



Предмет и основные понятия информатики

Лекция 1

На этой лекции мы рассмотрим:

- что такое информатика, и что мы будем изучать в рамках курса «Информатика»;
- что такое информация, и история развития понятия «информация»;
- **Информатика** -это и наука, и область прикладных исследований, и учебная дисциплина.

Информатика -

это комплексная, техническая наука
о способах

- *получения* информации,
- *накопления,*
- *хранения,*
- *преобразования,*
- *передачи*
- *и защиты* информации с помощью средств
вычислительной техники (**СВТ**)

Информатика -

Она включает дисциплины, относящиеся к обработке информации в *вычислительных машинах* и *вычислительных сетях*:

- абстрактные, вроде анализа алгоритмов, и
- конкретные,
- ❖ разработка *языков программирования*
- ❖ разработка *протоколов передачи данных*.

Информатика

- Термин "**информатика**" происходит от французского слова **Informatique** и образован из двух слов: информация и автоматика.
- Этот термин введен во Франции в середине 60-х лет XX ст.. Тогда в англоязычных странах вошел в употребление термин "**Computer Science**" для обозначения науки о преобразовании информации.
- Теперь эти термины являются синонимами.

Предмет информатики как науки составляют:

- аппаратное обеспечение **СВТ**;
- программное обеспечение **СВТ**;
- средства взаимодействия аппаратного и программного обеспечения;
- средства взаимодействия человека с аппаратными и программными средствами.

Средства взаимодействия

- Средства взаимодействия в информатике принято называть интерфейсом, а средства взаимодействия человека с аппаратными и программными средствами называют **интерфейсом пользователя.**

Данные

- ***Данные*** - составная часть информации, они представляют собой **зарегистрированные сигналы.**

Основные направления информатики для практического применения :

- архитектура вычислительных систем
(приемы и методы построения систем);
- интерфейсы вычислительных систем;
- программирование ;
- преобразование данных ;
- защита информации;

Информация

- **Информация** - это совокупность сведений (данных), которая воспринимается из окружающей среды (входная информация), выдается в окружающую среду (выходная информация) или сохраняется внутри определенной системы.

Информация в жизни человечества

- На первых этапах носителем данных была **память**, и информация от одного человека к другому передавалась устно. Этот способ передачи информации был не надежен.
- По мере развития цивилизации, объемы информации росли, и человеческой памяти стало не хватать; появилась **письменность**.
- Это великое изобретение, было сделано *шумерами* около шести тысяч лет назад.
- Изменился смысл информационных сообщений. Появилась возможность **обобщать, сопоставлять, переосмысливать** ранее сохраненные сведения.

Информация в жизни человечества

- Это дало толчок развитию истории, литературы, точным наукам и в конечном итоге изменило общественную жизнь.
- Изобретение письменности характеризует ***первую информационную революцию.***

Информация в жизни человечества

- Дальнейшее накопление человечеством информации привело к увеличению числа пользователей информации,
 - но письменные труды одного человека могли быть достоянием небольшого окружения.
- Возникшее противоречие было разрешено созданием **печатного станка**.

Эта была **вторая информационная революция** (началась в XVI веке).

- Доступ к информации перестал быть уделом избранных, появилась возможность многократно увеличить объем обмена информацией.

Это привело к масштабным изменениям в *науке, культуре и общественной жизни*.

Информация в жизни человечества

- ***Третья информационная революция*** связывается с **открытием электричества** и появлением (в конце XIX века) на его основе ***новых средств коммуникации:***
 - ❖ телефона,
 - ❖ телеграфа,
 - ❖ радио.
- Возможности накопления информации для тех времен стали *поистине безграничными*, а скорость обмена очень высокой.

Информация в жизни человечества

- К середине XX века, появились быстрые технологические процессы, управлять которыми человек не успевал.
- Проблема управления могла решаться только с помощью **универсальных автоматов**, которые обрабатывают данные и выдают решение в форме **управляющих команд**. Сейчас эти автоматы называются ***компьютерами***.

Информация в жизни человечества

- Наше время отмечается как ***четвертая информационная революция.***
- Пользователями информации стали миллионы людей.
- Появились дешевые компьютеры, доступные миллионам пользователей.

Информация в жизни человечества

- Компьютеры для совместного участия в информационном процессе соединяются в *компьютерные сети*, появилась всемирная компьютерная сеть *Интернет*. Сформировано **единое мировое информационное пространство.**

Информация в жизни человечества

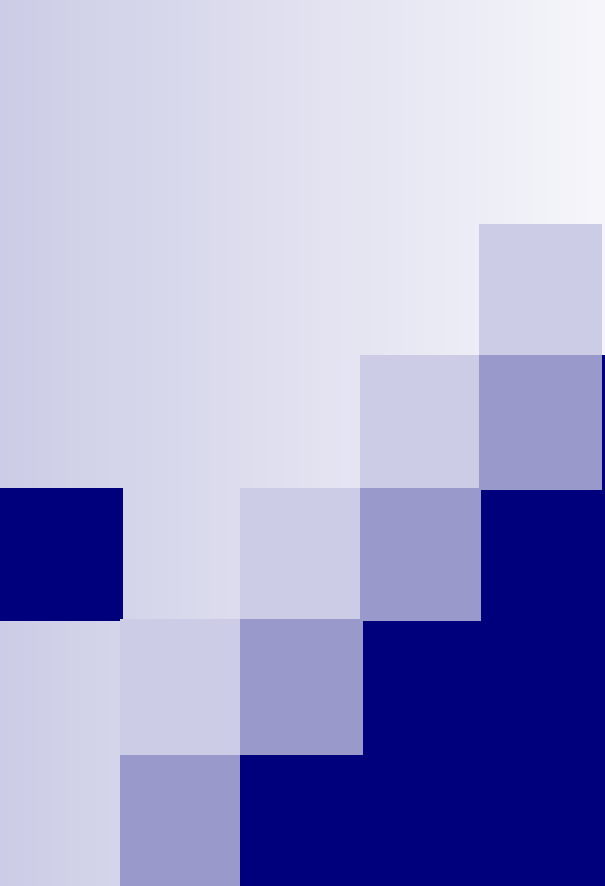
- На наших глазах появляется, так называемое, *информационное общество*, где акцент внимания и значимости смещается с традиционных видов ресурсов (материальные, финансовые, энергетические и пр.) на ***информационный ресурс***, который, хотя всегда существовал, но не рассматривался как экономическая категория.

Информационный ресурс

- *Информационные ресурсы* – это знания, подготовленные людьми для социального использования в обществе и зафиксированные на материальном носителе.
- Информационные ресурсы страны, региона, организации все чаще рассматриваются как **стратегические ресурсы**, аналогичные по значимости запасам сырья, энергии, ископаемых и прочим ресурсам.

Важнейшие свойства информации:

- объективность и субъективность;
- полнота;
- достоверность;
- адекватность;
- доступность;
- актуальность.



Кодирование информации. Системы счисления

Лекция 2

Кодирование информации

- Чтобы была возможность работы с данными различных видов (текст, звук, графика и т.п.), необходимо **унифицировать** форму их представления.
Это можно сделать с помощью *кодирования*.
- Кодированием мы занимаемся довольно часто, например, человек мыслит весьма расплывчатыми понятиями, и, чтобы донести мысль от одного человека к другому, применяется *язык*.
- Язык – это система кодирования понятий.
- Чтобы записать слова языка, применяется, опять же, кодирование – *азбука*.

Кодирование информации

- Для обработки различных видов информации на компьютере требуется *универсальная система кодирования*
- В вычислительной технике используется **двоичное кодирование**.
 - Оно основано на представлении данных последовательностью всего двух знаков: 0 и 1.
 - Эти знаки называются двоичными цифрами, по-английски — ***binary digit*** или сокращенно ***bit*** (бит).

Кодирование информации

- Одним битом могут быть выражены два понятия: 0 или 1 (**да** или **нет**, черное или белое, **истина** или **ложь** и т. п.).
- Если количество битов увеличить до двух, то уже можно выразить **четыре** (2^2) различных понятия:
00 01 10 11
- Тремя битами можно закодировать **восемь** (2^3) различных значений:
000 001 010 011 100 101 110 111
- восемь бит или 1 байт - $2^8 = 256$ и т.д.

Количество информации

- Увеличивая на единицу количество разрядов в системе двоичного кодирования, мы увеличиваем в два раза количество значений, которое можно закодировать.

Общая формула имеет вид:

- $N=2^m$, где N — количество независимых кодируемых значений;
- m — количество бит
- или $m = \text{LOG}_2 N$
- Последняя формула наз. *формулой Шеннона* (частный случай)

Количество информации

- **Бит** — слишком мелкая единица измерения информации. На практике чаще применяется **байт**.
- Именно **один байт** используется для того, чтобы закодировать **символы алфавита**, **клавиши клавиатуры** компьютера.
- Один **байт** является минимальной единицей адресуемой памяти компьютера, т.е. обратиться в память можно к байту, а не биту.
- **1 Байт = 8 бит**

более крупные производные **единицы информации**:

- 1 Килобайт (Кбайт) = 1024 байт = 2^{10} байт,
- 1 Мегабайт (Мбайт) = 1024 Кбайт = 2^{20} байт,
- 1 Гигабайт (Гбайт) = 1024 Мбайт = 2^{30} байт.
- 1 Терабайт (Тбайт) = 1024 Гбайт = 2^{40} байт,

Задачи

- Сколько чисел можно закодировать нулями и единицами в 10 позициях (10 битах)?
- Сколько байт нужно отвести для записи числа 1000?

Решение задач

- 1) В десяти битах можно закодировать $2^{10} = 1024$ чисел.
- 2) В одном байте можно записать числа от 0 до $255 = 2^8 - 1$ (1 байт = 8 бит) .
- Так как $1000 > 255$, следовательно для записи этого числа одного байта мало.
- 2 байта = 2^{16} бит. Следовательно в двух байтах можно записать числа от 0 до $2^{16} - 1 = 65535$.
- Поскольку $65535 > 1000$, то для записи заданного числа нужно отвести 2 байта.

Ячейки памяти ЭВМ

- Элементарная ячейка памяти ЭВМ имеет длину 8 бит (**1 байт**). Каждый байт имеет свой номер (его называют адресом).
- Наибольшую последовательность бит, которую ЭВМ может обрабатывать как единое целое, называют **машинным словом**.
- Длина машинного слова зависит от разрядности процессора и может быть равной 16, 32, 64 битам и т.д.

Система счисления

- **Система счисления** — это способ изображения чисел и соответствующие ему правила действия над числами.
- Если в системах счисления величина знака, которую он обозначает, не зависит от положения этого знака в записи числа, то они называются **непозиционными** системами счисления. Например, римские цифры:

I V X L C D M

- Система счисления, в которой значение каждой цифры **зависит от места** в последовательности цифр в записи числа, называется **позиционной**. Например, общепринятая десятичная система счисления является позиционной.

Позиционные системы

- Количество используемых цифр называется **основанием** позиционной системы счисления. Обозначается основание буквой **P**;
- Общеупотребительной формой записи числа является сокращенная форма записи разложения *по степеням основания системы счисления*

Позиционные системы

- если основание системы счисления равно p , число $N = a_K a_{K-1} \dots a_1 a_0, a_{-1} a_{-2} \dots$ записанное в этой системе, можно представить в виде:

$$N = a_K p^K + a_{K-1} p^{K-1} + \dots + a_1 p^1 + a_0 p^0 + a_{-1} p^{-1} + a_{-2} p^{-2} + \dots$$

- Пользуясь этой формулой можно легко перевести число из системы счисления с основанием p в десятичную.

Пример:

$$26,387 = 2 \cdot 10^1 + 6 \cdot 10^0 + 3 \cdot 10^{-1} + 8 \cdot 10^{-2} + 7 \cdot 10^{-3}.$$

- Здесь **P=10** служит **основанием** системы счисления,

а показатель степени десятки - это

номер позиции цифры в записи числа

(нумерация ведется слева на право, начиная с нуля).

- Пример перевода двоичного числа в десятичную систему

$$101,11_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = 4 + 1 + 1/2 + 1/4 = 5 + 0,5 + 0,25 = 5,75.$$

Позиционные системы

- Для записи чисел в позиционной системе с основанием n нужно иметь **алфавит** из n цифр. Обычно для этого при $n < 10$ используют n первых арабских цифр, а при $n > 10$ к десяти арабским цифрам добавляют буквы.
- Вот примеры алфавитов нескольких систем:

Основание	Основание системы	Алфавит
$n = 2$	двоичная	01
$n = 8$	восьмеричная	01234567
$n = 16$	шестнадцатеричная	0123456789A BCDEF

Позиционные системы

- Основание системы, к которой относится число, обозначается подстрочным индексом к этому числу.
- 101101_2 , 3671_8 , $3B8F_{16}$.

Кодирование целых чисел

- Целые числа могут представляться в компьютере со знаком или без знака.
- Для кодирования целых чисел от 0 до 255 достаточно иметь 8 разрядов двоичного кода (8 бит).
- Шестнадцать бит позволяют закодировать целые числа от 0 до 65 535, а 24 бита — более 16,5 миллионов разных значений.

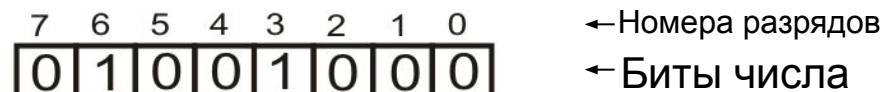
Кодирование целых чисел без знака

- Целые числа без знака обычно занимают в памяти компьютера один, два или 4 байта .
- В однобайтовом формате принимают значения от 00000000_2 до 11111111_2 .
- В двухбайтовом формате –
от $00000000\ 00000000_2$
до $11111111\ 11111111_2$

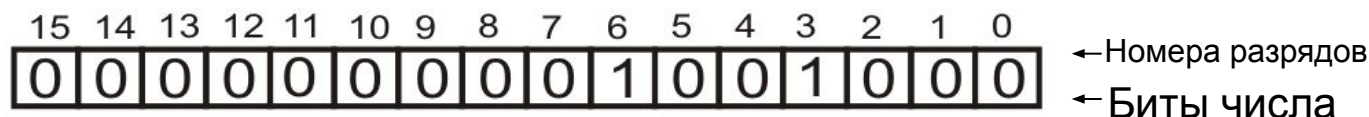
Кодирование целых чисел без знака

■ Примеры:

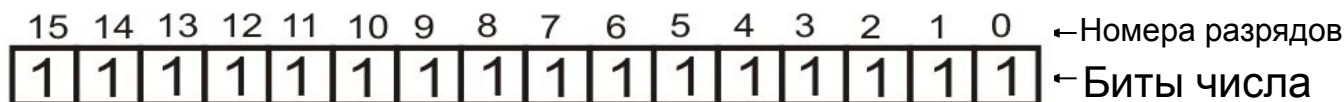
- а) число $72_{10} = 1001000_2$ в однобайтовом формате:



- б) число $72_{10} = 1001000_2$ в двухбайтовом формате:



- в) число 65535 в двухбайтовом формате:



Кодирование целых чисел со знаком

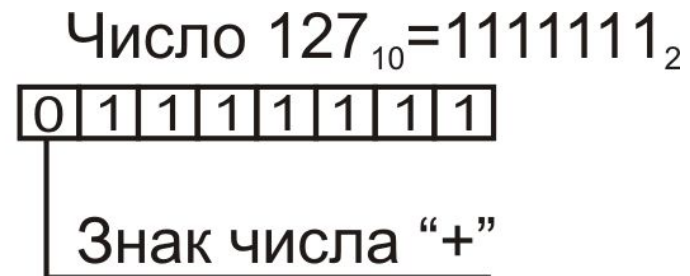
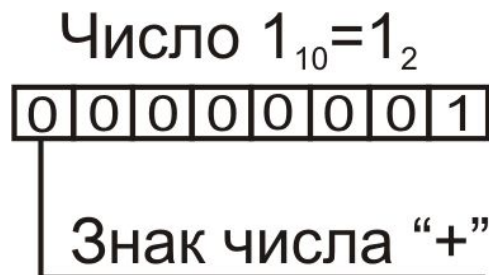
- Целые числа со знаком так же обычно занимают в памяти компьютера один, два или четыре байта.
- Самый левый (старший) разряд содержит информацию о знаке числа (ноль соответствует плюсу, единица - минусу).
- Рассмотрим особенности записи целых чисел со знаком на примере однобайтового формата, при котором для знака отводится один разряд, а для цифр абсолютной величины - семь разрядов.

Кодирование целых чисел со знаком

- В компьютерной технике применяются три формы записи (кодирования) целых чисел со знаком:
 - *прямой код*;
 - *обратный код*;
 - *дополнительный код*.
- Последние две формы применяются особенно широко, так как позволяют упростить конструкцию арифметико-логического устройства компьютера путем замены разнообразных арифметических операций операцией сложения.

Кодирование целых чисел со знаком

- Положительные числа в прямом, коде изображаются двоичными цифрами с цифрой 0 в знаковом разряде.
- Примеры:



Кодирование целых чисел со знаком

- Отрицательные числа в прямом коде имеют следующее изображение:
 - **Прямой код.** В знаковый разряд помещается цифра 1, а в разряды цифровой части числа — двоичный код его абсолютной величины.
 - **Примеры**

Прямой код числа -1_{10}

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

Знак числа “-”

Прямой код числа -127_{10}

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Знак числа “-”

Кодирование целых чисел со знаком

- Положительные числа в прямом, обратном и дополнительном кодах изображаются одинаково:

двоичными цифрами с цифрой 0 в знаковом разряде.

■ Число $1_{10} = 1_2$

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

Знак числа “+”

Число $127_{10} = 1111111_2$

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

Знак числа “+”

Кодирование целых чисел со знаком

- Отрицательные числа в прямом коде имеют следующее изображение:
 - **Прямой код.** В знаковый разряд помещается цифра 1, а в разряды цифровой части числа — двоичный код его абсолютной величины.
 - **Примеры**

Прямой код числа -1_{10}

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

Знак числа “-”

Прямой код числа -127_{10}

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Знак числа “-”

Кодирование целых чисел со знаком

- **Обратный код.** Получается инвертированием всех цифр двоичного кода абсолютной величины числа, включая разряд знака: нули заменяются единицами, а единицы — нулями.
- **Примеры:**

Число: -1_{10}

Код модуля числа: 00000001_2

Обратный код числа: 11111110_2

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

Число: -127_{10}

Код модуля числа: 01111111_2

Обратный код числа: 10000000_2

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

Кодирование целых чисел со знаком

- **Дополнительный код.** Получается образованием обратного кода с последующим прибавлением единицы к его младшему разряду.
- **Примеры:**

Число: -1_{10}

Обратный код числа: 11111110_2

Дополнительный код числа: 11111111_2

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Число: -127_{10}

Обратный код числа: 10000000_2

Дополнительный код числа: 10000001_2

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

Арифметические действия над целыми числами в ЭВМ

- В большинстве компьютеров операция вычитания не используется. Вместо нее производится сложение обратных или дополнительных кодов уменьшаемого и вычитаемого.

Сложение обратных кодов

- *А положительное, В отрицательное и по абсолютной величине больше, чем А.*
- Пример:

Десятичная запись

$$\begin{array}{r} 3 \\ + \\ -10 \\ \hline -7 \end{array}$$

Двоичные коды

$$\begin{array}{r} 0\ 0000011 \\ + \\ 1\ 1110101 \\ \hline 1\ 1111000 \end{array}$$

Обратный код числа -10

Обратный код числа -7

Получен правильный результат в обратном коде.

При переводе в прямой код биты цифровой части результата инвертируются:
 $1\ 0000111 = -7_{10}$.

Кодирование действительных чисел

- Действительные числа в математике представляются конечными или бесконечными дробями, т.е. точность представления чисел не ограничена. Однако в компьютерах числа хранятся ячейках памяти с ограниченным количеством разрядов. Следовательно, бесконечные или очень длинные числа усекаются до некоторой длины и в компьютерном представлении выступают как приближенные. (пример – иррациональное число **ПИ**)

Кодирование действительных чисел

- При написании действительных чисел в программах вместо привычной запятой принято ставить точку.
- Для отображения действительных чисел, которые могут быть как очень маленькими, так и очень большими, используется форма записи чисел с плавающей точкой.
- Например, десятичное число 1.25 в этой форме можно представить так:
 - $1.25 \cdot 10^0 = 0.125 \cdot 10^1 = 0.0125 \cdot 10^2$.

Кодирование действительных чисел

- Любое число N в системе счисления с основанием q можно записать в виде

$$N = M * q^p,$$

- где M — множитель, содержащий все цифры числа (*мантисса*),
- p — целое число, называемое *порядком*.

Действительное число записывается в память компьютера в виде мантиссы и порядка.

Кодирование действительных чисел

■ Примеры:

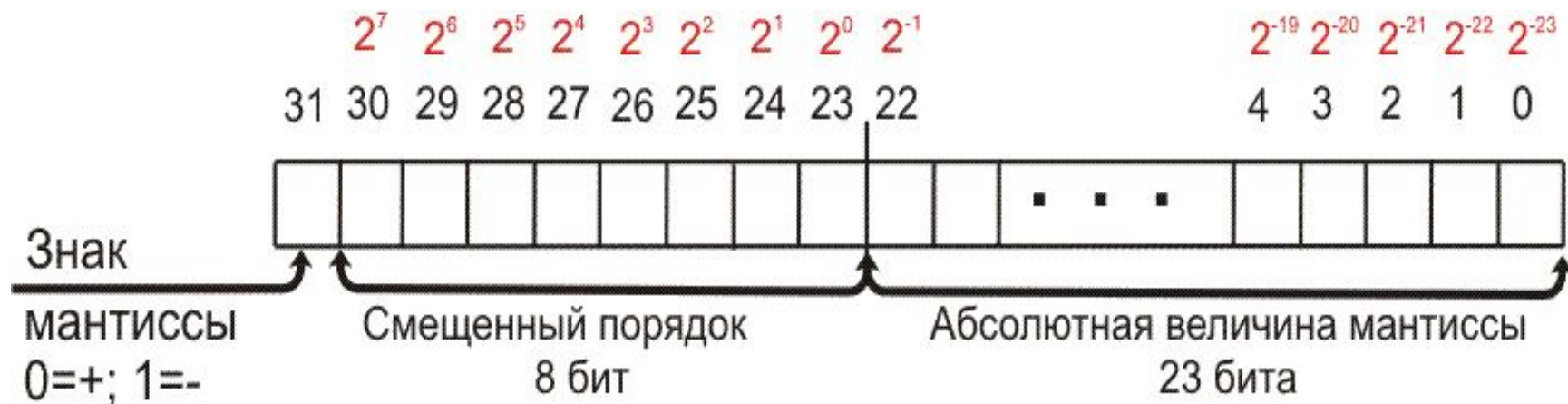
- $0,00012 = 0.12 \cdot 10^{-3};$

M=12; p=-3;

- $1234,5 = 0.12345 \cdot 10^4.$

M=12345; p=4;

Кодирование действительных чисел



Кодирование действительных чисел

- Величина порядка представляется с избытком, т.е. вместо истинного значения порядка хранится число, называемое *смещенным порядком*.
- Для получения *смещенного порядка* необходимо к порядку прибавить смещение. Например, при использовании для хранения порядка восьми бит, (значений от -128 до $+127$) используется смещение 128. Тогда для представления порядка будут использоваться значения от 0 до $+255$, т.е. только неотрицательные числа.

Кодирование текста. Представление СИМВОЛОВ В ДВОИЧНОМ КОДЕ.

- Двоичным кодом наз. *уникальное беззнаковое целое двоичное число*, поставленное в соответствие некоторому символу.
- Для представления *текстовой* информации в компьютере чаще всего используется алфавит мощностью 256 символов.
- Алфавит включает в себя:
 - 1) арабские цифры
 - 2) буквы латинского алфавита
 - 3) знаки препинания и арифметических операций
 - 4) специальные управляющие символы
 - 5) буквы и знаки национального алфавита

Кодирование текста

- Один символ из такого алфавита несет 8 бит информации, следовательно, занимает 1 байт памяти ЭВМ.
- *Все символы такого алфавита пронумерованы от 0 до 255.*
- Каждому номеру соответствует 8-разрядный двоичный код от **00000000** до **11111111**.
- Этот код является *порядковым номером символа* в двоичной системе счисления.

Кодирование текста

- Двоичное кодирование *символьных данных* производится заданием *кодовых таблиц*. Каждому символу ставят в соответствие одно- или двухбайтовый код.
- Помимо этого, *коддовая таблица* ставит в соответствие кодам *клавиши на клавиатуре* и *начертание символа* на экране монитора.

Кодирование текста

- Для разных типов ЭВМ и *операционных систем* используются различные таблицы кодировки.
- Они отличаются порядком размещения символов алфавита в кодовой таблице.
- Международным стандартом на персональных компьютерах является таблица кодировки **ASCII**.
- (American Standard Code of Information Interchange – Американский стандартный код информационного обмена)

Кодирование текста

- Таблица **ASCII** разработана институтом стандартизации США в 1981 г. Ее использовали, в частности, программные продукты, работающие под управлением операционной системы MS-DOS.

Кодирование текста

- В системе *ASCII* закреплены две таблицы кодирования — *базовая* и *расширенная*.
- ❖ Базовая таблица закрепляет значения кодов с номерами:
от 0 до 127
- ❖ Расширенная: относится к символам с номерами:
от 128 до 255.

Кодирование текста

- В базовой области размещаются:
 - ❖ управляющие коды, (например, код клавиши «Enter» 13 или 00001101)
 - ❖ коды символов английского алфавита
 - ❖ знаков препинания,
 - ❖ цифр,
 - ❖ арифметических действий
 - ❖ некоторых вспомогательных символов.
- Остальные 128 кодов, начиная со 128 (двоичный код 10000000) и кончая 255 (11111111), используются для кодировки букв *национальных алфавитов* и научных символов.

Кодирование текста

- Попробуем с помощью таблицы ASCII представить, как будут выглядеть слова в памяти компьютера.

- Слово **file**

01100110

01101001

01101100

01100101

Иногда бывает так, что текст, состоящий из букв русского алфавита, невозможно прочесть - на экране монитора видна какая-то "абракадабра". Это происходит оттого, что на компьютерах применяется разная кодировка символов русского языка.

Кодирование текста

- В последние несколько лет с однобайтовыми кодами для символов появились проблемы: необходимость обеспечить языки с письменностью **иероглифами**, такие как японский и китайский.
- Количество используемых знаков в этих языках даже по минимальным нормам во много раз превышает 256.
- Наиболее распространенной кодировкой символов, использующей по два байта на символ на сегодняшний день является **Unicode**.