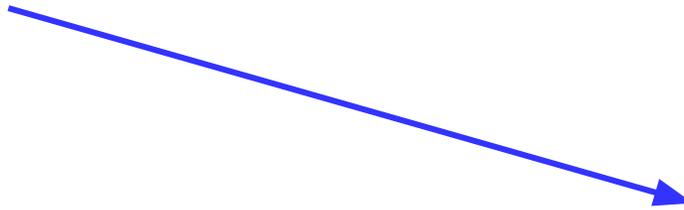


Представление информации в памяти ЭВМ

Системы счисления

Число – это некоторая величина

Система счисления – это способ записи чисел с помощью цифр



Непозиционная –
количественный эквивалент
(«вес») цифры
не зависит от её
положения в записи числа

CDXLIV

Позиционная –
количественный эквивалент
(«вес») цифры
зависит от её положения
в записи числа

444

Непозиционные системы счисления

Единичная (унарная)

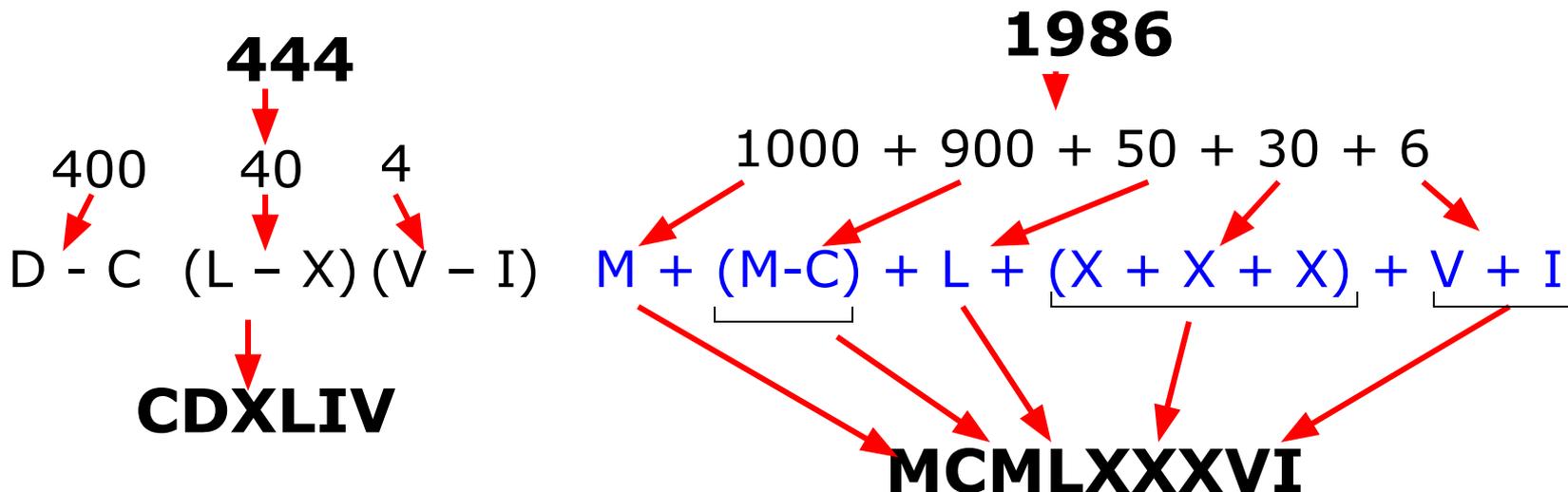
Египетская / единицы
∩ десятки

☪ сотни
⌊ тысячи

2376 =  ∩ - 1/10 ☪ - 2/3 ⌊ - 1/2

алфавитные системы

римская 1-I, 5-V, 10-X, 50-L, 100-C, 500-D, 1000-M



Позиционные системы счисления

Индийская мультипликативная система

X – десятки 323 → 3Y2X3
Y – сотни

Десятичная, двоичная, восьмеричная, шестнадцатеричная и др.

Основные достоинства любой позиционной системы

1. Простота выполнения арифметических действий
2. Ограниченное количество символов, необходимых для записи числа

Любое число в любой системе счисления можно представить с помощью развёрнутой формулы числа:

$$A = \pm (a_{n-1} g^{n-1} + a_{n-2} g^{n-2} \dots a_0 g^0 + a_1 g^{-1} + a_2 g^{-2} \dots a_m g^{-m})$$

A – само число

g – основание системы счисления

a – цифры данной системы счисления

n – число разрядов целой части числа

m – число разрядов дробной части числа

Во всех позиционных системах счисления арифметические операции выполняются по одним и тем же правилам:

1. справедливы одни и те же законы арифметики: коммутативный, ассоциативный, дистрибутивный;

Коммутативный закон: $a+b=b+a$

Ассоциативный закон: $a+(b+c)=(a+b)+c$

Дистрибутивный закон: $(a+b)c=ac+bc$

2. справедливы правила сложения, вычитания, умножения и деления столбиком;

3. правила выполнения арифметических операций опираются на таблицы сложения и умножения.

	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50

Перевод чисел из любой системы счисления в десятичную

Алгоритм

1. Представить число в развёрнутой форме. При этом основание системы счисления должно быть представлено в десятичной системе счисления
2. Найти сумму ряда. Полученное число является значением числа в десятичной системе счисления

Пример: $1101_2 \rightarrow A_{10}$

$$1) \quad 1101_2 = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$$

$$2) \quad 2^3 + 2^2 + 0 + 2^0 = 8 + 4 + 0 + 1 = 13_{10}$$

Задание:

Перевести в десятичную систему числа:

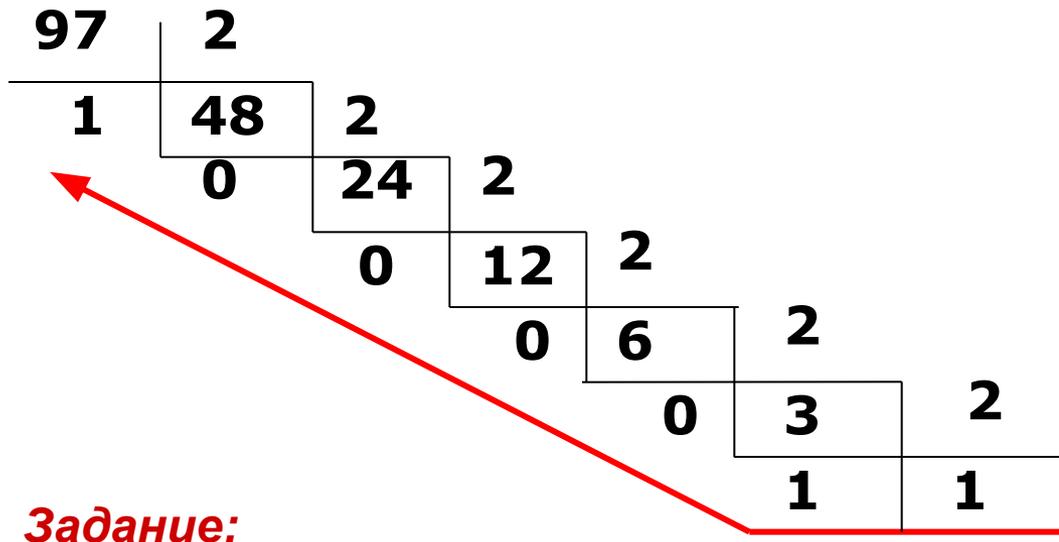
$$1111_2 = 15_{10}$$

$$1111_5 = 156_{10}$$

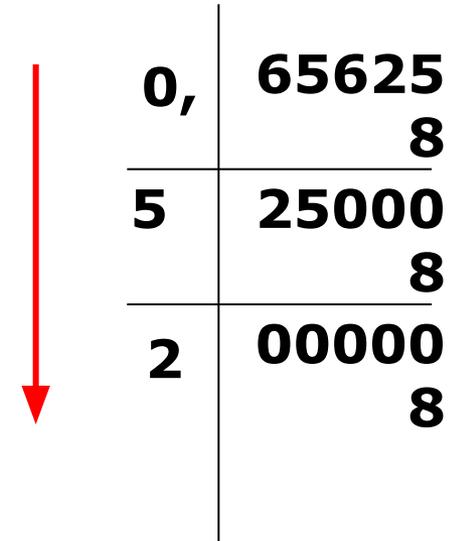
$$1111_3 = 40_{10}$$

Перевод чисел из десятичной системы счисления любую другую

Целое число



Дробное число



Задание:

Перевести число 356_{10} :

в восьмеричную $=544_8$

в двоичную $=101100100_2$

в пятеричную $=2411_5$

системы счисления

Системы счисления, используемые в ЭВМ

A_{10}	A_2	A_8	A_{16}
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Системы счисления, используемые в ЭВМ (с основанием 2^m)

Алгоритм перевода целых двоичных чисел в системах счисления с основанием 2^m

1. Двоичное число разбить справа налево на группы по n разрядов в каждой
2. Если в левой последней группе окажется меньше n разрядов, то её надо дополнить слева нулями до нужного числа разрядов
3. Рассмотреть каждую группу как n -разрядное двоичное число и записать её соответствующей цифрой в системе счисления с основанием $q=2^n$

$$\begin{array}{l} \mathbf{001100101001101010111}_2 \longrightarrow \mathbf{A}_8 \quad \mathbf{q=8=2^3} \\ \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \\ \mathbf{1 \quad 4 \quad 5 \quad 1 \quad 5 \quad 2 \quad 7} \end{array}$$
$$\begin{array}{l} \mathbf{01100101001101010111}_2 \longrightarrow \mathbf{A}_{16} \quad \mathbf{q=16=2^4} \\ \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \underbrace{\hspace{1em}} \\ \mathbf{6 \quad 5 \quad 3 \quad 5 \quad 7} \end{array}$$

Используя таблицу, перевести:

1. $10001101011001_2 \rightarrow A_8 \rightarrow A_{16} = 21531_8 = 2359_{16}$
2. $4AC2_{16} \rightarrow A_2 \rightarrow A_8 = 100101011000010_2 = 45302_8$
3. $713_8 \rightarrow A_2 \rightarrow A_{16} = 111001011_2 = 1CB_{16}$

Двоичная система счисления

Сложение

$$\begin{array}{r} + 101110101 \\ \quad \underline{1101101} \\ 111100010 \end{array}$$

Вычитание

$$\begin{array}{r} - 101110101 \\ \quad \underline{1101101} \\ 100001000 \end{array}$$

Умножение

$$\begin{array}{r} 101110101 \\ \quad \times \quad 1101 \\ \hline 101110101 \\ \quad 101110101 \\ \quad \quad 101110101 \\ \hline 1001011110001 \end{array}$$

Деление

$$\begin{array}{r} 101101 \quad | \quad \underline{101} \\ \underline{101} \quad \quad \quad 1001 \\ \quad 0101 \\ \quad \quad \underline{101} \\ \quad \quad \quad 0 \end{array}$$

Решить:

1. Произвести сложение, вычитание, умножение и деление двоичных чисел 1010_2 и 11_2
2. Сложить восьмеричные числа 5_8 и 4_8 , 17_8 и 41_8
3. Сложить числа 10_2 и 4_8
4. Прочитать число: MMIX
5. Записать число 3974 в римской системе счисления

Ответы:

1. 1101_2 , 111_2 , 100110_2 , 11_2 и 1 в остатке.
2. 11_8 , 60_8 .
3. 110_2
4. 2009
5. MMMCMLXXIV

Представление чисел в ЭВМ

Решение проблем математического моделирования в естественных науках, экономике и технике, работа с САПР, электронными таблицами невозможна без использования вещественных (действительных) чисел.

Все числовые данные хранятся в памяти компьютера в двоичном виде, т. е. в виде последовательностей нулей и единиц, однако формы хранения целых и вещественных чисел **различны**.

Необходимость различного представления целых и вещественных чисел вызвана тем, что скорость выполнения операций над целыми числами существенно выше, чем над вещественными числами.

Текстовая, графическая, звуковая информация, количество деталей, акций, сотрудников – эти и многие другие данные выражаются **целыми числами**.

Для решения математических и физических задач, в которых невозможно обойтись только целыми числами, используются **вещественные числа**.

Границы представления целых чисел

Целые числа могут быть представлены как **беззнаковые** - только неотрицательные, и как **знаковые** – положительные и отрицательные.

В зависимости от количества разрядов ячейки памяти границы представления целых чисел будут различными.

Разрядность	8	16	32
Минимум (без знака)	0	0	0
Максимум (без знака)	255	65 535	4 294 967 295
Минимум (со знаком)	- 128	- 32 768	- 2 147 483 648
Максимум (со знаком)	127	32 767	2 147 483 647

Представление целых чисел

Целые числа, как знаковые, так и беззнаковые, хранятся в формате с **фиксированной точкой**.

При таком представлении чисел все разряды ячейки, кроме знакового, если он есть, служат для изображения разрядов числа.

Причем каждому разряду ячейки соответствует один и тот же разряд числа. Именно поэтому такое представление называется с фиксированной точкой, так как **фиксируется место десятичной точки** перед определенным разрядом.

Для целых чисел десятичная точка находится после младшего разряда, то есть **вне разрядной сетки**.

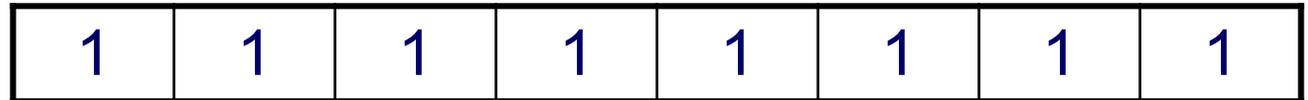
Форматы представления целых чисел

При представлении **беззнаковых** чисел все разряды ячейки отводятся под представление разрядов самого числа.

Минимальное
0



Максимальное
255



В случае представления **знаковых** целых чисел старший (левый) разряд ячейки отводится под хранение знака числа. В этот разряд заносится 0, если число положительное и 1 – если число отрицательное. Поскольку для хранения разрядов самого числа количество разрядов ячейки уменьшается на единицу, границы представления уменьшаются в два раза.

**Максимальное
знаковое
число** **127**



**Минимальное
знаковое
число** **-128**



Почему минимальное знаковое число в 8-разрядной ячейке –128, а максимальное +127?

Прямой код числа

Представление в форме «знак» - «величина», когда старший разряд ячейки отводится под знак, называется **прямым кодом** двоичного числа.

Число 1001_2

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

Положительные числа в ЭВМ всегда представляются с помощью прямого кода. Прямой код числа полностью совпадает с записью самого числа в ячейке памяти машины.

Для получения обратного кода числа все значения инвертируются.

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

Для положительного числа

Дополнительный код = прямому коду.

Например: Дано число 1001_2 . Записать его для 8-разрядной ячейки.

прямой код =

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

дополнительный код =

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

Дополнительный код

Отрицательные целые числа представляются в ЭВМ с помощью **дополнительного кода**.

Число 243 в одном байте будет выглядеть так:

Число 243_{10}

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

Но если эту запись рассматривать как запись числа со знаком, значением записи будет число -115_{10}

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

Дополнительный код отрицательного числа

Дополнительный код отрицательного числа $m = 2^k - |m|$, где k – количество разрядов в ячейке,
Дополнительный код отрицательного числа –
это дополнение $|m|$ до 2^k .

Если $k=8$, $|m| = 01100101_2$, то дополнительный код можно получить как разность $10000000_2 - 01100101_2 = 00011011_2 \rightarrow 10011011_2$

Или $01100101_2 + 00011011_2 = 10000000_2$ ($155+101=256$)
Алгоритм получения дополнительного кода

отрицательного числа в n двоичных разрядах:

1. Модуль числа записать в прямом коде в n двоичных разрядах.
2. Получить обратный код числа.
3. К полученному обратному коду прибавить единицу.

Например: записать дополнительный код отрицательного числа -2002 для 16-разрядного компьютерного представления.

Прямой код: $|-2002_{10}| = 2002_{10} = 0000011111010010_2$

Обратный код: 1111100000101101_2

Дополнительный код: $1111100000101101_2 + 1_2$

Нормализованная запись чисел

Недостатком представления чисел с фиксированной точкой является небольшой диапазон представляемых величин, недостаточный для решения математических, физических, экономических и других задач.

Вещественные (дробные) числа хранятся и обрабатываются в компьютере в формате с **плавающей точкой**. Формат чисел с плавающей точкой базируется на экспоненциальной форме записи, в которой может быть представлено любое число.

$$A = m * q^n$$

где m – мантисса числа
 q - основание системы счисления
 n - порядок числа

Для единообразия представления чисел с плавающей точкой, используется нормализованная форма, при которой мантисса отвечает условию:

$$1/ n \leq |m| < 1$$

Это означает, что мантисса должна быть правильной дробью и иметь после запятой цифру отличную от нуля.

Примеры нормализации чисел:

$$1) \quad 3.1415926 = 0.31415926 \cdot 10^1$$

$$2) \quad 1000 = 0.1 \cdot 10^4$$

$$3) \quad -0.123456789 = -0.123456789 \cdot 10^0$$

$$4) \quad 0.0000107_8 = 0.107_8 \cdot 8^{-4}$$

$$5) \quad 1000.0001_2 = 0.10000001_2 \cdot 2^4$$

$$6) \quad -0.0001101_2 = -0.1101_2 \cdot 2^{-3}$$

Запись нуля считается нормализованной, если и мантисса, и порядок равны нулю, т. е. **$0 = 0.0 \cdot 10^0$**

Числа в формате с плавающей точкой занимают в памяти компьютера 4 (число обычной точности) или 8 байтов (число двойной точности).

Компьютерное представление вещественных чисел

Как и для целых чисел, при представлении вещественных чисел используется двоичная система счисления, поэтому предварительно число должно быть переведено в двоичную систему.

При представлении чисел с плавающей точкой в разрядах ячейки отводится место для знака числа, знака порядка, абсолютной величины порядка, абсолютной величины мантиссы.

знак числа (-)

абсолютная величина порядка (13)

1	0	00001101	1011011000010111100110
---	---	----------	------------------------

знак порядка (+)

абсолютная величина мантиссы (5826486)

В ячейке записано отрицательное двоичное число -1011011000010.11110011

В десятичном представлении это будет число -5826.486

Контрольные задания

1. Как будут представлены в 8-битном знаковом типе числа:

а) -1 ; б) -10 ; в) -120 ; г) -102 ;

2. Запишите следующие двоичные числа в прямом, обратном и дополнительном коде для 8-разрядной ячейки:

а) -1000 ; б) -11101 ; в) -1 ; г) -1111111 ;

3. Приведите к нормализованному виду числа, оставляя их в тех же системах счисления, в которых они записаны:

а) -0.000001011101_2 ; б) 987654321_{10} ; в) 100.01_2 ; г) -0.001502_8 ;

4. Запишите в естественной форме с фиксированной запятой следующие нормализованные числа:

а) $0.1011_2 \cdot 2^1$; б) $0.1011_2 \cdot 2^{11}$;

в) $0.12345_{10} \cdot 10^{-3}$; г) $-0.40065_8 \cdot 8^{-4}$;