

## **Содержание:**

# **ВВЕДЕНИЕ**

Задачи накопления (хранения), обработки и передачи информации стояли перед человечеством на всех этапах его развития. Каждому этапу соответствовал определенный уровень развития средств информационного труда, прогресс развития которых всякий раз придавал человеческому обществу новое качество.

Информатика – комплексная научно-техническая дисциплина, занимающаяся изучением структуры и общих свойств информации, информационных процессов, разработкой на этой основе информационной техники и технологии, а также решением научных и инженерных проблем создания, внедрения и эффективного использования компьютерной техники и технологии во всех сферах общественной практики.

Роль информатики в развитии общества чрезвычайно велика. Она является научным фундаментом процесса информатизации общества. С ней связаны прогрессивное увеличение возможностей компьютерной техники, развитие информационных сетей, создание новых информационных технологий, которые приводят к значительным изменениям во всех сферах общества: в производстве, науке, образовании, медицине и т. д.

Развитие вычислительной техники в современном мире идет очень быстрыми темпами – растет частота и производительность процессоров, увеличиваются объемы памяти, и ускоряется время доступа к ней.

Вычислительная техника постоянно совершенствуется, поэтому алгоритмы сбора, передачи, хранения и сжатия данных должны также постоянно адаптироваться, используя как можно эффективнее возможности современной аппаратуры, такие как многопоточность, технологии вычислений с малой теплоотдачей и многие другие. Таким образом, задача разработки и исследования новых методов передачи, хранения и сжатия

данных является актуальной научной и прикладной задачей.

Для любой операции над информацией она должна быть как-то представлена (записана, зафиксирована). Следовательно, прежде всего, необходимо договориться об определенном способе представления информации (данных), т.е. ввести некоторые обозначения и правила их использования (порядок записи, возможности комбинации знаков и др.).

Когда все это аккуратно определено, используя указанные соглашения, информацию можно записывать, причем с уверенностью, что она будет однозначно воспринята. Вследствие важности данного процесса он имеет специальное название — кодирование информации. Отсюда и актуальность темы кодирование данных.

Целью данной курсовой работы является подробное рассмотрение методов кодирования данных различной природы и подробный анализ исследуемой информации. Исходя из поставленной цели, необходимо решить в работе следующие задачи:

- Разобрать понятия данные, кодирование.
- Изучить технику кодирования в вычислительной технике.
- Рассмотреть разные способы кодирования данных различной природы в вычислительной технике.
- И на основе изученного материала сделать соответствующие выводы и анализ.

## **Глава 1. Общие сведения и понятия в информатике**

### **1.1. Данные в информатике**

Любое взаимодействие материальных объектов имеет энергетическую природу. С точки зрения физики, любые изменения, происходящие во внутренней структуре вещества или в энергетических полях, сопровождаются образованием сигналов. Сигналы обладают способностью распространяться во времени и пространстве. Они затухают в результате взаимодействия с веществом. Сигналы окружают нас на каждом шагу. Как и все объекты материальной природы, сигналы не возникают из ничего и не исчезают бесследно. Их распространение в пространстве всегда завершается взаимодействием с веществом физических тел. Такое взаимодействие в информатике рассматривается как регистрация сигналов. Результат регистрации

сигналов информатика рассматривает как данные. Если сигнал зарегистрирован четко и легко различим на фоне регистрации побочных сигналов, то он может стать источником для получения информации о событиях, которые имели место, или источником информации о предполагаемых событиях (при прогнозировании).

Таким образом, в информатике данные — это зарегистрированные сигналы. Поскольку все сигналы имеют энергетическую, то есть материальную природу, то и данные — тоже объекты материальной природы.

Данные всегда объективны.

Информацию часто отождествляют с данными. Этот подход нашел очень широкое распространение в технических дисциплинах. Например, нам часто встречаются упоминания о том, что «информация передается по компьютерным сетям», «информация обрабатывается компьютерами», «информация хранится в базах данных». Во всех этих случаях происходит подмена понятий. Дело в том, что по компьютерным сетям передаются только данные, компьютеры обрабатывают только данные, а в базах данных хранятся тоже только данные. Станут ли эти данные информацией и если да, то какой, зависит не только от данных, а и от многочисленных аппаратных, программных и естественных методов.

Для иллюстраций (графических данных) контекстным является метод наблюдения, основанный на зрении. В таких случаях мы говорим о графической или визуальной информации.

Для текстовых данных подразумевается контекстный метод чтения, основанный на зрении и на знании языка и азбуки. В таких случаях говорят о текстовой информации.

Для данных, представленных радиоволнами, контекстными являются аппаратные методы преобразования данных и потребления информации с помощью радиоприемника или телевизора. Поэтому мы так часто используем понятия телевизионная информация, информационная программа, информационный выпуск и т. п. Есть свои особенности и у компьютерной информации.

Для данных, представленных в числовой форме и хранящихся в виде сигналов, зарегистрированных на магнитных (и других) носителях или циркулирующих в компьютерных сетях, контекстными являются аппаратные и программные методы вычислительной техники. Их еще называют средствами информационных технологий.

Одна и та же информация (данные) может быть представлена (закодирована) в нескольких формах. С появлением компьютеров возникла необходимость кодирования всех видов информации, с которыми имеет дело отдельный человек и человечество в целом.

## 1.2. Теория кодирования

Решать задачу кодирования информации человечество начало задолго до появления компьютеров.

Грандиозные достижения человечества — письменность и арифметика — есть не что иное, как система кодирования речи и числовой информации. Информация никогда не появляется в чистом виде, она всегда как-то представлена, как-то закодирована. Понятие кодирования используется в информатике необычайно широко, причем существуют даже разные уровни кодирования информации. Например, из практики известна проблема с выбором кодировки русских текстов; это своего рода теоретическая проблема, — какие коды выбрать для каждой буквы.

Теория кодирования информации является одной из дисциплин, которые входят в состав информатики. Она занимается вопросами экономичности (архивация, ускорение передачи данных), надежности (обеспечение восстановления переданной информации в случае повреждения) и безопасности (шифрование) кодирования информации.

Закодированная информация всегда имеет под собой какую-либо объективную основу, поскольку информация есть отражение тех или иных свойств окружающего нас мира. В то же время, одну и ту же информацию можно закодировать разными способами: число записать в десятичной или двоичной системе, данные о выпуске продукции по годам представить в виде таблицы или диаграммы, текст лекции записать на магнитофон или сохранить в печатном виде, собрание сочинений классика перевести и издать на всех языках народов мира.

Так что же такое код и для чего он нужен?

Код - набор условных обозначений для представления информации.

Кодирование - процесс представления информации в виде кода (представление символов одного алфавита символами другого; переход от одной формы

представления информации к другой, более удобной для хранения, передачи или обработки).

Обратное преобразование называется декодированием.

Для общения друг с другом мы используем код - русский язык.

При разговоре этот код передается звуками, при письме - буквами.

Водитель передает сигнал с помощью гудка или миганием фар.

Таким образом, кодирование сводится к использованию совокупности символов по строго определенным правилам.

Способ кодирования зависит от цели, ради которой оно осуществляется:

сокращение записи; засекречивание (шифровка) информации; удобство обработки; и т. д.

### **1.3. Виды кодирования**

В зависимости от целей кодирования, различают следующие его виды:

- кодирование по образцу - используется всякий раз при вводе информации в компьютер для её внутреннего представления;
- - криптографическое кодирование, или шифрование, - используется, когда нужно защитить информацию от несанкционированного доступа;
- - эффективное, или оптимальное, кодирование - используется для устранения избыточности информации, т.е. снижения ее объема, например, в архиваторах;
- - помехозащитное, или помехоустойчивое, кодирование - используется для обеспечения заданной достоверности в случае, когда на сигнал накладывается помеха, например, при передаче информации по каналам связи.

Коды можно классифицировать по различным признакам:

1. По основанию (количеству символов в алфавите):

бинарные (двоичные  $m=2$ ) и не бинарные ( $m \neq 2$ ).

2. По длине кодовых комбинаций (слов):

равномерные - если все кодовые комбинации имеют одинаковую длину;

неравномерные - если длина кодовой комбинации не постоянна.

3. По способу передачи:

последовательные и параллельные;

блочные - данные сначала помещаются в буфер, а потом передаются в канал и бинарные непрерывные.

4. По помехоустойчивости:

простые (примитивные, полные) - для передачи информации используют все возможные кодовые комбинации (без избыточности);

корректирующие (помехозащищенные) - для передачи сообщений используют не все, а только часть (разрешенных) кодовых комбинаций.

5. В зависимости от назначения и применения условно можно выделить следующие типы кодов:

Внутренние коды - это коды, используемые внутри устройств. Это машинные коды, а также коды, базирующиеся на использовании позиционных систем счисления (двоичный, десятичный, двоично-десятичный, восьмеричный, шестнадцатеричный и др.).

Коды для обмена данными и их передачи по каналам связи. Широкое распространение в ПК получил код ASCII (American Standard Code for Information Interchange). ASCII - это 7-битный код буквенно-цифровых и других символов.

Поскольку ЭВМ работают с байтами, то 8-й разряд используется для синхронизации или проверки на четность, или расширения кода.

В ЭВМ фирмы IBM используется расширенный двоично-десятичный код для обмена информацией EBCDIC (Extended Binary Coded Decimal Interchange Code).

В каналах связи широко используется телетайпный код МККТТ (международный консультативный комитет по телефонии и телеграфии) и его модификации (МТК и др.).

При кодировании информации для передачи по каналам связи, в том числе внутри аппаратным трактам, используются коды, обеспечивающие максимальную скорость передачи информации, за счет ее сжатия и устранения избыточности (например: коды Хаффмана и Шеннона-Фано). И коды обеспечивающие достоверность передачи данных, за счет введения избыточности в передаваемые сообщения (например: групповые коды, Хэмминга, циклические и их разновидности).

Коды для специальных применений - это коды, предназначенные для решения специальных задач передачи и обработки данных. Примерами таких кодов является циклический код Грея, который широко используется в АЦП угловых и линейных перемещений. Коды Фибоначчи используются для построения быстродействующих и помехоустойчивых АЦП.

#### ЦЕЛИ КОДИРОВАНИЯ:

- 1) Повышение эффективности передачи данных, за счет достижения максимальной скорости передачи данных.
- 2) Повышение помехоустойчивости при передаче данных.

В соответствии с этими целями теория кодирования развивается в двух основных направлениях:

1. Теория экономичного (эффективного, оптимального) кодирования занимается поиском кодов, позволяющих в каналах без помех повысить эффективность передачи информации за счет устранения избыточности источника и наилучшего согласования скорости передачи данных с пропускной способностью канала связи.
2. Теория помехоустойчивого кодирования занимается поиском кодов, повышающих достоверность передачи информации в каналах с помехами.

В зависимости от применяемых методов кодирования, используют различные математические модели кодов, при этом наиболее часто применяется представление кодов в виде: кодовых матриц; кодовых деревьев; многочленов; геометрических фигур и т.д.

В вычислительной технике существует своя (внутренняя) система кодирования — она называется двоичным кодированием и основана на представлении данных последовательностью всего двух знаков: 0 и 1. Эти знаки называются двоичными цифрами, по-английски — binarydigit, или, сокращенно, bit (бит).

## 1.4. Понятие избыточность информации и необходимость ее сжатия

Характерной особенностью большинства «классических» типов информации, с которыми работают люди, является их избыточность. В русском языке существуют слова, однозначно прочитываемые в случае «потери» некоторых букв. Например, С\_НТ\_БРЬ, МОС\_, Д\_Р\_ВО. Кроме того, имея текст на русском языке с «потерянными» буквами, человек, достаточно хорошо владеющий русским языком, может однозначно восстановить его. Например, вы без труда прочитаете предложение с пропущенными буквами Дм\_т\_ий Ива\_ов\_\_ Менд\_ле\_в - в\_л\_ки\_рус\_кий х\_мик. Однако если это предложение будет читать иностранец, едва знающий русский язык и русскую историю, то он не сможет его понять. Мы, носители русского языка, можем с лёгкостью восстановить окончания, пропущенные буквы в слогах, подобрать подходящие слова.

Для носителя языка обычный связный текст на его родном языке содержит избыточную информацию – её можно удалить, но смысл текста для него сохранится. Одним из примеров проявления избыточности информации и её сжатия является использование математических обозначений.

Например, сумма чисел 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40 записывается так:

$$2+4+6+\dots+37+38+40$$

Эта сумма составлена из всех чётных чисел от 2 до 40. Используя математическую нотацию, эту сумму можно записать намного короче:

Итак, введя новые обозначения, мы сумели сократить запись математического выражения, а значит, сжать информацию. Однако если первая форма записи понятна любому, то вторая понятна только тем, кто знает, как следует интерпретировать эту запись. Пример с суммой позволяет продемонстрировать ещё одну особенность и проблему современной вычислительной техники (и методов кодирования данных) – избыточность информации.

Хранение и передача информации требуют определённой затраты ресурсов. Сжатие данных (перед сохранением или передачей по каналам связи) позволяет

уменьшить эти затраты. На практике такие затраты можно даже выразить в денежном эквиваленте: например, на скачивание из Интернета сжатого музыкального файла потребуется меньше времени, а значит, придётся заплатить меньше денег за пользование Интернетом.

Возникновение различных методов сжатия – степень сжатия входных данных зависит от самих сжимаемых данных.

Сжатие информации является одним из способов ее кодирования. Вообще коды делятся на три большие группы - коды сжатия (эффективные коды), помехоустойчивые коды и криптографические коды.

Коды, предназначенные для сжатия информации, делятся, в свою очередь, на коды без потерь и коды с потерями. Кодирование без потерь подразумевает абсолютно точное восстановление данных после декодирования и может применяться для сжатия любой информации. Кодирование с потерями имеет обычно гораздо более высокую степень сжатия, чем кодирование без потерь, но допускает некоторые отклонения декодированных данных от исходных.

В связи с этим необходимо рассмотреть еще одно важное понятие как избыточность информации.

Кодирование информации является избыточным, если количество бит в полученном коде больше, чем это необходимо для однозначного декодирования исходной информации. Степень избыточности зависит от типа информации: у видеоинформации она в несколько раз больше, чем у графической информации, а степень избыточности последней в несколько раз больше, чем текстовой информации.

Вообще, степень избыточности естественной информации достаточно велика. Клод Шеннон, исследовав избыточность литературного английского языка, установил, что она составляет около 50%. Это означает, что если в английском тексте наугад стереть около половины букв, то по оставшимся буквам человек, знающий английский язык, почти наверняка сможет восстановить текст.

Избыточность языка выполняет очень важную функцию – обеспечивает человеку надёжность её восприятия, особенно в неблагоприятных условиях (просмотр телепередач при наличии помех, чтение тестов в условиях недостаточной освещённости, разговор в вагоне метро и т.п.).

Сегодня, в век информации, несмотря на то, что практически каждому пользователю доступны высокоскоростные каналы для передачи данных и носители больших объемов, вопрос сжатия данных остается актуальным. Существуют ситуации, в которых сжатие данных является просто необходимой операцией. В частности, это касается пересылки данных по электронной почте и размещения информации в интернете.

В основе всех методов сжатия лежит простая идея: если представлять часто используемые элементы короткими кодами, а редко используемые - длинными кодами, то для хранения блока данных требуется меньший объем памяти, чем, если бы все элементы представлялись кодами одинаковой длины. Связь между кодами и вероятностями установлена в классической теореме Шеннона о кодировании источника.

При бурном росте скоростей различных устройств скорость работы каналов связи растет значительно меньшими темпами. Сжатие мультимедийной информации позволяет ощутимо сгладить данный дисбаланс. В данном случае речь идет не только и не столько о персональных компьютерах, ноутбуках и серверах, а и о мобильной телефонии, цифровом телевидении и многих других устройствах с различной вычислительной способностью и различными каналами связи, по которым необходимо быстро и надежно передавать большое количество информации.

Вычислительная техника постоянно совершенствуется, поэтому алгоритмы сжатия данных должны также постоянно адаптироваться, используя как можно эффективнее возможности современной аппаратуры.

Таким образом, задача разработки и исследования новых методов сжатия данных является актуальной научной и прикладной задачей.

Сжатие данных полностью основано на кодирование данных.

## **Глава 2. КОДИРОВАНИЕ В ЭВМ**

### **2.1. Кодирование данных в вычислительной технике**

Существует два принципиально отличных способа представления информации: непрерывный и дискретный.

Если некоторая величина, несущая информацию, в пределах заданного интервала может принимать любое значение, то она называется непрерывной. Наоборот, если величина способна принимать только конечное число значений в пределах интервала, она называется дискретной. Хорошим примером, демонстрирующим различия между непрерывными и дискретными величинами, могут служить целые и вещественные числа. В частности, между значениями 2 и 4 имеется всего одно целое число, но бесконечно много вещественных (включая знаменитое  $\pi$ ).

Для наглядного представления о сути явления дискретности можно также сравнить таблицу значений функции и ее график, полученный путем соединения соответствующих точек плавной линией.

Очевидно, что с увеличением количества значений в таблице (интервал дискретизации сокращается) различия существенно уменьшаются, и дискретизированная величина все лучше описывает исходную (непрерывную).

Наконец, когда имеется настолько большое количество точек, что мы не в состоянии различить соседние, на практике такую величину можно считать непрерывной.

Компьютер способен хранить только дискретно представленную информацию. Его память, как бы велика она ни была, состоит из отдельных битов, а значит, по своей сути дискретна.

Заметим что, сама по себе информация не является непрерывной или дискретной: таковыми являются лишь способы ее представления. Например, давление крови можно с одинаковым успехом измерять аналоговым или цифровым прибором.

Принципиально важным отличием дискретных данных от непрерывных является конечное число их возможных значений. Благодаря этому каждому из них может быть поставлен в соответствие некоторый знак (символ) или, что для компьютерных целей гораздо лучше, определенное число. Иными словами, все значения дискретной величины могут быть тем или иным способом пронумерованы.

Рассмотрим такую, казалось бы, “неарифметическую” величину, как цвет, обычно представляемую в компьютере как совокупность интенсивности трех базовых цветов RGB. Тем не менее, записанные вместе, все три интенсивности образуют

единое “длинное” число, которое формально вполне можно принять за номер цвета.

Значение сформулированного выше положения трудно переоценить: оно позволяет любую дискретную информацию свести к единой универсальной форме — числовой. Не, случайно, поэтому в последнее время большое распространение получил термин “цифровой”, например, цифровой фотоаппарат. Заметим, что для цифрового фотоаппарата важно не столько существование дискретной светочувствительной матрицы из миллионов пикселей (в конце концов “химическая” фотопленка также состояла из отдельных зерен), сколько последующая запись состояния ячеек этой матрицы в числовой форме.

В свете сказанного выше вопрос об универсальности дискретного представления данных становится очевидным: дискретная информация любой природы сводится тем или иным способом к набору чисел. Кстати, данное положение лишний раз подчеркивает, что каким бы “мультимедийным” не выглядел современный компьютер, “в глубине души” он по-прежнему “старая добрая ЭВМ”, т.е. устройство для обработки числовой информации.

Таким образом, проблема кодирования информации для компьютера естественным образом распадается на две составляющие:

кодирование чисел и способ кодирования, который сводит информацию данного вида к числам.

Независимо от того, как изображает исходные данные пользователь (числа или строки символов), они преобразуются в последовательности двоичных цифр 0 и 1.

Двоичная цифра называется битом. Группа из  $n$  бит позволяет

представить  $2^n$  комбинаций от 00...00 до 11...11. Представление данных группой из восьми бит называется байтом. С помощью одного байта можно закодировать  $2^8 = 256$  различных комбинаций.

В информатике байт стал стандартной базовой единицей измерения количества информации. Содержание байта может интерпретироваться по-разному: оно может быть кодированным представлением символа внешнего алфавита, целым или вещественным числом, частью машинной команды и т.д. Интерпретацию байта фактически осуществляет программист в зависимости от контекста своей программы.

Биты в байте нумеруются справа налево, начиная с нуля. Четыре соседних бита в байте называются тетрадой.

Единица данных, кратная байту и равная разрядности регистров процессора ЭВМ, называется машинным словом.

Машинное слово является основной единицей данных, которой оперирует процессор ЭВМ. Длина слова может составлять 1, 2, 4, 8 и более байт.

Для представления больших объемов информации используются производные от байта крупные единицы измерения: килобайт (Кбайт), мегабайт (Мбайт), гигабайт (Гбайт). При этом выполняются следующие соотношения:

1 Кбайт =  $2^{10}$  = 1024 байт; 1 Мбайт =  $2^{20}$  = 1 048 576 байт; 1 Гбайт =  $2^{30}$  = 1 073 741 824 байт.

Процесс обработки информации в ЭВМ сопровождается преобразованием чисел из одной системы счисления в другую.

Система счисления — это способ наименования и представления чисел с помощью символов, имеющих определенные количественные значения. В зависимости от способа представления чисел системы счисления делятся на непозиционные и позиционные.

В информатике применяются только позиционные системы счисления. В каждой позиционной системе счисления используется определенный набор символов, называемых цифрами, последовательная запись которых изображает число. Количество символов в наборе соответствует основанию системы счисления. Позиция символа в изображении числа называется разрядом. Младший разряд целой части числа имеет номер 0. Каждой цифре соответствует число, которое меньше основания системы счисления. В зависимости от позиции цифры в числе ее значение умножается на степень основания, показатель которой равен номеру разряда. В современной информатике используются двоичная, восьмеричная, десятичная и шестнадцатеричная системы счисления.

Двоичная система счисления имеет набор цифр {0,1} и  $p = 2$ .

## **2.2. Двоичная система счисления**

Двоичная система счисления - позиционная система счисления с основанием 2. В этой системе счисления натуральные числа записываются с помощью всего лишь двух символов (в роли которых обычно выступают цифры 0 и 1). В двоичной системе счисления всего две цифры, называемые двоичными. Название разряда двоичного числа – бит. Веса разрядов в двоичной системе изменяются по степеням двойки. Поскольку вес каждого разряда уменьшается либо на 0, либо на 1, то в результате значение числа определяется как сумма соответствующих значений степени двойки.  $10102=1\cdot 2^3+0\cdot 2^2+1\cdot 2^1+0\cdot 2^0=10$ .

Известны следующие преимущества использования двоичной системы в вычислительной технике:

- простота технической реализации;
- надёжность и помехоустойчивость;
- возможность применения аппарата булевой алгебры для выполнения логических операций;
- простота правил двоичной арифметики.

Двоичная система используется в цифровых устройствах, т.к. является наиболее простой и соответствует требованиям:

Чем меньше значений существует в системе, тем проще изготовить отдельные элементы, оперирующие этими значениями.

В частности, две цифры двоичной системы счисления могут быть легко представлены многими физическими явлениями: есть ток — нет тока, индукция магнитного поля больше пороговой величины или нет и т. д.

Чем меньше количество состояний у элемента, тем выше помехоустойчивость и тем быстрее он может работать.

Например, чтобы закодировать три состояния через величину индукции магнитного поля, потребуется ввести два пороговых значения, что не будет способствовать помехоустойчивости и надёжности хранения информации.

Двоичная арифметика является довольно простой. Простыми являются таблицы сложения и умножения - основных действий над числами.

Возможно применение аппарата алгебры логики для выполнения побитовых операций над числами.

# ГЛАВА 3. ПРЕДСТАВЛЕНИЕ ДАННЫХ РАЗЛИЧНОЙ ПРИРОДЫ В ЭВМ

## 3.1. Представление логических данных

Логические данные (переменные) принимают два значения: Истина или Ложь, 1 или 0 — и обозначаются прописными латинскими буквами. Л, В, С, D, ..., X, Y, Z.

В компьютере для логической переменной отводится два байта, или 16 разрядов, которые заполняются 1, если она имеет значение Истина, и 0, если значение Ложь.

С логическими данными выполняются логические операции, среди которых наибольшее применение получили операции конъюнкции, дизъюнкции, отрицания и импликации. Результатом логической операции является также значение Истина или Ложь, 1 или 0.

Значение логической функции F можно определить с помощью ее таблицы истинности, которая показывает, какие значения принимает логическая функция при всех наборах ее аргументов, например A и B.

## 3.2. Кодирование текстовых данных

Главная идея кодирования текстовой информации в компьютере заключается в следующем. Текст состоит из отдельных символов, поэтому необходимо установить правила кодирования именно для символов. Общепринятый подход заключается в том, что определяется стандартная последовательность символов, и они нумеруются. В результате описанного подхода любой текстовый символ сохраняется в виде числа, которое является не чем иным, как его номером в общем стандартном списке символов (своеобразном алфавите). Кодирование символов с помощью номера в стандартной последовательности или, как часто говорят, в кодовой таблице очень компактно и удобно для компьютерной реализации.

Если каждому символу алфавита сопоставить определенное целое число (например, порядковый номер), то с помощью двоичного кода можно легко кодировать текстовую информацию.

Восьми двоичных разрядов достаточно для кодирования 256 различных символов. Этого хватит, чтобы выразить различными комбинациями восьми битов все символы английского и русского алфавитов, как строчные, так и прописные, а также знаки препинания, символы основных арифметических действий и некоторые общепринятые специальные символы, например символ «§». Технически это выглядит очень просто, однако всегда существовали достаточно веские организационные сложности.

В первые годы развития вычислительной техники они были связаны с отсутствием необходимых стандартов, а в настоящее время вызваны, наоборот, избытком одновременно действующих и противоречивых стандартов. Для того чтобы весь мир одинаково кодировал текстовые данные, нужны единые таблицы кодирования, а это пока невозможно из-за противоречий между символами национальных алфавитов, а также противоречий корпоративного характера.

Институт стандартизации США (ANSI — American National Standard Institute) ввел в действие систему кодирования ASCII (American Standard Code for Information Interchange — стандартный код информационного обмена США).

В системе ASCII закреплены две таблицы кодирования: базовая и расширенная. Базовая таблица закрепляет значения кодов от 0 до 127, а расширенная относится к символам с номерами от 128 до 255. Первые 32 кода базовой таблицы, начиная с нулевого, отданы производителям аппаратных средств (в первую очередь производителям компьютеров и печатающих устройств). В этой области размещаются так называемые управляющие коды, которым не соответствуют никакие символы языков, и, соответственно, эти коды не выводятся ни на экран, ни на устройства печати, но ими можно управлять тем, как производится вывод прочих данных. Начиная, с кода 32 по код 127, размещены коды символов английского алфавита, знаков препинания, цифр, арифметических действий и некоторых вспомогательных символов.

Базовая таблица кодировки ascii

37 % 52 4 67 C 82 R 97 a 112 p

7 beep (звуковой сигнал)

38 & 53 5 68 D 83 S 98 b 113 q

39 ' 54 6 69 E 84 T 99 c 114 r

8 backspace (удаление предыдущего символа) 40 ( 55 7 70 F 85 U 100 d 115 s

41 ) 56 8 71 G 86 V 101 e 116 t

9 tab (табуляция)

42 \* 57 9 72 H 87 W 102 f 117 u

43 + 58 : 73 I 88 X 103 g 118 v

10 linefeed (перевод строки)

44 , 59 ; 74 J 89 Y 104 h 119 w

45 - 60 < 75 K 90 Z 105 i 120 x

13 carriage return (возврат каретки)

46 . 61 = 76 L 91 [ 106 j 121 y

32 space (пробел)

47 / 62 > 77 M 92 \ 107 k 122 z

33 !

48 0 63 ? 78 N 93 ] 108 l 123 {

34 «

49 1 64 @ 79 O 94 ^ 109 m 124 |

35 #

50 2 65 A 80 P 95 \_ 110 n 125 }

36 \$

51 3 66 B 81 Q 96 ` 111 o 126 ~

Аналогичные системы кодирования текстовых данных были разработаны и в других странах. Так, например, в СССР в этой области действовала система кодирования КОИ-7 (код обмена информацией, семизначный). Однако поддержка производителей оборудования и программ вывела американский код ASCII на уровень международного стандарта, и национальным системам кодирования

пришлось «отступить» во вторую, расширенную часть системы кодирования, определяющую значения кодов со 128 по 255.

Отсутствие единого стандарта в этой области привело к множественности одновременно действующих кодировок. Только в России можно указать три действующих стандарта кодировки и еще два устаревших. Так, например, кодировка символов русского языка, известная как кодировка Windows-1251, была введена «извне» — компанией Microsoft, но, учитывая широкое распространение операционных систем и других продуктов этой компании в России, она глубоко закрепились и нашла широкое распространение.

Кодировка Windows-1251 (синоним CP1251)

### Кодировка Windows-1251 (синоним CP1251)

**.0 .1 .2 .3 .4 .5 .6 .7 .8 .9 .A .B .C .D .E .F**

**8.** Ъ Ѓ , Є „ … † ‡ € ‰ Љ ‹ Њ Ѓ Ѕ Ї  
402 403 201A 453 201E 2026 2020 2021 20AC 2030 409 2039 40A 40C 40B 40F

**9.** ъ ‘ ’ “ ” • – — ™ ъ ‹ ъ ѓ х ц  
452 2018 2019 201C 201D 2022 2013 2014 2122 459 203A 45A 45C 45B 45F

**A.** Ў ў Ј ѡ Ѓ † § Ђ © € « ¬ ® Ї  
A0 40E 45E 408 A4 490 A6 A7 401 A9 404 AB AC AD AE 407

**B.** ° ± І і ґ μ ¶ · ё № € » ј S s ѳ  
B0 B1 406 456 491 B5 B6 B7 451 2116 454 BB 458 405 455 457

**C.** А Б В Г Д Е Ж З И Й К Л М Н О П  
410 411 412 413 414 415 416 417 418 419 41A 41B 41C 41D 41E 41F

**D.** Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я  
420 421 422 423 424 425 426 427 428 429 42A 42B 42C 42D 42E 42F

**E.** а б в г д е ж з и й к л м н о п  
430 431 432 433 434 435 436 437 438 439 43A 43B 43C 43D 43E 43F

**F.** р с т у ф х ц ч ш щ ъ ы ь э ю я  
440 441 442 443 444 445 446 447 448 449 44A 44B 44C 44D 44E 44F

Эта кодировка используется на большинстве локальных компьютеров, работающих на платформе Windows. Де-факто она стала стандартной в российском секторе World Wide Web.

Другая распространенная кодировка носит название КОИ-8 (код обмена информацией, восьмизначный) — ее происхождение относится ко временам действия Совета Экономической Взаимопомощи государств Восточной Европы.

Кодировка KOI8-R (русская).

Кодировка KOI8-R (русская).

**.0 .1 .2 .3 .4 .5 .6 .7 .8 .9 .A .B .C .D .E .F**

**8.** — | Г ґ Լ Ԑ ƒ † ‡ ⊥ † ■ ■ ■ | |  
2500 2502 250C 2510 2514 2518 251C 2524 252C 2534 253C 2580 2584 2588 258C 2590

9.    |  • √ ≈ ≤ ≥ J ° ² · ÷  
 2591 2592 2593 2320 25A0 2219 221A 2248 2264 2265 A0 2321 B0 B2 B7 F7

A. = || Ғ ё п ғ җ п җ ҫ Ӣ Ӥ Ӧ ӧ Ө ө Ӫ  
 2550 2551 2552 451 2553 2554 2555 2556 2557 2558 2559 255A 255B 255C 255D 255E

B. † ‡ † Ё † † † † † † † † † † † † ©  
 255F 2560 2561 401 2562 2563 2564 2565 2566 2567 2568 2569 256A 256B 256C A9

C. ю а б ц д е ф г х и й к л м н о  
 44E 430 431 446 434 435 444 433 445 438 439 43A 43B 43C 43D 43E

D. п я р с т у ж в ь ы з ш э щ ч ь  
 43F 44F 440 441 442 443 436 432 44C 44B 437 448 44D 449 447 44A

E. Ю А Б Ц Д Е Ф Г Х И Й К Л М Н О  
 42E 410 411 426 414 415 424 413 425 418 419 41A 41B 41C 41D 41E

F. П Я Р С Т У Ж В Ъ Ы З Ш Э Щ Ч Ъ  
 41F 42F 420 421 422 423 416 412 42C 42B 417 428 42D 429 427 42A

На базе этой кодировки ныне действуют кодировки КОИ8-Р (русская) и КОИ8-У (украинская). Сегодня кодировка КОИ8-Р имеет широкое распространение в компьютерных сетях на территории России и в некоторых службах российского сектора Интернета. В частности, в России она де-факто является стандартной в сообщениях электронной почты и телеконференций. Международный стандарт, в

котором предусмотрена кодировка символов русского алфавита, носит название кодировки /50 (International Standard Organization) международный институт стандартизации. На практике данная кодировка используется редко.

На компьютерах, работающих в операционных системах MS-DOS, могут действовать еще две кодировки (кодировка ГОСТ и кодировка ГОСТ-альтернативная). Первая из них считалась устаревшей даже в первые годы появления персональной вычислительной техники, но вторая используется, и по сей день.

Таблица символов Альтернативной кодировки ГОСТ, коды 128—255, коды 0—127 соответствуют ASCII.

Таблица символов **Альтернативной кодировки ГОСТ**, коды 128—255, коды 0—127 соотв

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A
<b>80h</b>	А	Б	В	Г	Д	Е	Ж	З	И	Й	К
	U+0410	U+0411	U+0412	U+0413	U+0414	U+0415	U+0416	U+0417	U+0418	U+0419	U+
<b>90h</b>	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ
	U+0420	U+0421	U+0422	U+0423	U+0424	U+0425	U+0426	U+0427	U+0428	U+0429	U+
<b>A0h</b>	а	б	в	г	д	е	ж	з	и	й	к
	U+0430	U+0431	U+0432	U+0433	U+0434	U+0435	U+0436	U+0437	U+0438	U+0439	U+
<b>B0h</b>	⋮	⋮	⋮		†	‡	‡	π	¶	‡	
	U+2591	U+2592	U+2593	U+2502	U+2524	U+2561	U+2562	U+2556	U+2555	U+2563	U+

**C0h**

Л	л	Т	т	—	†	‡		ℓ	ѓ	Ⓕ
U+2514	U+2534	U+252C	U+251C	U+2500	U+253C	U+255E	U+255F	U+255A	U+2554	U+

**D0h**

Ⓕ	ѳ	π	ℓ	ℓ	ƒ	ѓ		‡	Ј	ѓ
U+2568	U+2564	U+2565	U+2559	U+2558	U+2552	U+2553	U+256B	U+256A	U+2518	U+

**E0h**

р	с	т	у	ф	х	ц	ч	ш	щ	ъ
U+0440	U+0441	U+0442	U+0443	U+0444	U+0445	U+0446	U+0447	U+0448	U+0449	U+

**F0h**

Ё	ё	г	г	Ј	Л	→	←	↓	↑	÷
U+0401	U+0451	U+256D	U+256E	U+256F	U+2570	U+2192	U+2190	U+2193	U+2191	U+

Для работы в среде операционной системы MS-DOS используется «альтернативная» кодировка, в терминологии фирмы Microsoft - CP866 (КОИ-7).

Фирма Apple разработала для компьютеров Macintosh свою собственную кодировку русских букв (Mac).

Международная организация по стандартизации (International Standards Organization, ISO) утвердила в качестве стандарта для русского языка еще одну кодировку под названием ISO 8859-5.

Одному и тому же двоичному коду в разных кодовых таблицах ставится в соответствие различные символы.

Двоичный код Десятичный код КОИ8 CP1251 CP866 Mac ISO

11000010	194	б	В	-	-	Т
----------	-----	---	---	---	---	---

К счастью, в большинстве случаев пользователь не должен заботиться о перекодировках текстовых документов. При работе в приложениях Windows предусмотрена возможность автоматической перекодировки документов,

созданных в приложениях MS-DOS.

При работе в Интернет с использованием браузеров Internet Explorer и Netscape Communicator происходит автоматическая перекодировка Web-страниц.

### **3.3. Единая система кодирования текстовых данных**

Если проанализировать организационные трудности, связанные с созданием единой системы кодирования текстовых данных, то можно прийти к выводу, что они вызваны ограниченным набором кодов (256).

В то же время очевидно, что если, например, кодировать символы не восьмиразрядными двоичными числами, а числами с большим количеством разрядов, то и диапазон возможных значений кодов станет намного больше. Такая система, основанная на 16-разрядном кодировании символов, получила название универсальной —UNICODE. Шестнадцать разрядов позволяют обеспечить уникальные коды для 65536 различных символов — этого поля достаточно для размещения в одной таблице символов большинства языков планеты.

Несмотря на тривиальную очевидность такого подхода, простой механический переход на данную систему долгое время сдерживался из-за недостаточных ресурсов средств вычислительной техники (в системе кодирования UNICODE все текстовые документы автоматически становятся вдвое длиннее).

Во второй половине 90-х годов технические средства достигли необходимого уровня обеспеченности ресурсами, и сегодня мы наблюдаем постепенный перевод документов и программных средств на универсальную систему кодирования.

Тысячи знаков языковой группы юго-восточной Азии никак невозможно было описать в одном байте информации, который выделялся для кодирования символов в расширенных версиях ASCII. В результате был создан консорциум под названием Юникод (Unicode — Unicode Consortium) при сотрудничестве многих лидеров IT индустрии (те, кто производит софт, кто кодирует железо, кто создает шрифты), которые были заинтересованы в появлении универсальной кодировки текста.

Первой вариацией, вышедшей под эгидой консорциума Юникод, была UTF 32.

Цифра в названии кодировки означает количество бит, которое используется для кодирования одного символа. 32 бита составляют 4 байта информации, которые понадобятся для кодирования одного единственного знака в новой универсальной кодировке UTF.

В результате чего, один и тот же файл с текстом, закодированный в расширенной версии ASCII и в UTF-32, в последнем случае будет иметь размер (весить) в четыре раза больше. Это плохо, но зато теперь у нас появилась возможность закодировать с помощью ЮТФ число знаков, равное двум в тридцать второй степени (миллиарды символов, которые покроют любое реально необходимое значение с колоссальным запасом).

Но многим странам с языками европейской группы такое огромное количество знаков использовать в кодировке вовсе и не было необходимости, однако при задействовании UTF-32 они ни за что ни про что получали четырехкратное увеличение веса текстовых документов, а в результате и увеличение объема интернет трафика и объема хранимых данных. Это много, и такое расточительство себе никто не мог позволить.

В результате развития Юникода появилась UTF-16, которая получилась настолько удачной, что была принята по умолчанию как базовое пространство для всех символов, которые у нас используются. Она использует два байта для кодирования одного знака. Сколько символов можно закодировать в UTF-16 с помощью 16 бит? 65 536 (два в степени шестнадцать), и именно это число было принято за базовое пространство в Юникоде. Помимо этого существуют способы закодировать с помощью нее и около двух миллионов знаков, но ограничились расширенным пространством в миллион символов текста.

Но даже эта удачная версия кодировки Юникода не принесла особого удовлетворения тем, кто писал, допустим, программы только на английском языке, ибо у них, после перехода от расширенной версии ASCII к UTF-16, вес документов увеличивался в два раза (один байт на один символ в Аски и два байта на тот же самый символ в ЮТФ-16).

Вот именно для удовлетворения всех и вся в консорциуме Unicode было решено придумать кодировку переменной длины. Ее назвали UTF-8. Несмотря на восьмерку в названии, она действительно имеет переменную длину, т.е. каждый символ текста может быть закодирован в последовательность длиной от одного до шести байт.

На практике же в UTF-8 используется только диапазон от одного до четырех байт, потому что за четырьмя байтами кода ничего уже даже теоретически невозможно представить. Все латинские знаки в ней кодируются в один байт, так же как и в старой доброй ASCII.

Что примечательно, в случае кодирования только латиницы, даже те программы, которые не понимают Юникод, все равно прочитают то, что закодировано в UTF-8. Т.е. базовая часть Аски просто перешла в это детище консорциума Unicode.

Кириллические же знаки в UTF-8 кодируются в два байта, а, например, грузинские — в три байта. Консорциум Юникод после создания UTF 16 и 8 решил основную проблему — теперь у нас в шрифтах существует единое кодовое пространство. И теперь их производителям остается только, исходя из своих сил и возможностей, заполнять его векторными формами символов текста.

## 3.4. Представление и обработка числовой информации в компьютере

Целое число без знака располагается в регистре (слове, полуслове или двойном слове) так, что его самый младший двоичный разряд записывается в крайний правый бит разрядной сетки, причем все разряды должны быть обязательно заполнены, даже если в этом разряде будет храниться “незначащий ноль”. Например, десятичное число  $19_{10} = 10011_2$  в 16-разрядном представлении (полуслове) запишется так:

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

**0 0 0 0 0 0 0 0 0 0 1 0 0 1 1**

При такой форме представления целых чисел без знака диапазон их возможных значений находится в пределах от **0** до  **$2^n - 1$** , где  **$n$**  – число разрядов в регистре (разрядной сетке).

В таблице приведены максимальные значения десятичных чисел без знака и соответствующее им число разрядов (бит):

Число разрядов.	Максимальное двоичное число без знака.	Максимальное десятичное число без знака.
8	11111111	$2^8 - 1 = 255$
16	1111111111111111	$2^{16} - 1 = 65535$
32	11111111111111111111111111111111	$2^{32} - 1 = 4294967295$

Для представления целых чисел со знаком один разряд, как правило, старший отводится под знак числа. Знак положительного числа кодируется нулем, а знак отрицательного - единицей в этом разряде.

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

**0 0 0 0 0 0 0000010011**

Знак числа «+»

Такая форма представления целых чисел со знаком, когда крайний левый бит разрядной сетки отводится под знак числа, а остальные  $n-1$  бит отводятся под цифры числа в двоичной системе счисления, называется прямым кодом двоичного числа.

Прямой код положительного числа фактически совпадает с самим числом, а прямой код отрицательного числа отличается от положительного только наличием единицы в знаковом разряде.

Естественно, что выделение одного разряда под знак числа приводит к уменьшению имеющихся в нашем распоряжении разрядов регистра на единицу. Поэтому, максимальное значение числа, которое можно представить в  $n$ -разрядном регистре, также уменьшится. Теперь оно будет равно  $2^{n-1}-1$ .

Форма представления двоичных чисел в виде прямого кода используется в компьютере только для представления целых положительных чисел.

Сложение положительных чисел в компьютерной арифметике осуществляется над прямыми кодами двоичных чисел. Для реализации операции вычитания, используется специальная форма представления отрицательных чисел, называемая дополнительным кодом, что позволяет заменить операцию вычитания простым сложением. При этом операция сложения выполняется над всеми разрядами полученного дополнительного кода, т.е. распространяется и на разряды знаков, рассматриваемых в данном случае как разряды целой части числа.

Дополнительный код отрицательного двоичного числа, получается, по следующему правилу:

1. отрицательное двоичное число записывается в прямом коде;
2. все двоичные разряды, кроме знакового, инвертируются (единицы заменяются нулями, а нули – единицами);
3. к инвертированному числу прибавляется единица по правилам сложения двоичных чисел.

Таким образом, при алгебраическом сложении (алгебраическое сложение – это сложение чисел со знаками) двух двоичных чисел с использованием дополнительного кода, положительные слагаемые представляются в компьютере в прямом коде, а отрицательные – в дополнительном. Затем производится суммирование этих кодов, включая разряды знаков. При возникновении переноса из знакового разряда единица переноса отбрасывается, т.к. она вышла за пределы разрядной сетки. В результате получается алгебраическая сумма в прямом коде, если она положительна, и в дополнительном коде, если эта сумма получилась отрицательной.

При решении конкретных физических, математических и других задач фигурируют как очень малые, так и очень большие числа.

Диапазон изменения величин может при этом составлять от  $10^{-30}$  до  $10^{+30}$ .

Представление чисел в естественной форме в виде записи целой и дробной части числа, отделенных друг от друга запятой, называется формой представления чисел с фиксированной запятой. Ясно, что занимать разряды для хранения нулей в конце числа или в начале после запятой крайне неэффективно. Поэтому для представления вещественных чисел в современных компьютерах применяется

другая форма, которую принято называть форма представления вещественных чисел с плавающей запятой.

В этом формате разряды регистра разбиваются на два поля, имеющие название мантисса и порядок. К тому же используется нормализованная форма записи чисел, при которой у мантиссы старший разряд обязательно меньше 1, а порядок – целое положительное или отрицательное число:

$$A = \pm m \cdot K^{\pm p},$$

где  $m$  – мантисса числа,  $p$  – порядок числа ( $|m| < 1$ ),  $K$  – основание системы счисления.

При записи числа с плавающей точкой выделяются разряды для хранения знака мантиссы, порядка и мантиссы, например (для простоты взято 4-х байтовое представление):

31 30 29 28 27 26 25 24 23 22 ... 6 5 4 3 2 1 0

Зн. м Смещенный порядок Мантисса без старшей единицы

Порядок числа запоминается увеличенным на  $2^{k-1} - 1$ , где  $k$  – количество разрядов, отведенное для кодирования порядка. Такой порядок называется смещенным:

$$p_M = p + (2^{k-1} - 1), \text{ где } p - \text{порядок числа, } p_M - \text{машинный порядок.}$$

Для 4-х байтового представления машинный порядок будет вычисляться по формуле:  $p_M = p + (2^7 - 1)$  или  $p_M = p + 127$ .

При таком способе представления отпадает необходимость кодирования знака порядка.

Т.к. при нормализованной форме записи вещественного числа в двоичной системе счисления первая цифра мантиссы у любого числа всегда 1, то отпадает необходимость хранить эту единицу. Поэтому мантисса запоминается без первой единицы. Она считается скрытым разрядом. Таким образом, при хранении мантиссы этой единицы нет, однако при аппаратном выполнении операций она, естественно учитывается.

Всякое десятичное число, прежде чем оно попадает в память компьютера, преобразуется по схеме:

1.  $A_{10} \Rightarrow m \cdot 10^P$ , мантисса и порядок числа при этом записываются в двоичной системе счисления.
2. В полученном нормализованном двоичном числе мантисса и порядок заменяется прямым кодом, если числа положительны и дополнительным кодом, если они отрицательны.
3. Полученные коды порядка и мантиссы помещаются в ячейки памяти компьютера.

Это преобразование осуществляется автоматически при вводе данных в компьютер. В процессе выполнения вычислений коды мантисс и порядка конкретных чисел извлекаются из ячеек памяти и направляются в арифметическое устройство, где над кодами по специальным правилам двоичной арифметики выполняются действия.

## 3.5. Кодирование звуковых данных

С начала 90-х годов персональные компьютеры получили возможность работать со звуком. Каждый компьютер, имеющий звуковую плату (аудиоадаптер), микрофон и колонки, может записывать, сохранять и воспроизводить звуковые данные с помощью специальных программных средств (редакторов аудиофайлов).

Метод FM (Frequency Modulation).

Звуковой сигнал – это непрерывная волна с изменяющейся амплитудой и частотой. Чем больше амплитуда сигнала, тем он громче для человека, чем больше частота сигнала, тем выше тон.

В процессе кодирования фонограммы, непрерывная звуковая волна разбивается по времени на отдельные маленькие временные участки (элементарные звуки). Для каждого участка устанавливается определенная величина амплитуды, каждому значению амплитуды присваивается двоичный код. Качество кодирования зависит от количества измерений уровня сигнала в единицу времени.

Аудиоадаптер (звуковая плата) – специальное устройство, подключаемое к компьютеру, предназначенное для преобразования электрических колебаний звуковой частоты в числовой двоичный код при вводе звука и для обратного

преобразования (из числового кода в электрические колебания) при воспроизведении звука.

В процессе записи звука аудиоадаптер с определенным периодом измеряет амплитуду электрического тока и заносит в регистр двоичный код полученной величины. Затем полученный код из регистра переписывается в оперативную память компьютера. Качество компьютерного звука определяется характеристиками аудиоадаптера: частотой дискретизации и разрядностью.

Частота дискретизации – это количество измерений входного сигнала за 1 секунду. Частота измеряется в герцах (Гц). Одно измерение за одну секунду соответствует частоте 1 Гц. 1000 измерений за 1 секунду – 1 кГц.

Разрядность регистра – число бит в регистре аудиоадаптера. Разрядность определяет точность измерения входного сигнала. Чем больше разрядность, тем меньше погрешность каждого отдельного преобразования величины электрического сигнала в число и обратно. Если разрядность равна 8 (16), то при измерении входного сигнала может быть получено  $2^8 = 256$  ( $2^{16} = 65536$ ) различных значений.

При таких преобразованиях неизбежны потери информации, связанные с кодированием, вместе с тем этот метод обеспечивает весьма компактный код.

Метод таблично-волнового синтеза (Wave-Table) лучше соответствует современному уровню развития техники. В заранее подготовленных таблицах хранятся образцы звуков для множества различных музыкальных инструментов. Такие образцы называются сэмплами. Числовые коды выражают тип инструмента, номер его модели, высоту тона, продолжительность и интенсивность звука, динамику его изменения, некоторые параметры среды, в которой происходит звучание, а также и другие параметры, характеризующие особенности звучания.

Так как в качестве образцов используются реальные звуки, качество звука получается высоким близким к звучанию реальных музыкальных инструментов.

## **3.6. Кодирование графической информации**

Растр — это метод кодирования графической информации, издавна принятый в полиграфии.

Если рассмотреть с помощью увеличительного стекла черно-белое графическое изображение, напечатанное в газете или книге, то можно увидеть, что оно состоит из мельчайших точек, образующих характерный узор, называемый растром.

Поскольку линейные координаты и индивидуальные свойства каждой точки (яркость) можно выразить с помощью целых чисел, то можно сказать, что растровое кодирование позволяет использовать двоичный код для представления графических данных. Общепринятым на сегодняшний день считается представление черно-белых иллюстраций в виде комбинации точек с 256 градациями серого цвета, и, таким образом, для кодирования яркости любой точки обычно достаточно восьмиразрядного двоичного числа.

Для кодирования цветных графических изображений применяется принцип декомпозиции произвольного цвета на основные составляющие. В качестве таких составляющих используют три основных цвета: красный (Red, R), зеленый (Green, G) и синий (Blue, B).

На практике считается (хотя теоретически это не совсем так), что любой цвет, видимый человеческим глазом, можно получить путем механического смешения этих трех основных цветов. Такая система кодирования называется системой RGB по первым буквам названий основных цветов.

Если для кодирования яркости каждой из основных составляющих использовать по 256 значений (восемь двоичных разрядов), как это принято для полутонных черно-белых изображений, то на кодирование цвета одной точки надо затратить 24 разряда. При этом система кодирования обеспечивает однозначное определение 16,5 млн. различных цветов, что на самом деле близко к чувствительности человеческого глаза. Режим представления цветной графики с использованием 24 двоичных разрядов называется полноцветным (True Color).

Каждому из основных цветов можно поставить в соответствие дополнительный цвет, то есть цвет, дополняющий основной цвет до белого. Нетрудно заметить, что для любого из основных цветов дополнительным будет цвет, образованный суммой пары остальных основных цветов. Соответственно, дополнительными цветами являются: голубой (Cyan, C), пурпурный (Magenta, M) и желтый (Yellow, Y).

Принцип декомпозиции произвольного цвета на составляющие компоненты можно применять не только для основных цветов, но и для дополнительных, то есть любой цвет можно представить в виде суммы голубой, пурпурной и желтой составляющей. Такой метод кодирования цвета принят в полиграфии, но в

полиграфии используется еще и четвертая краска — черная (Black, K). Поэтому данная система кодирования обозначается четырьмя буквами CMYK (черный цвет обозначается буквой K, потому, что буква B уже занята, синим цветом), и для представления цветной графики в этой системе надо иметь 32 двоичных разряда. Такой режим тоже называется полноцветным (True Color).

Если уменьшить количество двоичных разрядов, используемых для кодирования цвета каждой точки, то можно сократить объем данных, но при этом диапазон кодируемых цветов заметно сокращается. Кодирование цветной графики 16-разрядными двоичными числами называется режимом High Color.

При кодировании информации о цвете с помощью восьми бит данных можно передать только 256 цветовых оттенков. Такой метод кодирования цвета называется индексным. Смысл названия в том, что, поскольку 256 значений совершенно недостаточно, чтобы передать весь диапазон цветов, доступный человеческому глазу, код каждой точки раstra выражает не цвет сам по себе, а только его номер (индекс) в некоей справочной таблице, называемой палитрой.

Разумеется, эта палитра должна прикладываться к графическим данным — без нее нельзя воспользоваться методами воспроизведения информации на экране или бумаге (то есть, воспользоваться, конечно, можно, но из-за неполноты данных полученная информация не будет адекватной: листва на деревьях может оказаться красной, а небо — зеленым).

## **ЗАКЛЮЧЕНИЕ**

Каждый новый вид данных, добавляемый к компьютерной обработке, исторически тем или иным способом сводится к числовому представлению.

Исходя из принципов устройства компьютера, можно утверждать, что любые данные хранятся и обрабатываются в нем в двоичном виде, так как все виды данных так или иначе преобразуются в числовую форму, а числа хранятся в двоичной системе.

В современных компьютерах вся информация также хранится в виде последовательностей нулей и единиц. Однако двоичная система счисления в чистом виде обладает рядом принципиальных недостатков, тормозящих развитие компьютерной техники.

Главными из этих недостатков являются проблема представления отрицательных чисел и «нулевая избыточность» (т.е. отсутствие избыточности, из чего вытекает невозможность определения, произошло ли искажение информации при её передаче). Практическая потребность в решении этих вопросов вызывает сегодня повышенный интерес к способам представления информации в компьютере и новым компьютерным арифметикам. Так, например, Джон фон Нейман доказал теорему о том, что троичная система счисления позволяет наиболее эффективно среди всех основных позиционных систем счисления «сворачивать» информацию о вещественном числе.

В связи с избытком систем кодирования текстовых данных, действующих в России, возникает задача межсистемного преобразования данных — это одна из распространенных задач информатики.

Если проанализировать организационные трудности, связанные с созданием единой системы кодирования текстовых данных, то можно прийти к выводу, что они вызваны ограниченным набором кодов (256). В то же время очевидно, что если, например, кодировать символы не восьмиразрядными двоичными числами, а числами с большим количеством разрядов, то и диапазон возможных значений кодов станет намного больше. Такая система, основанная на 16-разрядном кодировании символов, получила название универсальной — UNICODE.

Шестнадцать разрядов позволяют обеспечить уникальные коды для 65536 различных символов — этого поля достаточно для размещения в одной таблице символов большинства языков планеты. Несмотря на тривиальную очевидность такого подхода, простой механический переход на данную систему долгое время сдерживался из-за недостаточных ресурсов средств вычислительной техники (в системе кодирования UNICODE все текстовые документы автоматически становятся вдвое длиннее).

Во второй половине 90-х годов технические средства достигли необходимого уровня обеспеченности ресурсами, и сегодня мы наблюдаем постепенный перевод документов и программных средств на универсальную систему кодирования. Для индивидуальных пользователей это еще больше добавило забот по согласованию документов, выполненных в разных системах кодирования, с программными средствами, но это надо понимать как трудности переходного периода.

Память компьютера, как бы велика она не была, состоит из конечного числа отдельных битов. Данные в ЭВМ представлены длинными цепочками цифр, это

перегружает память и создает определенные трудности с передачей и сохранением данных. Отсюда вытекает необходимость разработок новых алгоритмов и кодов сжатия информации и ее восстановления.

Компьютер способен работать только с дискретными данными.

Например, система целых чисел является дискретной, а система действительных чисел непрерывной.

Непрерывная (аналоговая) величина ассоциируется с графиком функции, а дискретная – с таблицей ее значений. При рассмотрении этих двух объектов разной природы можно сделать вывод о том, что с уменьшением интервала дискретизации (увеличением количества точек в таблице) различия между величинами существенно уменьшаются. Это означает, что при таких условиях дискретизированная величина хорошо описывает исходную (непрерывную).

Таким образом, в тех случаях, когда рассматриваемая величина имеет настолько большое количество значений, что мы не в состоянии их различить, ее практически можно считать непрерывной.

Так снимается существующая проблема преобразования естественной информации в пригодную для компьютера дискретную форму.

Наше время называют эпохой цифровых технологий. На сегодня совершенно очевидно лишь одно – цифровые технологии находятся лишь в начале своего пути, и нам еще только предстоит понять, что значит их повсеместное внедрение совместно с миниатюризацией, наращиванием вычислительных мощностей и объемов памяти. Достаточно быстрое развитие и постоянное удешевление цифровых устройств дает основание утверждать, что совсем скоро «цифра» полностью вытеснит аналоговые методы записи и обработки информации.

## **СПИСОК ЛИТЕРАТУРЫ**

- 1.Кудинов Ю. И., Пашенко Ф. Ф. “ Основы современной информатики”.
- 2.И.В.Кузьмин, В.А. Кедрус “ Основы теории информации и кодирования”.
3. Профессор Игорь Н. Бекман “ КОМПЬЮТЕРЫ В ИНФОРМАТИКЕ” Курс лекций  
Лекция 4. КОДИРОВАНИЕ ИНФОРМАЦИИ

4. Плоткин Дмитрий Арнольдович “НОВЫЕ ЭФФЕКТИВНЫЕ МЕТОДЫ ЭНТРОПИЙНОГО КОДИРОВАНИЯ МЕДИАДАННЫХ”.
5. Книга: “Информатика Базовый курс” Под редакцией С. В. Симоновича.
6. <http://profbeckman.narod.ru/informlekc.htm>
7. Ф. И. Соловьева “ВВЕДЕНИЕ В ТЕОРИЮ КОДИРОВАНИЯ”.
8. [zdamsam.ru/a24104.html](http://zdamsam.ru/a24104.html)
9. <http://megalektsii.ru/s21176t2.html>
10. Н.Н. Трушин “Информатика”.
11. [http://books.sernam.ru/book\\_htc.php?id=2](http://books.sernam.ru/book_htc.php?id=2)
12. Чепкунова Е.Г. Название: Пособие к подготовке к экзамену по дисциплине «Теоретические основы информатики». Раздел «Кодирование информации»: