Системы счисления и представление данных в компьютере.

Система счисления (С.С.) – знаковая система, в которой числа записываются по определенным правилам с помощью символов некоторого алфавита, называемых цифрами.

Системы счисления

делятся на две группы:

- Непозиционные
- Позиционные

Непозиционная – с.с., в которой значение цифры не зависит от ее позиции в записи числа.

Позиционная – характеризуется тем, что количественное значение цифры зависит от ее позиции в числе.

Каждая позиционная с.с. имеет определенный алфавит цифр и основание (**р**), равное количеству цифр (знаков в алфавите).

Системы счисления

Цифры, используемые в С.С. с различными основаниями:

- p=10 (0,1, 2, 3, 4, 5, 6, 7, 8, 9)
- p=8 (0, 1, 2, 3, 4, 5, 6, 7)
- p=2 (0, 1)
- p=16 (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F)

Системы счисления

Для представления чисел используется схема Горнера:

$$A_p = A_p^{ц} + A_p^{др}$$

где $$A_p^{ц} = \sum_{k=0}^{n-1} a_k \cdot p^k$$ целая часть числа

$$A_p^{др} = \sum_{k=-m}^{-1} a_k \cdot p^k$$ дробная часть числа

n – число целых разрядов (нумерация справа с 0)
m – число дробных разрядов
k – порядковый номер разряда в числе
a_k - цифры

Системы счисления

$$A_p = a_n \cdot p^n + \ldots + a_0 \cdot p^n + a_{-1} p^{-1} + \ldots + a_{-m} \cdot p^{-m}$$

- *n* – число целых разрядов (нумерация справа с 0)
- *m* – число дробных разрядов
- $a_n \ldots a_{-m}$ - цифры

Системы счисления

Перевод в десятичную С.С.

$29_{10} \quad = 2*10^1+9*10^0$

$35_8 \quad = 3*8^1+5*8^0=29_{10}$

$1D_{16} \quad = 1*16^{1+}13*16^{0=}29_{10}$

$11101_2=1*2^4+1*2^3+1*2^2+0*2^1+1*2^0=29_{10}$

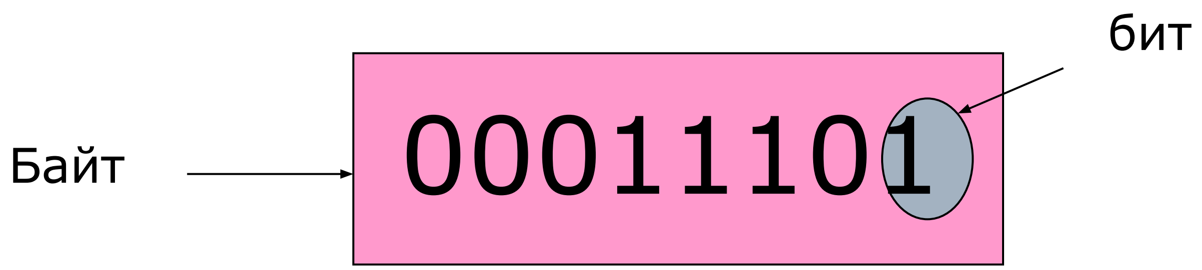
Перевести в десятичную систему счисления:

- 1100_2
- 1010111_2
- 147_8
- 243_8
- $A15_{16}$
- $1EF_{16}$

Перевести в десятичную систему счисления:

- $1100_2 = 12_{10}$
- $1010111_2 = 87_{10}$
- $147_8 = 103_{10}$
- $243_8 = 163_{10}$
- $A5_{16} = 165_{10}$
- $1EF_{16} = 495_{10}$

Единицы измерения количества информации



1 бит = 1 двоичный разряд

1 байт = 8 бит (2^8 бит)

1 Килобайт = 1024 байт (2^{10} байт)

1 Мегабайт= 1024 Килобайт (2^{10} Кбайт)

1 Гигабайт = 1024 Мегабайт (2^{10} Мбайт)

1 Терабайт = 1024 Гигабайт (2^{10} Гбайт)

Системы счисления.
Перевод чисел из десятичной системы счисления в систему счисления с основанием *P*

Алгоритм перевода чисел из десятичной системы счисления в систему счисления с основанием *P* позволяет оперировать с числами в той системе счисления, из которой число переводится, и может быть сформулирован следующим образом.

Системы счисления.
Перевод чисел из десятичной системы счисления в систему счисления с основанием *P*

При переводе смешанного числа следует переводить его целую и дробную части отдельно.

1. Для перевода целой части числа его, а затем целые части получающихся частных от деления следует последовательно делить на основание ***P*** до тех пор, пока очередная целая часть частного не окажется равной 0. Остатки от деления, записанные последовательно справа налево, образуют целую часть числа в системе счисления с основанием ***P***.

Системы счисления.
Перевод чисел из десятичной системы счисления в систему счисления с основанием *P*

2. Для перевода дробной части числа его, а затем дробные части получающихся произведений следует последовательно умножать на основание ***P*** до тех пор, пока очередная дробная часть произведения не окажется равной 0 или не будет достигнута нужная точность дроби. Целые части произведений, записанные после запятой последовательно слева направо, образуют дробную часть числа в системе счисления с основанием ***P***.

Перевод чисел из десятичной системы счисления в систему счисления с основанием *P*

Рассмотрим перевод смешанного числа из десятичной в двоичную систему счисления на примере числа 46,625.

Переводим целую часть числа:

46 : 2 = 23 (остаток **0**)
23 : 2 = 11 (остаток **1**)
11 : 2 = 5 (остаток **1**)
5 : 2 = 2 (остаток **1**)
2 : 2 = 1 (остаток **0**)
1 : 2 = 0 (остаток **1**).

Записываем остатки последовательно справа налево — **101110**

то есть $46_{10} = \mathbf{101110_2}$

Переводим дробную часть числа:

0,625 • 2 = **1**,250
0,250 • 2 = **0**,500
0,500 • 2 = **1**,000

Записываем целые части получающихся произведений после запятой последовательно слева направо— **0,101**
то есть $0{,}625_{10} = \mathbf{0{,}101_2}$.

Окончательно $\mathbf{46{,}625_{10} = 101110{,}101_2}$.

Перевести в двоичную с/с

- 65_{10}
- 124_{10}
- $0,125_{10}$
- $15,75_{10}$
- $231,146_{10}$ (с точностью 6 знака после запятой)

Перевести в двоичную с/с (ответы)

- $65_{10} = 1000001_2$
- $124_{10} = 1111100_2$

- $0{,}125_{10} = 0{,}05_2$

- $15{,}75_{10} = 1111{,}11_2$
- $231{,}146_{10} = 11100111{,}001001_2$

Системы счисления

Связь двоичной С.С. с восьмеричной и шестнадцатеричной

$$\underbrace{011}_{3}\ \ \underbrace{101}_{5}$$

$$\underbrace{0001}_{1}\ \ \underbrace{1101}_{D}$$

Примеры:

- Перевести двоичное число 101000110 в восьмеричную с/с

  101 000 110

- Перевести двоичное число 101000110 в шестнадцатеричную с/с

0001 0100 0110

Примеры:

- Перевести восьмеричное число 315 в двоичную с/с

  $\underbrace{3}\quad\underbrace{1}\quad\underbrace{5}$

- Перевести шестнадцатеричное число 12D в двоичную с/с

  $\underbrace{1}\quad\underbrace{2}\quad\underbrace{D}$

Задания по теме С/С:

- Сколько единиц в двоичной записи десятичного числа 172,25?
- Переведите восьмеричное число 37 в четверичную систему счисления.
- Вычислить $B15_{16} – 151_{8}$. Результат представить в шестнадцатеричной системе счисления
- Чему равна разность чисел 100_{16} и 1010101_{2}? Результат приведите в десятичной системе счисления.

Представление чисел с фиксированной и плавающей запятой

В вычислительных машинах применяются две формы представления двоичных чисел:

- естественная форма или форма с фиксированной запятой (точкой);
- нормальная форма или форма с плавающей запятой (точкой).

Представление чисел с фиксированной запятой

В форме представления с фиксированной запятой все числа изображаются в виде последовательности цифр с постоянным для всех чисел положением запятой, отделяющей целую часть от дробной.

Например: в десятичной системе счисления имеется 5 разрядов в целой части числа (до запятой) и 5 разрядов в дробной части числа (после запятой); числа, записанные в такую разрядную сетку, имеют вид:

+00721,35500

+00000,00328

-10301,20260

Представление чисел с фиксированной запятой

Эта форма наиболее проста, естественна, но имеет небольшой диапазон представления чисел и поэтому чаще всего не приемлема при вычислениях.

В современных компьютерах естественная форма представления используется как вспомогательная и только для целых чисел.

Представление чисел с плавающей запятой

В форме представления с плавающей запятой каждое число изображается в виде двух групп цифр. Первая группа цифр называется **мантиссой**, вторая — **порядком**, причем абсолютная величина мантиссы должна быть меньше 1, а порядок — целым числом.

Представление чисел с плавающей запятой

В общем виде число в форме с плавающей запятой может быть представлено так:

$$N = \pm M \bullet P^{\pm r},$$

где

M — мантисса числа ($|M| < 1$);
r — порядок числа (целое число);
P — основание системы счисления.

Представление чисел с плавающей запятой

Например, приведенные ранее числа в нормальной форме запишутся так:

$+0,721355 \bullet 10^3$

$+0,328 \bullet 10^{-2}$

$-0,103012026 \bullet 10^5$

Нормальная форма представления имеет огромный диапазон отображения чисел и является основной в современных компьютерах.

Все числа с плавающей запятой хранятся в машине в так называемом нормализованном виде. Нормализованным называют такое число, в старшем разряде мантиссы которого стоит единица.

Алгебраическое представление двоичных чисел

Знак числа обычно кодируется двоичной цифрой, при этом код 0 означает знак + (плюс), код 1 — знак - (минус). Для алгебраического представления чисел, то есть для представления чисел с учетом их знака, в машинах используются специальные коды:

- **прямой** код числа;
- **обратный** код числа;
- **дополнительный** код числа.

Алгебраическое представление двоичных чисел

При этом *обратный* и *дополнительный* коды позволяют заменить неудобную для компьютера операцию вычитания на операцию сложения с отрицательным числом.

Чаще применяется *дополнительный* код, т.к. обеспечивает более быстрое выполнение операций.

Алгебраическое представление двоичных чисел

Правила образования машинных кодов:

1. *прямой код положительного и отрицательного чисел отличается только знаковыми разрядами, модуль числа не изменяется;*
2. *положительное число в прямом, обратном и дополнительном кодах имеет одинаковое изображение;*
3. *обратный код отрицательного двоичного числа об разуется из прямого кода положительного числа путем замены всех единиц на нули, а нулей на единицы, включая знаковый разряд;*
4. *дополнительный код отрицательного числа образуется путем добавления единицы к младшему разряду обратного кода этого же числа или заменой в коде положительного числа всех нулей на единицы, а единиц на нули, исключая последнюю единицу и следующие за ней нули.*

Алгебраическое представление двоичных чисел

Числа, представленные в *естественной* форме, в памяти ЭВМ представляются в *дополнительном* коде, числа в *нормальной* форме хранятся в *прямом* коде. Действия в ЭВМ выполняются в прямом и дополнительном кодах, обратный код используется для получения дополнительного кода.

Действия над числами, представленными в естественной форме

Даны два числа:

A = 254, B = 175.

Найти сумму чисел при разных знаках слагаемых в 16-ти разрядном формате.

Решение

а) Представим исходные числа в двоичной системе счисления:

$A_{16} = FE \sim A_2 =$ 11111110;

B_{16} = AF $\sim B_2$=- 10101111.

б) Составим машинные коды этих чисел с разными знаками:

$[A]_{пк}$ = 0.000000011111110

$[B]_{пк}$ = 0.000000010101111

[-A]дк =1.111111100000010

[-B]дк =1.111111101010001

Действия над числами, представленными в естественной форме

в) Выполним действия:

C1 = A + B

$[A]_{пк} = 0.000000011111110$
$[B]_{пк} = 0.000000010101111$
$[C]_{пк} = 0.000000110101101 > 0;$

C2 = *A-B*

$[A]_{пк} = 0.000000011111110$
$[-B]_{дк} = 1.111111101010001$
$[C2]_{пк} = 10.000000001001111 > 0;$

C3 = B-A

$[B]_{пк} = 0.000000010101111$
$[-A]_{дк} = 1.111111100000010$
$[C3]_{пк} = 1.111111110110001 < 0;$

C4 = *-A-B*

$[-A]_{дк} = 1.111111100000010;$
$[-B]_{дк} = 1.111111101010001$
$[C4]_{пк} = 11.111001010011 < 0.$

Задания по теме «Представление чисел»

- Получить внутреннее двоичное представление числа 120 в однобайтовой ячейке.
- Получить внутреннее двоичное представление числа -127 в двухбайтовой ячейке.