнения такого [***алгоритма***](http://edufuture.biz/index.php?title=%D0%A7%D1%82%D0%BE_%D1%82%D0%B0%D0%BA%D0%BE%D0%B5_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC) единица в старшем разряде получается автоматически. Она и является **признаком** отрицательного значения.

Форму записи положительных чисел называют ***прямым кодом***числа.

Отрицательные числа хранятся в памяти в ***дополнительном коде.***

Если при хранении целых положительных чисел использовать бит, отведённый под знак, то можно почти вдвое увеличить числовой диапазон. Так, например, число

6553510 FFFF1611111111 111111112

объявленное как целое без знака, в двух байтах запишется в виде

**11111111 11111111**

1-й байт 2-й байт