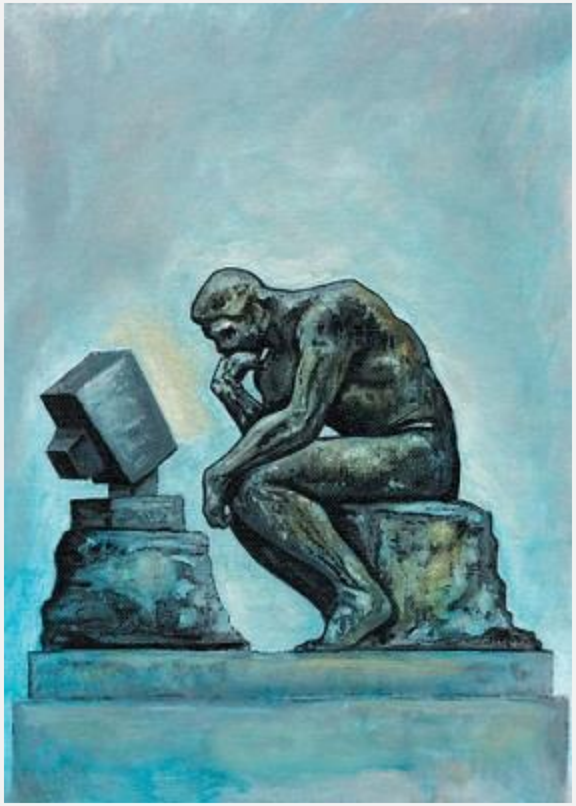
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Государственное автономное образовательное учреждение высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Тема 1.2. Кодирование информации

Преподаватель каф. ЭАФУ
Егорова Ольга Викторовна

Томск -2017



- системы счисления
- преобразование чисел из одной системы счисления в другую
- основные арифметические операции над двоичными числами
- представление числовой, символьной, звуковой и графической информации в ЭВМ

СИСТЕМА СЧИСЛЕНИЯ (СС)

совокупность приемов записи и наименования чисел

Непозиционная

система, в которой цифры не меняют своего количественного значения при смене позиции в числе

Позиционная

величина числа зависит не только от набора цифр, но и от того, в какой последовательности записаны цифры, т.е. от позиции, занимаемой цифрой

ПРИМЕРЫ

Римская система счисления

Алфавит: I V X L C D M

XXX

10 + 10 +10

||

30

Значение цифры не зависит от позиции в числе

Арабская система счисления

Алфавит: 0 1 2 3 4 5 6 7 8 9

333

3 x 100 +3 x 10 + 3 x 1

||

333

Значение цифры зависит от позиции в числе

Современные цифры	Арабские цифры
0	٠
1	١
2	٢
3	٣
4	٤ ۴
5	٥ ۵
6	٦ ۶
7	٧
8	٨
9	٩

Основание системы счисления (P) – количество цифр, используемых для записи числа

Представление числа (X) в позиционной СС с основанием P

$$X_{(p)} = X_{n-1} \cdot P^{n-1} + X_{n-2} \cdot P^{n-2} + \ldots + X_1 \cdot P^1 + X_0 \cdot P^0 + X_{-1} \cdot P^{-1} + X_{-2} \cdot P^{-2} + \ldots + X_{-m} \cdot P^{-m} \quad (1)$$

$$X_{(P)} = \underbrace{X_{n-1} X_{n-2} X_1 X_0 , X_{-1} X_{-2} X_{-m}}_{\text{закодированная запись числа в P-ой СС}}$$

$X_{(p)}$ – запись числа в системе счисления с основанием P

X_i – значащие числа в пределах от 0 до P-1

n – количество разрядов целой части числа

m – количество разрядов дробной части числа

Максимальное значащее число целой части числа в P-ой СС :

$$R_{max} = P^{n} - 1$$

Минимальное значащее (не равное 0) число дробной части числа в P-ой СС :

$$R_{min} = P^{-m}$$

Характеристики широко используемых СС:

Система счисления	**Основание**	**Используемые символы**
десятичная	10	0 - 9
двоичная	2	0 -1
восьмеричная	8	0 - 7
шестнадцатеричная	16	0 – 9, 10 – A, 11 – B, 12 – C, 13 – D, 14 – E, 15 - F

Примеры представления чисел в позиционной СС

Десятичное число

$$357{,}4_{(10)} = 3 \cdot 10^{2} + 5 \cdot 10^{1} + 7 \cdot 10^{0} + 4 \cdot 10^{-1}$$

357,4

Двоичное число

$$10110{,}1_{(2)} = 1 \cdot 2^{4} + 0 \cdot 2^{3} + 1 \cdot 2^{2} + 1 \cdot 2^{1} + 0 \cdot 2^{0} + 1 \cdot 2^{-1}$$

10110,1

$$1 \cdot 2^{4} + 0 \cdot 2^{3} + 1 \cdot 2^{2} + 1 \cdot 2^{1} + 0 \cdot 2^{0} + 1 \cdot 2^{-1} =$$

$$16+4+2+0{,}5=22{,}5_{(10)}$$

Восьмеричное число

$$357{,}4_{(8)} = 3 \cdot 8^{2} + 5 \cdot 8^{1} + 7 \cdot 8^{0} + 4 \cdot 8^{-1} = 239{,}5_{(10)}$$

357,4

Шестнадцатеричное число

$$3BE{,}4 = 3 \cdot 16^{2} + 11 \cdot 16^{1} + 14 \cdot 16^{0} + 4 \cdot 16^{-1} = 58{,}25_{(10)}$$

3BE,4

Правила преобразования числа из 10-й СС в другие

Число Х = А,В (пример 10,25)

Прямой перевод:

Целая часть числа:

1) **Целая часть числа** делится последовательно на основание новой СС на (**2 (8 или 16) для 2, 8 и 16-й СС**) , так чтобы остатки от деления были меньше основания новой СС (**1 для 2-й СС, 1-7 для 8-й СС и т.д.**).

2) Делим до тех пор, пока результат деления не станет равен **1** или меньше **1**.

3) Новое число составляется путем записи полученных **остатков** от деления в **обратном порядке**.

Дробная часть числа:

1) **Дробная часть числа** умножается на основание новой СС. Целая часть результата умножения используется для записи числа, а дробная снова умножается на основание новой СС.

2) Умножение продолжается до тех пор, пока дробная часть не станет равной **нулю** или не будет достигнута **требуемая точность** перевода.

3) Новое число составляется путем записи целых частей результатов умножения в **прямом порядке**.

Обратный перевод:

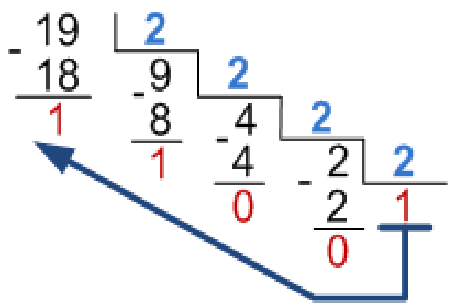
число представляется полиномом (1) (см. слайд 4) и рассчитывается соответствующая сумма.

ПРИМЕР 1

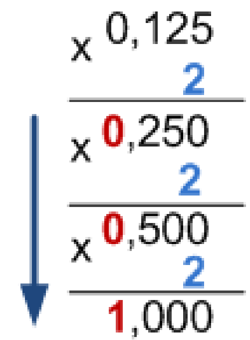
Переведем десятичное число **19,125$_{(10)}$** в двоичную систему счисления и затем из двоичной обратно

Прямой перевод:

целой части числа (19):



дробной части числа (0,125):



Результат перевода: $19{,}125_{(10)} = \mathbf{10011{,}001}_{(2)}$

Обратный перевод: **(здесь n=5, m=3, P=2)**

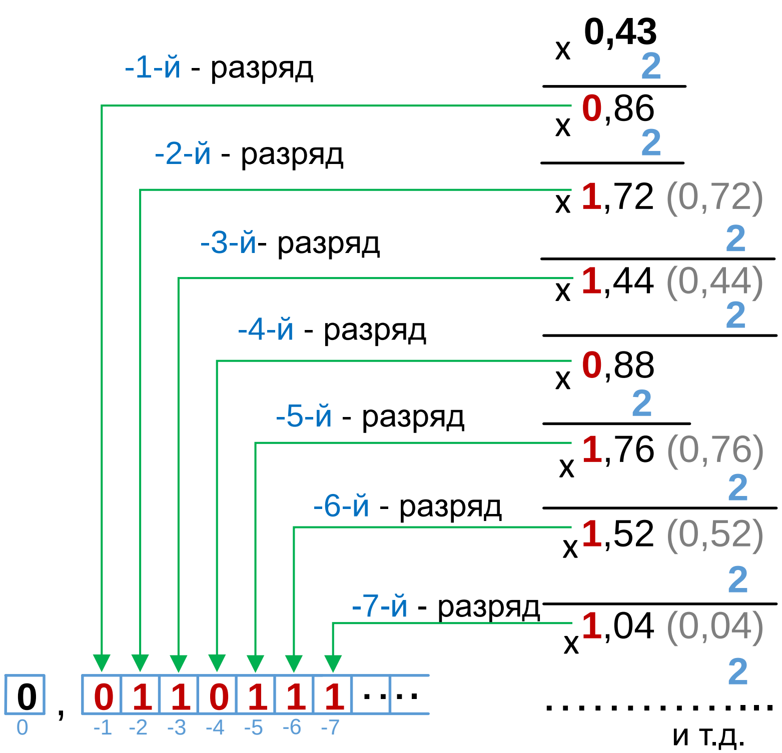
$$X_{(p)} = X_{n-1}\cdot P^{n-1} + X_{n-2}\cdot P^{n-2} + \ldots + X_1\cdot P^1 + X_0\cdot P^0 + X_{-1}\cdot P^{-1} + X_{-2}\cdot P^{-2} + \ldots + X_{-m}\cdot P^{-m}$$

$$10011{,}001 = 1\cdot 2^4 + 0\cdot 2^3 + 0\cdot 2^2 + 1\cdot 2^1 + 1\cdot 2^0 + 0\cdot 2^{-1} + 0\cdot 2^{-2} + 1\cdot 2^{-3} = 19{,}125$$

Примеры преобразования числа из 10-й СС в другие

ПРИМЕР 2

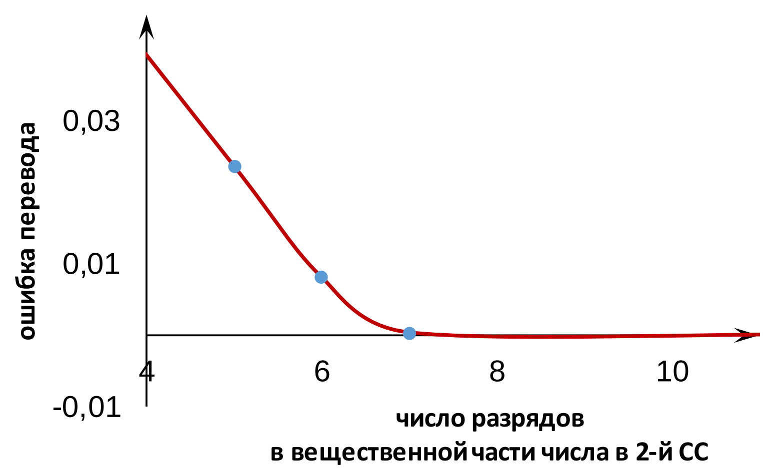
Переведем десятичное число **0,43**$_{(10)}$ в двоичную систему счисления



запись числа **0,43** в 2-ой СС

Ошибка преобразования в 2 СС:

Число разрядов в вещественной части числа в 2-й СС	Исходное число в 10 СС (N)	Число в 2-й СС	Число после обратного перевода из 2-й в 10-ю СС (M)	Ошибка перевода (N – M)
5	0,43	0,01101	**0,40625**	0,02375
6		0,011011	**0,421875**	0,008125
7		0,0110111	**0,4296875**	0,000312



Особенности преобразования из 10-ой СС в 2-ю СС

Следует отметить, что

- в **10-й СС** правильная дробь переводится в десятичную дробь в конечном виде только в том случае, если она имеет вид $\dfrac{k}{2^m \cdot 5^n}$
- аналогично в **2-й СС** конечный вид получают дроби, где в знаменателе только степени двойки
- большинство десятичных конечных дробей в **2-й СС будут бесконечными периодическими дробями**
- если ведутся приближенные вычисления, то *последний разряд является сомнительным*, и для обеспечения в приближенных вычислениях *одинаковой* точности в 2-й и 10-й записях числа без бесконечных дробей, достаточно взять число 2-х разрядов в ($\log_2 10 = 3{,}3$) 4 разряда больше, чем 10-х разрядов

Прямой перевод:

1) исходное двоичное число разбивается на **триады** для 8-й СС (**тетрады** для 16-й СС)

триада – группа из 3-х цифр;
тетрада – группа из 4-х цифр;

2) затем каждая **триада** (**тетрада**) заменяется соответствующей 8-ой (16-ой) цифрой

3) для целой части числа **триады** (**тетрады**) формируются от запятой справа налево, для дробной – слева направо

4) недостающие двоичные цифры заменяются 0

Обратный перевод:

каждая восьмеричная (шестнадцатеричная) цифра числа заменяется соответствующей **триадой** (**тетрадой**) из двоичных цифр

Примеры преобразования числа из 2 СС в 8 или 16 СС

ПРИМЕР 1. Переведем двоичное число **$11101111{,}1_{(2)}$** в восьмеричную и шестнадцатеричную СС

Перевод в 8-ю СС

$$\underbrace{011}_{3}\ \underbrace{101}_{5}\ \underbrace{111}_{7},\ \underbrace{100}_{4}$$

Результат перевода:

$$011\ 101\ 111{,}100_{(2)} \longrightarrow 357{,}4_{(8)}$$

Перевод в 16-ю СС

$$\underbrace{1110}_{E}\ \underbrace{1111}_{F},\underbrace{1000}_{8}$$

Результат перевода:

$$1110\ 1111{,}1000_{(2)} \longrightarrow EF{,}8_{(16)}$$

Примеры преобразования числа из 2 СС в 8 или 16 СС

ПРИМЕР 2. Переведем восьмеричное число **$315_{(8)}$** и шестнадцатеричное **$3BE,4_{(16)}$** в двоичную СС.

Перевод числа 315 в 2-ю СС

3 1 5

011 001 101

Результат перевода:

$315_{(8)} \longrightarrow 011001101_{(2)}$

Перевод числа 3BE,4 в 2-ю СС

3 B E, 4

0011 1011 1110, 0100

Результат перевода:

$3BE,4_{(16)} \longrightarrow 001110111110,0100_{(2)}$

Основные арифметические операции над двоичными числами

Правила выполнения операций для одноразрядных двоичных чисел			
Сложение	**Вычитание**	**Умножение**	**Деление**
0 + 0 = 0 0 + 1 = 1 1 + 0 = 1 1 + 1 =10	0 - 0 = 0 1 - 0 = 1 1 - 1 = 0 10 - 1 = 1	0 · 0 = 0 0 · 1 = 0 1 · 0 = 0 1 · 1 = 1	
ПРИМЕРЫ			
+ 1100 (12) 110 (6) 10010 (18)	- 10010 (18) 101 (5) 01101 (13)	× 1001 (9) 110 (6) 0000 + 1001 1001 110110 (54)	- 110110 (54) | 110 110 (6) | 1001 - 000110 110 000

.

Формы представления чисел

С фиксированной точкой

С плавающей точкой

Число представляется в виде:

последовательности двоичных цифр, разделенной **точкой** на целую и дробную части

Точка фиксируется:

- **перед старшим разрядом** (для чисел по модулю < 1)
- **справа от младших разрядов** (для целых чисел)

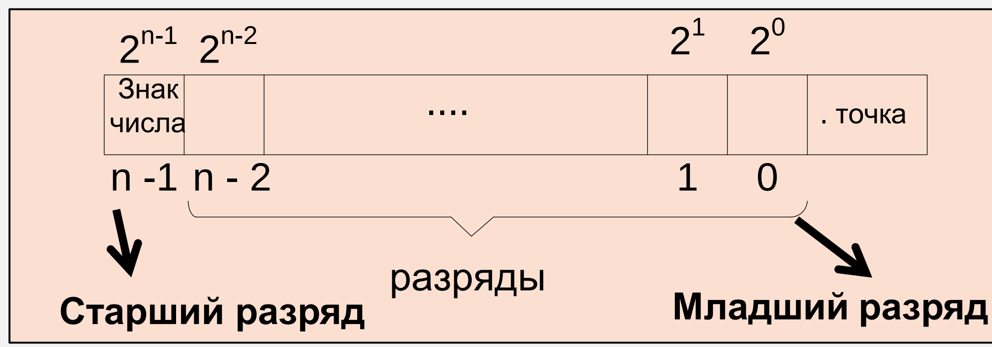
Используются два формата:

- со знаком («-» кодируется 1, «+» - 0)
- без знака

Диапазон хранимых чисел:

- со знаком - $[-2^{n-1}, +2^{n-1} - 1]$
- без знака - $[0, 2^n - 1]$

Представление числа в ЭВМ в форме с фиксированной точкой



ПРИМЕР для числа $Y = -5_{10} = -101_2$

Знаковый разряд

1	0	0	0	0	1	0	1
7	6	5	4	3	2	1	0

→ 8 разрядная сетка ЭВМ

Число представляется в виде:

$$\boxed{N = M \times P^{R}}$$

где

P – основание системы счисления

M – мантисса

R – порядок числа

Например, число 4235,25 можно записать:

$4235,25 \times 10^{0}$

$423,525 \times 10^{1}$

$42,3525 \times 10^{2}$

$4,23525 \times 10^{3}$

$0,423525 \times 10^{4}$

P=10 M=0,423525 R=4

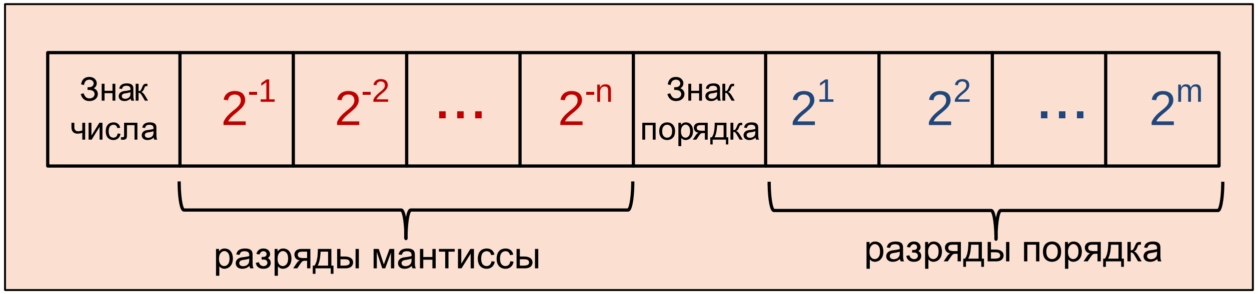
Условия нормализации числа:

$$\boxed{\frac{1}{P} \le |M| < 1}$$

для десятичной СС: $0,1 \le |M| < 1$

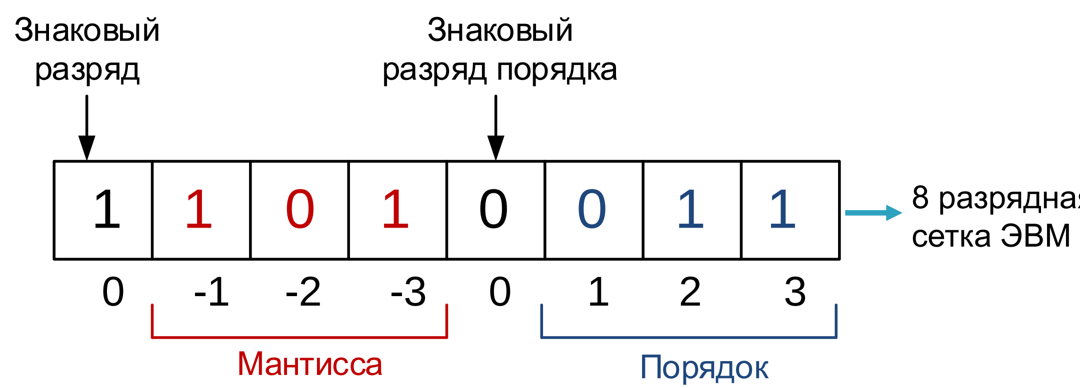
т.е. первая цифра мантиссы после запятой должна быть значимой (≠ 0)

Представление числа в ЭВМ в форме с плавающей точкой



ПРИМЕР для числа $Y = -5_{10} = -101_2$

В нормализованной форме с плавающей точкой Y: -0.101×2^3



мантисса числа Y: $-101_2 = 1101_2$

порядок числа Y: $+3_{10} = 0011_2$

Разрядная сетка ЭВМ - совокупность двоичных разрядов, предназначенных для хранения и обработки чисел

Бит (один разряд двоичного кода) - наименьшая единица информации

Байт - группа из 8-и двоичных разрядов

Машинное слово – группа из нескольких байтов, количество которых зависит от разрядности ЭВМ

Примеры разрядных сеток для формы с фиксированной точкой

16 разрядная ячейка для хранения целых со знаком
(половина машинного слова)

Знак числа	2^{14}	2^{13}	…..	2^{0}

диапазон хранимых значений
$[-2^{15}, +2^{15}-1]$

16 разрядная ячейка для хранения целых без знака
(половина машинного слова)

2^{15}	2^{14}	2^{13}	….	2^{1}	2^{0}

диапазон хранимых значений
$[0, 2^{16}-1]$

32 разрядная ячейка для хранения целых со знаком
(целое машинное слово)

Знак числа	2^{31}	…	2^{0}

диапазон хранимых значений
$[-2^{31}, +2^{31}-1]$

Представление числа в форме с плавающей точкой в коротком формате фиксированной длины

Разрядная сетка для хранения вещественного числа в коротком формате:

31 бит	30 – 24 бит	23 – 16 бит	15 – 8 бит	7 – 0 бит
знак числа	машинный порядок	М АН	Т И С	С А
1-й байт		2-й байт	3-й байт	4-й байт

Соответствие между машинным порядком и истинным (математическим):

Машинный порядок	0	1	2	3	...	64	65	....	125	126	127
Математический порядок	-64	-63	-62	-61	...	0	1	....	61	62	63

В 10-ой системе счисления:

$$Mp = p + 64$$

В 2-ой системе счисления:

$$Mp_2 = p_2 + 100\ 0000_2$$

где Mp - машинный порядок, p - математический

Пример представления числа 25,324 в коротком формате фиксированной длины с плавающей точкой

1) Переведем данное число в 2-ю СС с 24 значащими цифрами:

$$25{,}324_{10} = 11001{,}0101001011110001101_2$$

2) Запишем в форме нормализованного 2-го числа с плавающей точкой:

$$0{,}110010101001011110001101 \times 10^{101}$$

где

мантисса (0,110010101001011110001101)

основание системы счисления **($2_{10}=10_2$)**

порядок **($5_{10}=101_2$) записаны в двоичной системе**

Пример представления числа 25,324 в формате с плавающей точкой

3) Вычислим машинный порядок:

$$Mp_2 = 101 + 100\ 0000 = 100\ 0101$$

4) Запишем представление числа в ячейке памяти:

31	30 – 24 бит	23 – 16 бит	15 – 8 бит	7 – 0 бит
0	1000101	11001010	10010111	10001101

Внутреннее представление отрицательного числа -25,324:

31	30 – 24 бит	23 – 16 бит	15 – 8 бит	7 – 0 бит
1	1000101	11001010	10010111	10001101

Понятие скрытого бита в формате с плавающей точкой фиксированной длины

Короткий формат (в соответствии со стандартом IEEE 754)

31	30	23	22	0
Знак числа	e_7	e_0	**1 .** f22	f0

скрытый бит **десятичная точка**

где

$e_0 \ldots e_7$ – разряды машинного порядка

$f_0 \ldots f_{22}$ – разряды мантиссы

Соотношение между машинным и математическим порядками:

$Mp_{10} = p + 127$ $Mp_2 = p_2 + 111\ 1111_2$

где Mp - машинный порядок, p - математический

Диапазон нормализованной мантиссы (в 10 СС):

$$1 \le |M| < 2$$

О диапазоне чисел, представимых в форме с плавающей точкой

$$R_{max} = |R_{min}|$$

где R_{max}, R_{min} - максимальное и минимальное число, в форме с плавающей точкой

Максимальное число (R_{max}) – это число с самой большой мантиссой и самым большим порядком

Для рассмотренного примера:

- $R_{max} = 0{,}111111111111111111111111 \cdot 10_2^{1111111}$ [$(1 - 2^{-24}) \cdot 2^{64} = 10^{19}$ в 10-ой СС]
- Наименьшее по модулю ненулевое значение: $(1/2) \cdot 2^{-64} = 2^{-65}$

Количество вещественных чисел, точно представимых в памяти машины:

$$N = 2^n \cdot (U - L + 1) + 1$$

где n — количество двоичных разрядов мантиссы

U — максимальное значение математического порядка

L — минимальное значение порядка

Для рассмотренного примера (n = 24, U = 63, L = -64):

N = **2 146 683 548**

ПРИМЕР

(представления двухбайтного целого числа **-5**)

Типы кодов

- **Прямой**

Знак числа	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

- **Обратный**

Знак числа	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0

- **Дополнительный**

Знак числа	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1

- **Модифицированные**

Знаковые разряды		2^{13}	2^{12}			….								2^{0}	
1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	обратный

Знаковые разряды		2^{13}	2^{12}			….								2^{0}	
1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	дополнительный

Сложение и вычитание чисел в формах с фиксированной и плавающей точками

Операция **вычитания** **заменяется** операцией **сложения** с использованием **обратного** или **дополнительного** кодов

В форме с фиксированной точкой

Этапы выполнения:

- преобразование прямых кодов складываемых (вычитаемых) чисел в обратные (или дополнительные)
- поразрядное сложение получившихся кодов (с учетом возможных переносов из разряда в разряд)
- преобразование результата в прямой код при его передаче в другие устройства ЭВМ

Следует помнить:

- ✓ при поразрядном сложении обратных (или дополнительных) кодов знаковые разряды складываются как и числовые
- ✓ если в результате сложения образуется единица переноса из знакового разряда, то в случае использования:
  - обратных кодов ее **прибавляют** к младшему разряду суммы
  - дополнительных кодов этот перенос **не учитывают**

Возможны четыре случая:

1 - ый случай: $X > 0, Y > 0, X + Y > 0$

Этот случай не отображает особенностей использования обратного и дополнительного кодов, так как для положительных чисел эти коды совпадают с прямым

2 - ой случай: $X > 0, Y < 0, X + Y < 0$

ПРИМЕР (X – Y = 3 – 6 = -3)

Число в 10 СС	Коды		
	Прямой	Обратный	Дополнительный
X = 3	0011	0011	0011
Y = -6	1110	1001	1010

В обратных кодах:

$$\begin{array}{r} + \begin{array}{r}[X]_{обр} = 0011 \\ [Y]_{обр} = 1001\end{array} \\ \hline [X+Y]_{обр} = 1100 \end{array}$$

В дополнительных кодах:

$$\begin{array}{r} + \begin{array}{r}[X]_{доп} = 0011 \\ [Y]_{доп} = 1010\end{array} \\ \hline [X+Y]_{доп} = 1101 \end{array}$$

Переведем полученные результаты в прямой код:

$$\left.\begin{array}{l} [1100]_{обр} \rightarrow [1011]_{пр} \\ [1101]_{доп} \rightarrow [1100]_{обр} \rightarrow [1011]_{пр} \end{array}\right\} -011_2 = -3_{10}$$

Сложение и вычитание чисел в форме с фиксированной точками

3 - ий случай: $X > 0, Y < 0, X + Y > 0$

ПРИМЕР (X – Y = 6 – 3 = 3)

Число в 10 СС	Коды		
	Прямой	Обратный	Дополнительный
X = 6	0110	0110	0110
Y = -3	1011	1100	1101

В обратных кодах:

$$\begin{array}{r} + \begin{array}{r} [X]_{обр} = 0110 \\ [Y]_{обр} = 1100 \end{array} \\ \hline 10010 \\ + \quad 1 \\ \hline [X+Y]_{обр} = 0011 \end{array}$$

В дополнительных кодах:

$$\begin{array}{r} + \begin{array}{r} [X]_{доп} = 0110 \\ [Y]_{доп} = 1101 \end{array} \\ \hline [X+Y]_{доп} = 10011 = 0011 \end{array}$$

Переведем полученные результаты в прямой код:

$[0011]_{обр} = [0011]_{доп} = [0011]_{пр} \longrightarrow 011_2 = 3_{10}$

4 - ий случай: $X < 0, Y < 0, X + Y < 0$

ПРИМЕР (-X – Y = -3 – 4 = -7)

Число в 10 СС	Коды		
	Прямой	Обратный	Дополнительный
X = -3	1011	1100	1101
Y = -4	1100	1011	1100

В обратных кодах:

$$\begin{array}{r} + \begin{array}{r} [X]_{обр} = 1100 \\ [Y]_{обр} = 1011 \end{array} \\ \hline 10111 \\ + \quad 1 \\ \hline [X+Y]_{обр} = 1000 \end{array}$$

В дополнительных кодах:

$$\begin{array}{r} + \begin{array}{r} [X]_{доп} = 1101 \\ [Y]_{доп} = 1100 \end{array} \\ \hline [X+Y]_{доп} = 11001 = 1001 \end{array}$$

Переведем полученные результаты в прямой код:

$$\left.\begin{array}{l} [1000]_{обр} \rightarrow [1111]_{пр} \\ [1001]_{доп} \rightarrow [1000]_{обр} \rightarrow [1111]_{пр} \end{array}\right\} -111_2 = -7_{10}$$

Сложение и вычитание чисел в форме с плавающей точкой

Этапы выполнения:

- выравниваются порядки слагаемых, для чего меньший порядок увеличивается до большего, а его мантисса нормализуется на соответствующее число разрядов (порядком суммы является общий порядок слагаемых)
- складываются мантиссы с использованием обратного или дополнительного кода (обычного или модифицированного). Сложение выполняется по тем же правилам, как и в форме с фиксированной точкой

Возможны три случая:

1 - ый случай: сложение происходит без переполнения разрядной сетки и нарушения нормализации

ПРИМЕР (- X + Y = -2 + 7 = 5)

Число в 10 СС	Число в 2 СС	Нормальная форма
X = -2	-10	$-0{,}10 \cdot 2^{10}$
Y = 7	+111	$+0{,}111 \cdot 2^{11}$

Сложение и вычитание чисел в форме с плавающей точками

Запись чисел в формате с плавающей точкой:

Число в нормальной форме	Знак мантиссы	Мантисса	Знак порядка	Порядок
$X = -0{,}10 \cdot 2^{10}$	1	100	0	10
$Y = +0{,}111 \cdot 2^{11}$	0	111	0	11

Выровняем порядки чисел X и Y, для этого увеличим порядок числа X на 1, путем смещения его мантиссы на 1 разряд в право:

Число	Знак мантиссы	Мантисса	Знак порядка	Порядок
$X = -0{,}010 \cdot 2^{11}$	1	010	0	11
$Y = +0{,}111 \cdot 2^{11}$	0	111	0	11

Сложение и вычитание чисел в форме с плавающей точками

Выполним сложение мантисс чисел X и Y в модифицированном обратном коде:

$$
\begin{array}{r}
+\begin{array}{l}[X]^{M}_{обр} = 11101 \\ [Y]^{M}_{обр} = 00111\end{array} \\
\hline
100100 \\
+\quad 1 \\
\hline
[X+Y]^{M}_{обр} = 00101
\end{array}
$$

Переведем полученные результаты в прямой код:

Число	Знак мантиссы	Мантисса	Знак порядка	Порядок
X + Y	0	101	0	11

Что соответствует: $X + Y = +0{,}101 \cdot 2^{11} = +101_2 = +5_{10}$

Сложение и вычитание чисел в форме с плавающей точками

2 - ой случай: сложение происходит без переполнения разрядной сетки, но результат после перевода в прямой код оказывается ненормализованным (старший разряд мантиссы равен нулю – нарушение нормализации вправо)

Нормализацию результата выполняют: сдвигом мантиссы влево и соответствующим уменьшением порядка суммы

ПРИМЕР (X – Y = 5 - 7 = - 2)

Число в 10 СС	Число в 2 СС	Нормальная форма
$X = 5$	+101	$+0{,}101 \cdot 2^{11}$
$Y = -7$	-111	$-0{,}111 \cdot 2^{11}$

Запись чисел в формате с плавающей точкой:

Число в нормальной форме	Знак мантиссы	Мантисса	Знак порядка	Порядок
$X = +0{,}101 \cdot 2^{11}$	0	101	0	11
$Y = -0{,}111 \cdot 2^{11}$	1	111	0	11

Выравнивать порядки не требуется, та как они одинаковые и равны $3_{10} = 11_2$

Выполним сложение мантисс чисел X и Y в модифицированном обратном коде:

$$
\begin{array}{r}
[X]^{М}_{обр} = 00101 \\
+\; [Y]^{М}_{обр} = 11000 \\
\hline
[X+Y]^{М}_{обр} = 11101
\end{array}
$$

Переведем полученные результаты в прямой код:

Число	Знак мантиссы	Мантисса	Знак порядка	Порядок
X + Y	1	010	0	11

Нормализуем результат сдвигом мантиссы влево на один разряд и соответствующим уменьшением порядка:

Число	Знак мантиссы	Мантисса	Знак порядка	Порядок
X + Y	1	100	0	10

Что соответствует: $X + Y = -0{,}100 \cdot 2^{10} = -10_2 = -2_{10}$

Сложение и вычитание чисел в форме с плавающей точками

3 - й случай: при сложении мантиссы происходит переполнение разрядной сетки (модуль суммы ≥ 1 – нарушение нормализации влево)

Признак переполнения разрядной сетки (при использовании модифицированных обратных кодов): в знаковых разрядах появляются разные цифры (01- для положительной суммы, 10 – для отрицательной, *знак суммы устанавливается по левому из 2-х знаковых разрядов*)

Нормализация результата: путем сдвига результата суммирования на один разряд вправо, увеличением порядка на единицу и внесением в правый знаковый разряд цифры, стоящей в левом знаковом разряде

ПРИМЕР (-X – Y = -2 - 7 = - 9)

Число в 10 СС	Число в 2 СС	Нормальная форма
X = -2	-10	$-0{,}10 \cdot 2^{10}$
Y = -7	-111	$-0{,}111 \cdot 2^{11}$

Запись чисел в формате с плавающей точкой:

Число в нормальной форме	**Знак мантиссы**	**Мантисса**	**Знак порядка**	**Порядок**
$X = -0{,}10 \cdot 2^{10}$	1	1000	0	10
$Y = -0{,}111 \cdot 2^{11}$	1	1110	0	11

Выровняем порядки чисел X и Y:

Число	**Знак мантиссы**	**Мантисса**	**Знак порядка**	**Порядок**
$X = -0{,}0100 \cdot 2^{11}$	1	0100	0	11
$Y = -0{,}1110 \cdot 2^{11}$	1	1110	0	11

Выполним сложение мантисс чисел X и Y в модифицированном обратном коде:

$$
\begin{array}{r}
[X]^{М}_{обр} = 111011 \\
+\ [Y]^{М}_{обр} = 110001 \\
\hline
1101100 \\
+\ 1 \\
\hline
[X+Y]^{М}_{обр} = 101101
\end{array}
$$

Произошло нарушение нормализации влево. После сдвига на один разряд вправо и увеличения порядка суммы на единицу переведем результат в прямой код:

$$X + Y = -0{,}1001 \cdot 2^{100} = -1001_2 = -9_{10}$$

Представление символьной информации в ЭВМ

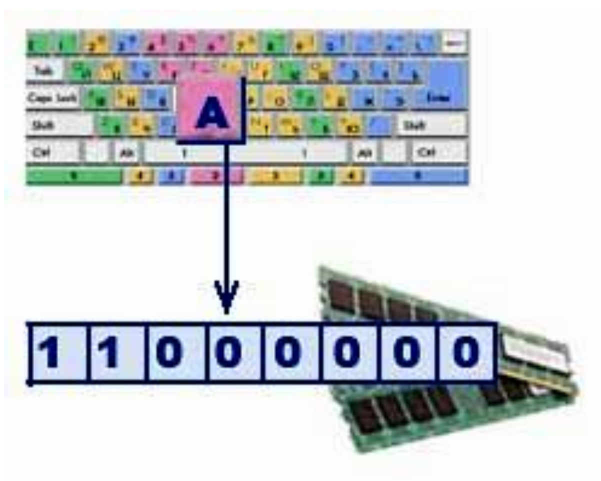
Алфавит - множество символов, с помощью которых записывается текст

Мощность алфавита (N) - число символов в алфавите

Число символов определяем по: $N = 2^b$
где b – количество бит (информационный вес символа)

Кодовая таблица - таблица соответствий символов и их компьютерных кодов

Кодовые таблицы – ASCII, КОИ-8, ISO, Unicode и др.



Кодовая таблица ASCII

№	Символ	Код
0		
...		
31		
32	Пробел	
33	·	
...	...	
71	G	01000111
...		
103		
...		
127	O	
128	А	
...	...	
159	Я	
	...	
255		

Управляющие символы

Латинские символы

Национальный алфавит

1 символ = 8 бит =1 байт

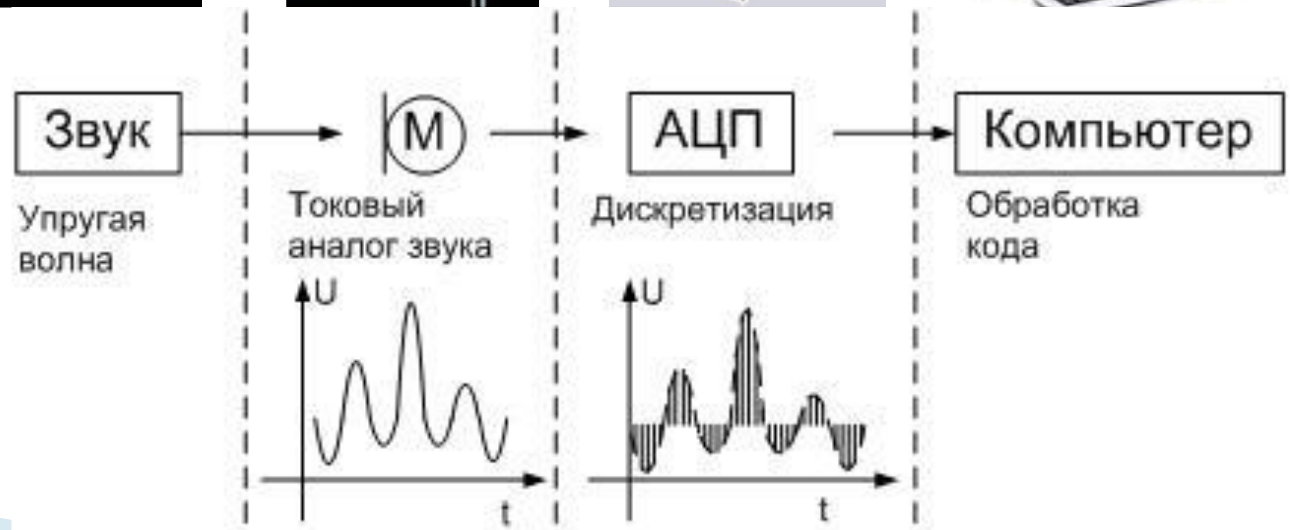
Мощность алфавита = 256 символов

ЗВУК - упругая волна, продольно распространяющаяся в какой-либо среде и создающая в ней механические колебания

ПРОЦЕДУРА «ОЦИФРОВКИ» ЗВУКА







ДИСКРЕТИЗАЦИЯ ПО ВРЕМЕНИ

Частота дискретизации: $\upsilon = \dfrac{1}{\Delta t}$ (в герцах)

где Δt – шаг дискретизации, сек

ДИСКРЕТИЗАЦИЯ ПО УРОВНЮ (квантование)

Квант - промежуток величиной $\Delta X = \dfrac{(X_{\max} - X_{\min})}{N}$

где

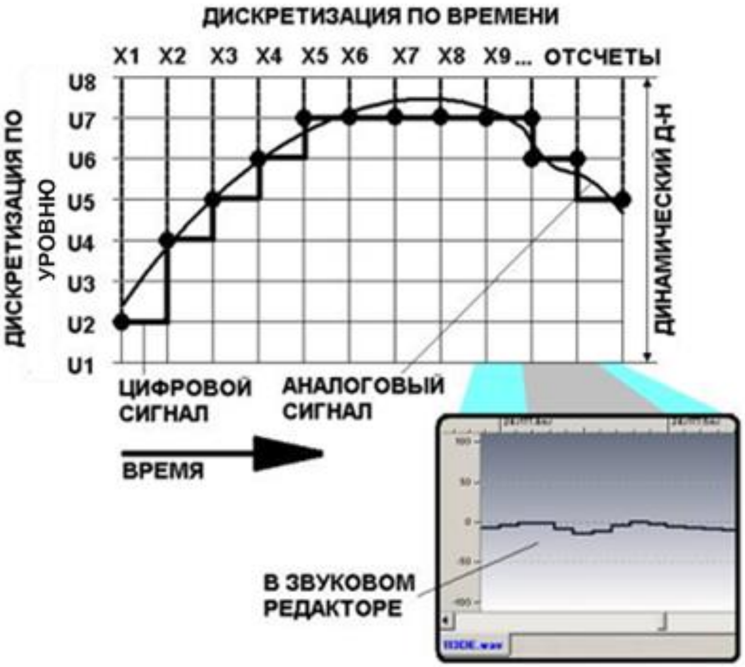
X_{max} – наибольшее значение сигнала

X_{min} - наименьшее значение сигнала

N - число квантов

Значение уровня: $X_i = X_{\min} + \Delta X \cdot (i-1)$

если $X_{i-1} \le X \le X_i$ то $X \rightarrow код\ i$



Высокое качество воспроизведения при:

частоте дискретизации – **44,1 кГц**

квантование – **16 бит**

При таких параметрах

1 сек стереозвука займет

2 байта×44100 байт/с×2 кан×1 сек

= **176 400 байт**

Способы хранения цифрового звука

Способы хранения цифрового звука

- **В прямом виде**
  - Импульсно-кодовая модуляция PCM (Pulse Code Modulation)
  - Адаптивная относительная импульсно-кодовая модуляция ADPCM (Adaptive Delta PCM)
- **В сжатом виде**
  - Кодирование данных без потерь (lossless coding)
  - Кодирование данных с потерями (lossy coding)

ПРИМЕРЫ

Monkey's Audio

Apple Lossless — **ALAC**

Audio Lossless Coding — также известен как **MPEG-4 ALS**

Direct Stream Transfer — **DST**

Meridian Lossless Packing — **MLP**

RealAudio Lossless — **RealPlayer**

True Audio Lossless — **TTA**

Windows Media Lossless — **WMA Lossless**

и др.

ПРИМЕРЫ

MPEG-1 Layer I,II,III (последним является известный **MP3**)

Advanced audio coding (**MPEG-2 AAC**) ,

Ogg Vorbis,

Windows Media Audio (**WMA**),

MPEGPlus,

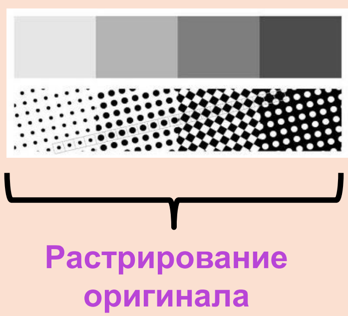
TAC,

и др.

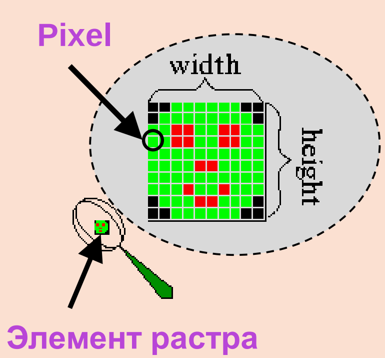
РАСТРОВАЯ ГРАФИКА

Графические объекты формируются в виде множества точек разных цветов и разных яркостей, распределенных по строкам и столбцам (**РАСТРИРОВАНИЕ**)

(если изображение экранное, то точка называется **пиксел**
(от английского **pixel - picture element**)



Частота сетки растра (ЛАНИТУРА) - число линий на дюйм,
lines per inch (lpi)



Виды разрешений:

- разрешение оригинала (**dots per inch – dpi**)
- разрешение экранного изображения
- разрешение печатного изображения

Рекомендации по разрешениям:

- для экранной копии изображения достаточно разрешение 72 dpi
- для распечатки на цветном или лазерном принтере 150-200 dpi
- для вывода на фотоэкспонирующеем устройстве 200-300 dpi

Связь между растрированием и разрешением:

$$N = \left(\frac{dpi}{lpi}\right)^2 + 1; \qquad lpi = \frac{dpi}{\sqrt{N-1}}$$

N - число градаций уровней тона (оттенков)

dpi - разрешение устройства вывода (отображения)

lpi - ланитура растра

Характеристики пикселя растрового изображения:

- ❑размер
- ❑тоновое значение
- ❑глубина (интенсивность) цвета
- ❑позиция

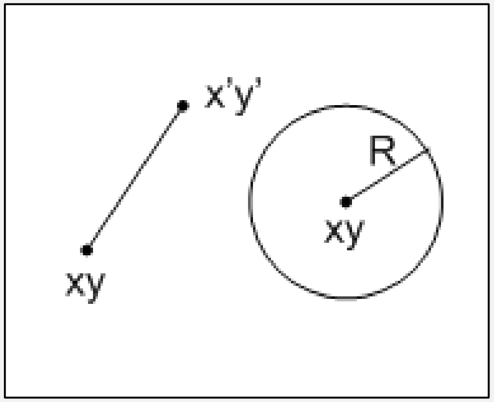
Графический режим вывода изображения на экран определяется **разрешающей способностью экрана** и **глубиной цвета**

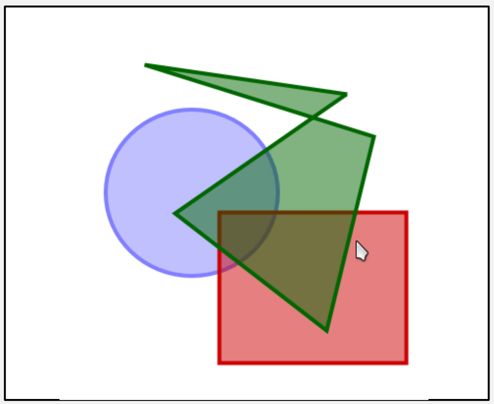
Для персонального компьютера, например:

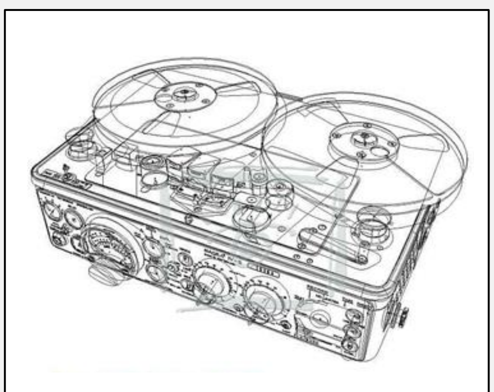
- ➢разрешающая способность экрана **640x480, 800x600, 1024x768, 1280x1024** пикселя;
- ➢глубина цвета – **4, 8, 16, 24** бит на точку

Полная информация о всех точках изображения, хранящаяся в видеопамяти, называется **битовой картой изображения**

ВЕКТОРНАЯ ГРАФИКА







Графический объект представляется в виде отрезков прямых (**векторов**)

Для каждого вектора задается:

- пара точек - концов вектора (или точка, направление вектора и его длина)
- атрибуты - цвет, толщина линии и т.п.

Сложный рисунок

- разбивается на простые фигуры (**объекты**)
- объекты можно редактировать независимо друг от друга
- для каждого объекта в векторном файле хранятся его размеры, кривизна, местоположение в виде числовых коэффициентов

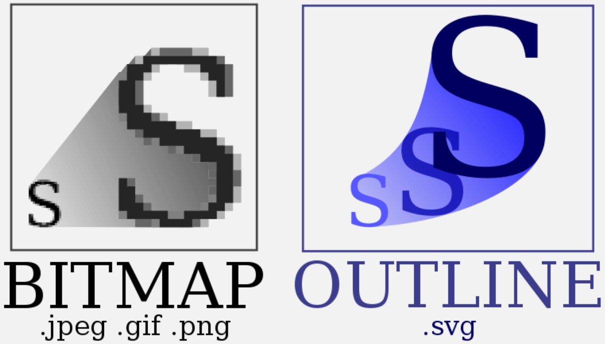
Примеры форматов графических файлов

Векторные:

- **WMF** (англ. Windows MetaFile - метафайл Windows)
- **CGM** (англ. Computer Graphic Metafile - метафайл компьютерной графики)
- **CDR** (англ. CorelDRaw files - файлы CorelDRaw)
- **AI** - формат, поддерживаемый редактором Adobe Illustrator
- **SVG** (от англ. Scalable Vector Graphics — масштабируемая векторная графика)
- и др.

Растровые:

- **PCX**
- **BMP** - сокращение от bitmap, т.е. битовый, растровый
- **JPEG**
- **GIF** (Graphics Interchange File - файл графического обмена)
- **PNG**
- **PCD**
- **PSD**
- **TIF** - Tagged Image File Format
- **EPS**
- **DCS**
- **PDF**
- и др.



Достоинства и недостатки растровой и векторной графики

Векторная графика	Растровая графика
Достоинства: ▪ преобразования без искажений ▪ независимое редактирование частей рисунка ▪ высокая точность прорисовки (до 1 000 000 точек на дюйм) ▪ маленький размер графического файла ***Недостатки:*** ▪условность получаемых изображения **[для высокой степени близости изображения к оригиналу может потребоваться большое количество объектов, что ведет к увеличению объема занимаемой изображением памяти и времени для его отображения]** ▪перевод векторной графики в растр достаточно прост, а обратно сложен **[трассировка растра, при том что требует значительных вычислительных мощностей и времени, не всегда обеспечивает высокое качества векторного рисунка]**	***Достоинства:*** ▪ обеспечивают высокую точность передачи градаций цветов и полутонов ***Недостатки:*** ▪ «пикселизация изображений» при увеличении ▪ зависимость качества изображение от его размеров **Размер файла, хранящего растровое изображение зависит от :** **– размера изображения** **– глубины цвета изображения (чем больше цветов представлено на картинке, тем больше размер файла)** **– разрешения**