Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
«Сибирский федеральный университет»

Лесосибирский педагогический институт - филиал
федерального государственного автономного образовательного учреждения
высшего профессионального образования
«Сибирский федеральный университет»
(ЛПИ-филиал СФУ)

ТЕОРИЯ ИНФОРМАЦИИ

Учебное пособие

Лесосибирск 2012

Киргизова Е.В. Теория информации: Учебное пособие. 2012. – 113 с.

Учебное пособие освещает один из разделов дисциплины «Теоретическая информатика».

Пособие предусматривает последовательность изучения теоретического материала, которое излагается в форме доступной пониманию студентов. В пособии представлены практические работы, позволяющие научить студентов применять теоретические знания и практической деятельности.

Учебное пособие соответствует ГОС ВПО по специальности «Информатика».

Предисловие

От автора

Учебно-методическое пособие по курсу «Теоретические основы информатики» соответствует Государственному образовательному стандарту высшего специального образования по специальности 030100 «Информатика» и охватывает основные разделы теории информации.

В учебно-методическом пособии излагаются основные понятия и факты теории информации. Рассмотрены способы измерения, передачи и обработки информации. Значительное внимание уделено вопросам кодирования информации.

Материал учебно-методического пособия разбит на ??? разделов. По каждому разделу представлен необходимый теоретический материал и практическая работа. Цель практической работы - выработать практические навыки основных положений теории информации и ее методов.

Высокая требовательность студенческой аудитории является постоянным стимулом в поиске более простых и понятных форм изложения. Автор надеется, что учебно-методическое пособие «Теория информации» не получилось чрезмерно сложным.

Данное учебно-методическое пособие предназначено студентам специальности «Информатика», а также может быть использовано учителями информатики для углубленного изучения информатики в профильных классах.

1. ПОНАТИЕ ИНФОРМАЦИИ. ТЕОРИЯ ИЗМЕРЕНИЯ ИНФОРМАЦИИ.

1.1 Информация как философская категория

К понятию информации можно подойти с двух сторон. Философское понятие определяет информацию как отражение реального мира; практическое понятие считает информацией все сведения, являющиеся объектом хранения, передачи и преобразования.

Основополагающим в философии является понятие материи. *Материя* – философская категория для представления объективной реальности, которая отображается нашими ощущениями, существуя независимо от них. Материя несотворима и неуничтожима, вечна и бесконечна.

Немало ученых и философов считают [10], что материи присущи три свойства (ипостаси) существования: *вещество,* отражающее постоянство материи; *движение,* отражающее изменение материи и *отражение,* показывающее структуру и организацию материи.

Мерой вещества является *масса*, мерой движения – *энергия*, а мерой отражения – *информация*. При вещественных преобразованиях происходит изменение массы (вспомните закон сохранения вещества), для движения нужны затраты энергии, а вот при отражении происходит как бы передача организации одного объекта другому, но сама структура материального объекта не изменяется. Это свойство хорошо подметил писатель Б. Шоу: «Если у Вас и у меня есть по яблоку, и мы обменяемся ими, у нас будет по одному яблоку. Но если у меня и у Вас есть по идее, и мы обменяемся ими, то у каждого теперь будет две идеи».

Для того чтобы отражение материи осуществлялось, необходимы, во-первых, средства сделать структуру материального объекта известной другому объекту, а во-вторых, средства увидеть эту структуру другим объектом. Свойство отражения имеет место и в неживой природе. Например, все твердые тела построены на основе кристаллической решетки, определяемой для каждого вещества взаимодействием атомов.

Наиболее ярко свойство отражения проявляется в живой материи – совокупности живых организмов, способных к самовоспроизведению. Для нормального существования и воспроизводства необходимо общение отдельных особей. Природа наделила их органами чувств для восприятия окружающей среды. Общение отдельных особей (передача информации) обеспечивает лучшую организацию деятельности живого сообщества для достижения поставленных целей (для низших организмов – выживание и размножение).

1.2 Понятие и виды информации

Можно считать, что термин «информация» в начальном представлении является общим понятием, означающим некоторые сведения, совокупность данных, знаний и т.д. Понятие информации должно быть с определенным объектом, свойства которого она отражает. Кроме того, наблюдается относительная независимость информации от ее носителя, поскольку возможны ее преобразование и передача по различным физическим средам с помощью разнообразных физических сигналов безотносительно к ее содержанию, т.е. к семантике, что и явилось центральным вопросом многих исследований, в том числе и в философской науке. Информация о любом материальном объекте может быть получена путем наблюдения, натурного либо вычислительного эксперимента, а также на основе логического вывода.

Поэтому говорят о доопытной (или априорной) информации и послеопытной (т.е. апостериорной) полученной, в итоге эксперимента.

Для человека любое восприятие реальных объектов окружающей действительности происходит через ощущения. Органы чувств человека и высшая нервная система позволяют ему воспринимать объекты. При обмене информацией имеют место источник в виде объекта материального мира и приемник - человек либо какой-то материальный объект. Информация возникает за счет отражения, которое является свойством всей материи, любой материальной системы. Свойство отражения совершенствуется по мере развития материи от элементарного отражения до высшей его формы - сознания. Процесс отражения означает взаимодействие объектов материального мира. Этот процесс наиболее прост в неорганической природе. Здесь преобладают механические, химические и физические взаимодействия. При таком отражении объекты пассивны. Новые формы отражения (физиологическое и психологическое) возникают в органической природе. В живом организме на основе отражения формируется способность приспосабливаться к изменяющимся окружающим условиям. У человека получают развитие более сложные формы отражения: познавательная и творческая. Эти формы носят сознательный характер и позволяют человеку активно воздействовать на окружающий мир.

Выделяют следующие аспекты информации:

- прагматический,
- семантический,
- синтаксический.

Прагматический аспект связан с возможностью достижения поставленной цели с использованием получаемой информации. Этот аспект информации влияет на поведение потребителя. Если информация была эффективной, то поведение потребителя меняется в желаемом направлении, т. е. информация имеет прагматическое содержание. Таким образом, этот аспект характеризует поведенческую сторону проблемы.

Семантический аспект позволяет оценить смысл передаваемой информации, соотнося ее с информацией, хранящейся до появления данной. Семантические связи между словами или другими смысловыми элементами языка отражают словарь-тезаурус. Он состоит из двух частей: списка слов и устойчивых словосочетаний, которые сгруппированы по смыслу, и некоторого ключа, т е. алфавитного словаря, позволяющего расположить слова и словосочетания в определенном порядке. Тезаурус имеет особое значение в системах хранения информации, в которые могут вводиться семантические отношения, в основном подчинения, что позволяет на логическом уровне осуществлять организацию информации в виде отдельных записей, массивов и их комплексов. Существуют развитые тезаурусы, в которые включаются сложные высказывания и семантические связи между ними. Это позволяет хранить более сложную информацию и детально оценивать семантическое содержание вновь поступающей информации. Наличие тезауруса позволяет переводить поступающую семантическую информацию на некоторый стандартизированный семантический язык в соответствии с выбранным тезаурусом. Таким образом, при возникновении информации можно изменить исходный тезаурус. Степень изменения тезауруса может быть принята как характеристика количества информации.

Синтаксический аспект информации связан со способом ее представления. В зависимости от реального процесса, в котором участвует информация, т.е. осуществляется ее сбор, передача, преобразование, отображение, представление, ввод или вывод, она представляется в виде специальных знаков, символов. Характерным носителем информации является **сообщение,** под которым обычно понимают все то, что подлежит передаче. Сообщения представляют в виде электрического сигнала, передаваемого по выбранной физической среде. Для этого сообщение подвергают преобразованию, т. е. придают ему электрический характер, далее кодированию, при котором сообщение превращается в некоторую. последовательность символов, однозначно его отображающих, и модуляции, при которой каждый элемент кода (либо код в целом) переводится в электрический сигнал, способный передаваться на заданное расстояние по выбранному каналу связи. Процессы преобразования, кодирования и модуляции исключительно многообразны, а синтаксический аспект информации при ее передаче в настоящее время хорошо развит. Иной характер синтаксический аспект имеет, например, при хранении информации. В этом случае могут быть предложены такие формы, при которых удается осуществить быстрый поиск, введение новой информации, вывод требуемой информации из информационной базы и в целом обновления базы данных. Требуемому представлению информации при ее хранении отвечают разработанные к настоящему времени типовые структуры баз и банков данных, которые позволяют наилучшим образом реализовать информационное обслуживание пользователей в системе управления. Таким образом, развитие общества привело к тому, что оказалось необходимым хранить, обрабатывать, передавать, преобразовывать огромные объемы данных.

Виды информации

Все виды деятельности человека по преобразованию природы и общества сопровождались получением новой информации. Логическая, адекватно отображающая объективные закономерности природы, общества и мышления получила название *научной информации.* Ее делят по областям получения или пользования на следующие виды: политическую, техническую, биологическую, химическую, физическую и т.д.; **по назначению -** на массовую и специальную. Часть информации, которая занесена на бумажный носитель, получила название *документальной информации.*. Любое производство при функционировании требует перемещения документов, т.е. возникает документооборот. Для автоматизированных систем управления информация в документах составляет внешнее информационное обеспечение. В то же время большая часть информации хранится в памяти ЭВМ на магнитных лентах, дисках и т.д. Она определяется как внутримашинное информационное обеспечение.

Наряду с научной информацией в сфере техники при решении производственных задач используется *техническая информация.* Она сопровождает разработку новых изделий, материалов, конструкций агрегатов, технологических процессов. Научную и техническую информацию объединяют термином научно-техническая *информация:* в сфере материального производства может циркулировать технологическая информация, закрепленная в конструкторско-технологической документации. В плановых расчетах существует планово-экономическая информация, которая содержит интегральные сведения о ходе производства, значения различных экономических показателей.

Информация с точки зрения ее возникновения и совершенствования проходит следующий путь: человек наблюдает некоторый факт окружающей действительности, это факт отражается в виде совокупности данных, при последующем структурировании в соответствии с конкретной предметной областью данные превращаются в знания. Таким образом, верхним уровнем информации как результата отражения окружающей действительности (результата мышления) являются знания. Знания возникают как итог теоретической и практической деятельности. Информация в виде знаний отличается высокой структуризацией. Это позволяет выделить полезную информацию при анализе окружающих нас физических, химических и прочих процессов и явлений. На основе структуризации информации формируется информационная модель объекта. По мере развития общества информация как совокупность научно-технических данных и знаний превращается в базу системы информационного обслуживания научно-технической деятельности общества.

В настоящее время информация используется всеми отраслями народного хозяйства и наряду с энергией, полезными ископаемыми является ресурсом общества. С развитием общества возникает необходимость целесообразной организации информационного ресурса, т.е. конкретизации имеющихся фактов, данных и знаний по направлениям науки и техники. Признание информации как ресурса и появление понятия *информационный ресурс* дало толчок развитию нового научного направления - информатики. Информатика как область науки и техники связана со сбором и переработкой больших объемов информации на основе современных программно-аппаратных средств вычислительной техники и техники связи. Информатика изучает свойства информационных ресурсов, разрабатывает эффективные методы и средства их организаций, преобразования и применения. На основе достижений информатики формируются новые методы и алгоритмы преобразования информации, при которых не квалифицированный в вычислительной технике пользователь, на языке, близком к, естественному, может общаться с вычислительной средой для решения требуемых практических задач. На пользовательском уровне информатика дает основу для создания современных информационных систем, таких как автоматизированные системы управления, автоматизированные системы научных исследований, информационно-справочные системы, интеллектуальные системы, системы управления реального времени и др. Учитывая, что техническими средствами информатики являются вычислительные средства, ее современное состояние и направления дальнейшего развития в значительной степени определяются перспективами создания, развития и внедрения персональных ЭВМ, сетей связи, языков общения пользователя с вычислительной техникой. Информатика как область науки и техники требует своего дальнейшего развития. В качестве основных направлений исследований в области информатики можно определить следующее: разработка новой информационной технологии проектирования систем; развитие интеллектуальных методов доступа пользователя к вычислительной среде; создание моделей анализа и синтеза информационных процессов: совершенствование программных и аппаратных средств вычислительной техники и техники связи: переход к интеллектуальным АСОИУ (автоматизированная система обработки информации управления) на основе гибридных экспертных систем.

1.3 Непрерывная и дискретная информация

Информация о явлениях и технологических процессах воспринимается человеком с помощью органов чувств и измерительных приборов. Явление или процесс может характеризоваться одной или многими величинами и зависеть от одного или нескольких (вектор) параметров. Интересующая нас величина зависит от многих параметров и может быть выражена функцией

$$Y = f(x_1, x_2, \dots, x_n, t).$$

Обычно мы интересуемся зависимостью только от времени. В этом случае функция $Y = f(t)$ изображается графиком (рис. 1.1). Значение Y определено для любого момента времени и может принимать любое из диапазона возможных значений. Такая величина называется непрерывной. Величины непрерывного значения называют еще аналоговыми.

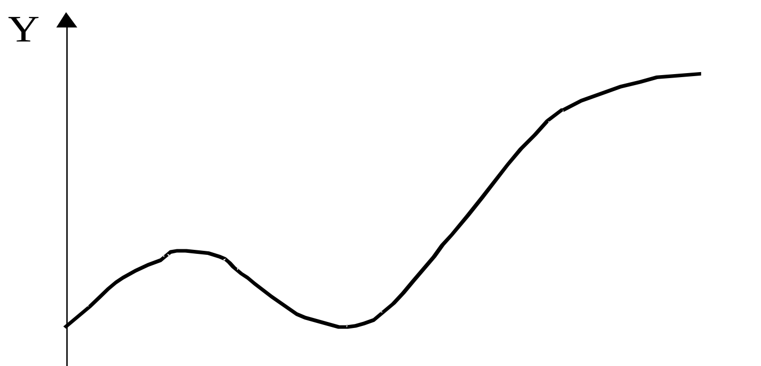


Рис. 1.1 – Непрерывная во времени и по значению величина

Измерение значений величины Y производится в некоторые моменты времени $(t_1, t_2, \dots, t_n)$, обычно через интервал Δt. В этом случае Y *дискретна* по времени, но *непрерывна* по значению (рис. 1.3).

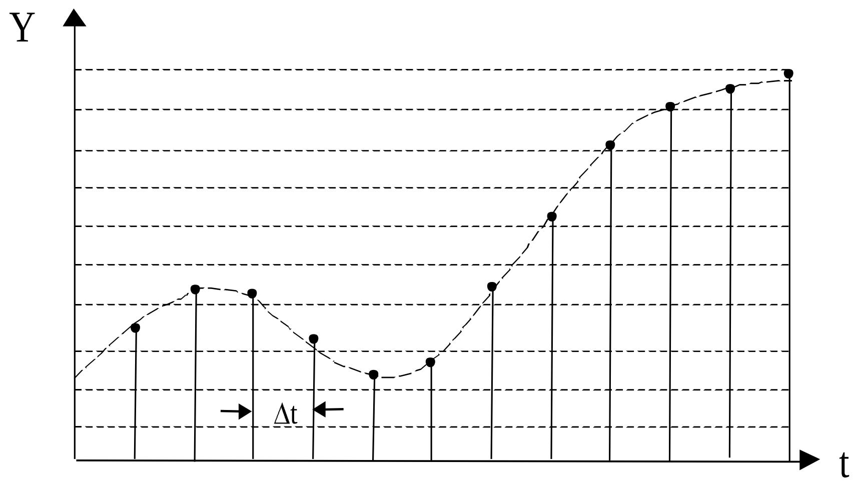


Рис. 1.2 – Непрерывная по значению и дискретная во времени величина

Точность измерительных приборов ограничена, а результат выражается числом с ограниченным количеством цифр. Величина *Y* может принимать отдельные фиксированные (дискретные, квантованные) значения с шагом *ΔY* (рис. 1.4). Величина *Y* меняется скачком, когда ее приращение превышает размер кванта *ΔY*.

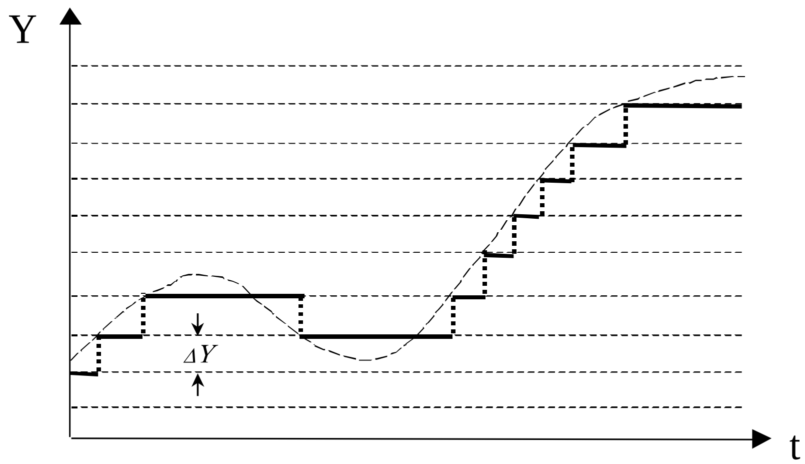


Рис. 1.3 – Квантованная по значению и непрерывная во времени величина

Величина *Y*, дискретная во времени и квантованная по величине, приведена на рисунке 1.5. Дискретную форму представления данных часто отождествляют с цифровой информацией. Обработка информации на ЭВМ в цифровой форме и в определенные моменты времени предполагает именно такое представление данных.

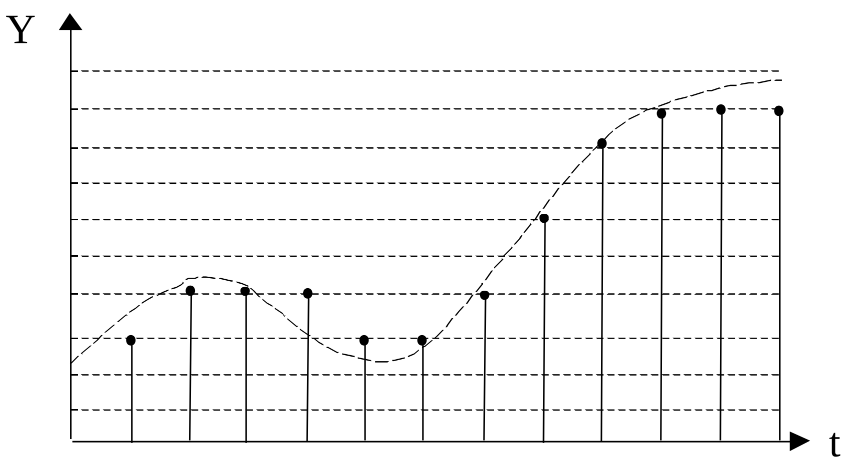


Рис. 1.4 – Квантованная по значению и дискретная во времени величина

Любая непрерывная информация может быть аппроксимирована дискретной информацией с любой степенью точности, поэтому можно говорить об универсальности дискретной информации.

Представление воспринятой от объекта информации чаще всего заключается в дискретизации во времени и квантованию по значению, т.е. в преобразовании ее в цифровую форму.

Хранение, измерение, обработка и передача информации

Для хранения информации используются специальные устройства памяти. Дискретную информацию хранить гораздо проще непрерывной, т. к. она описывается последовательностью чисел. Если представить каждое число в двоичной системе счисления, то дискретная информация предстанет в виде последовательностей нулей и единиц. Присутствие или отсутствие какого-либо признака в некотором устройстве может описывать некоторую цифру в какой-нибудь из этих последовательностей. Например, позиция на дискете описывает место цифры, а полярность намагниченности — ее значение. Для записи дискретной информации можно использовать ряд переключателей, перфокарты, перфоленты, различные виды магнитных и лазерных дисков, электронные триггеры и т.п. Одна позиция для двоичной цифры в описании дискретной информации называется битом (bit, binary digit). Бит служит для измерения информации. Информация размером в один бит содержится в ответе на вопрос, требующий ответа “да” или “нет”. Не прерывную информацию тоже измеряют в битах. Бит — это очень маленькая единица, поэтому часто используется величина в 8 раз большая — байт (byte), состоящая из двух 4-битных полубайт или тетрад. Байт обычно обозначают заглавной буквой B или Б. Как и для прочих стандартных единиц измерения для бита и байта существуют производные от них единицы, образуемые при помощи приставок кило (K), мега (M), гига (G или Г), тера (T), пета (P или П) и других. Но для битов и байтов они означают не степени 10, а степени двойки: кило – $2^{10} = 1024 \approx 10^3$, мега – $2^{20} \approx 10^6$, гига – $2^{30} \approx 10^9$, тера – $2^{40} \approx 10^{12}$, пета – $2^{50} \approx 10^{15}$. Например, 1 KB = 8 Kbit = 1024 B = 8192 bit, 1 МБ = 1024 КБ = 1048576 Б = 8192 Кбит. Для обработки информации используют вычислительные машины, которые бывают двух видов: ЦВМ (цифровая вычислительная машина) – для обработки дискретной информации, АВМ (аналоговая вычислительная машина) – для обработки непрерывной информации. ЦВМ – универсальны, на них можно решать любые вычислительные задачи с любой точностью, но с ростом точности скорость их работы уменьшается. ЦВМ – это обычные компьютеры. Каждая АВМ предназначена только для узкого класса задач, например, интегрирования или дифференцирования. Если на вход такой АВМ подать сигнал, описываемый функцией $f(t)$, то на ее выходе появится сигнал $F(t)$ или $f'(t)$. АВМ работают очень быстро, но их точность ограничена и не может быть увеличена без аппаратных переделок. Программа для АВМ — это электрическая схема из заданного набора электронных компонент, которую нужно физически собрать. Бывают еще и гибридные вычислительные машины, сочетающие в себе элементы как ЦВМ, так и АВМ. На рис. 2 изображена схема передачи информации.

Кодированием, например, является шифровка сообщения, декодированием — его дешифровка. Процедуры кодирования и декодирования могут повторяться много раз. Ошибки при передаче информации происходят из-за шума в канале (атмосферные и технические помехи), а также при кодировании и декодировании. Теория информации изучает, в частности, способы минимизации количества таких ошибок.

КОДИРОВАНИЕ КАНАЛ СВЯЗИ ДЕКОДИРОВАНИЕ

$$\begin{array}{ccccccc} & \text{КОДИРОВАНИЕ} & & \text{КАНАЛ СВЯЗИ} & & \text{ДЕКОДИРОВАНИЕ} & \\ & \downarrow & & \downarrow & & \downarrow & \\ \text{ИСТОЧНИК} & \longrightarrow & \text{ПЕРЕДАТЧИК} & \longrightarrow & \text{ПРИЕМНИК} & \longrightarrow & \text{ПОЛУЧАТЕЛЬ.} \end{array}$$

Рис. 1.5

Скорость передачи информации измеряется в количестве переданных за одну секунду бит или в бодах (baud): 1 бод = 1 бит/сек (bps). Производные единицы для бода такие же как и для бита и байта, например, 10 Kbaud = 10240 baud.

Информацию можно передавать *последовательно*, т. е. бит за битом, и *параллельно*, т.е. группами фиксированного количества бит. Параллельный способ быстрее, но он часто технически сложнее и дороже особенно при передаче данных на большие расстояния. Параллельный способ передачи используют, как правило, только на расстоянии не более 5 метров.

1.5 Мера неопределенности. Энтропия

В процессе управления объектами или в бытовой ситуации исполнитель должен выбирать действия, приводящие к достижению намеченной цели. Всегда есть выбор из нескольких возможных действий, одни действия лучше в данной ситуации, другие – хуже. Получение информации приводит к более правильному выбору. Собираясь утром на работу или учебу, мы должны одеться по погоде. Осенью, когда погода может резко меняться на следующий день, мы можем ограничиться легким плащом либо надеваем теплое пальто. Проблему одежды мы обычно решаем получением сведений о температуре наружного воздуха. Для этого достаточно выглянуть в окно или послушать сводку погоды. Сводка погоды обеспечивает более точный выбор одежды, значит, дает нам больше информации, чем взгляд в окно.

Получение информации связано с уменьшением неопределенности. Мера неопределенности должна быть связана с числом событий, которые могут произойти. Чем больше событий, тем больше неопределенность.

Одни события могут происходить чаще, другие реже. С каждым событием связана частота появления или вероятность. Пусть

$S_1S_2S_3...S_n$ – события;

$P_1P_2P_3...P_n$ – вероятности.

При этом $\sum_{i=1}^{n} P_i = 1$ (полная группа событий).

Мера неопределенности должна удовлетворять условиям:

а) чем больше число событий, тем неопределенность больше;

б) неопределенность наибольшая в ансамбле событий, если все события равновероятны, т.е. $P_i = \frac{1}{n}$;

в) неопределенность не должна зависеть от конкретных значений наблюдаемых явлений, а должна зависеть только от распределения случайных событий.

Мера неопределенности, удовлетворяющая этим условиям, была предложена американским инженером Клодом Шенноном и названа *энтропией*:

$$H(S) = -\sum_{i=1}^{n} P_i \log P_i \;.$$

Если взять основание логарифма 2, то

$$H(S) = -\sum_{i=1}^{n} P_i \log_2 P_i \;.$$

Если все события равновероятны, то $P_i = \frac{1}{n}$ и

$$H(S) = -\frac{1}{n}\sum_{i=1}^{n} \log\frac{1}{n} = \frac{1}{n}\sum_{i=1}^{n} \log n = \log n \,.$$

При n=2 и основании логарифма 2 значение $H(S)$=1. Эта единица неопределенности является двоичной и называется *бит*.

В русском языке 32 буквы, если считать буквы Е и Ё за одну. При одинаковых вероятностях использования в тексте всех 32 букв неопределенность на одну букву составит

$$H(S) = \log_2 32 = 5\,\textit{бит}\,, \quad p_i = 1/n = 1/32 = 0.03125 \,.$$

Но если взять распределение букв в словах реального русского текста художественных произведений: О=0.096; Е=0.074; А=0.064; …; Ч=0.013; Щ и Э=0.003, то по формуле неопределенности получаем только $H(S) = 4.42$ двоичные единицы.

Свойства энтропии:

– энтропия – неотрицательная величина, так как $P_i < 1$, а $\log P$ – отрицателен;

– если $P_i = 1$, а все остальные $P_i = 0$, то $H(S) = 0$;

– энтропия максимальна, если $P_i = \frac{1}{n}$;

– энтропия объединения нескольких независимых источников информации равна сумме энтропий

$$H(S_1, S_2) = H(S_1) + H(S_2) \,.$$

Проверим второе свойство энтропии.

$$H(S) = -1 \cdot \log 1 + (-0) \cdot \log 0 = 1 \cdot 0 + (-0) \cdot (-\infty).$$

Последнее произведение есть неопределенность.

Найдем предел второго слагаемого

$$\lim_{P_i \to 0} (-P_i \log P_i) = \lim_{P_i \to 0} \left(\frac{\log {}^1\!/_{P_i}}{{}^1\!/_{P_i}} \right) = \left(\frac{\log \infty}{\infty} \right) = \frac{\infty}{\infty}. \,.$$

Обозначим $\alpha = {}^1\!/_{P_i}$ и найдем предел по правилу Лопиталя как предел отношения производных числителя и знаменателя

$$\lim_{P_i \to 0} \frac{(\log \alpha)'}{\alpha'} = \lim_{P_i \to 0} \frac{({}^1\!/_{\alpha}) \log e}{1} = \lim_{P_i \to 0} \frac{P_i \log e}{1} \to 0.$$

Для n=2

$$H(S) = -[P \log P + (1-P) \log(1-P)].$$

График зависимости энтропии от P приведен на рис. 1.6. Действительно H(S) максимальна при P = 0.5.

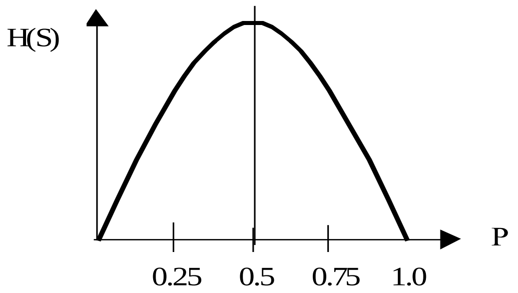


Рис. 1.6 – Энтропия двух событий

1.6 *Способы измерения информации*

1.6.1 Количество информации

Получение сообщения уменьшает неопределенность выбора решения исполнителем. Количество информации, даваемое сообщением, можно определить как разность энтропий до и после получения сообщения: $I = H_0 - H_1$. Если $H_1 = 0$, то количество информации равно начальной неопределенности H_0.

Возьмем простой случай бросания монеты. Для правильной монеты вероятности выпадения одной из сторон одинаковы: $P_1 = P_2 = 0.5$.

Начальная неопределенность

$H_0 = -0.5\log_2 0.5 - 0.5\log_2 0.5 = -1\log_2 0.5 = 1\log_2 2 = 1$ бит.

Если выпал герб, то вероятности после бросания $P_1 = 1, P_2 = 0$. Неопределенность после бросания монеты будет $H_1 = 0$ по второму свойству энтропии, а количество информации будет равно начальной энтропии H_0.

Вернемся к проблеме выбора одежды для выхода из дома. Осенью, когда погода меняется быстро, мы одинаково часто надеваем плащ и пальто. Значит, начальная энтропия, как и в случае бросания монеты, будет $H_0 = 1$. Взгляд в окно не дает точного значения температуры, поэтому неопределенность выбора одежды остается, но она становится меньше (скорее плащ, чем теплое пальто). Предположим, что вероятности можно считать $P_1 = 0.9, P_2 = 0.1$ для плаща и пальто. Тогда остаточная неопределенность и количество информации будут:

$H_1 = -0.9\log_2 0.9 - 0.1\log_2 0.1 = 0.137 + 0.322 = 0.469$ бит;

$I = H_0 - H_1 = 1 - 0.469 = 0.531$ бит.

Неопределенность не снята полностью. Надев плащ, мы иногда можем замерзнуть, если температура воздуха будет ниже предполагаемой.

1.6.2 Подходы к определению «Количество информации»

Комбинаторный поход

Пусть переменное x способно принимать значения, принадлежащие конечному множеству X, которое состоит из N элементов. Говорят, что «энтропия» переменного числа равна $H(x) = log_2 N$.

Указывая определенное значение $x = a$ переменного x, мы «снимаем» эту энтропию, сообщая «информацию»

$I = log_2 N$.

Если переменные $x_1, x_2, \dots, x_k$ способны независимо пробегать множества, которые состоят соответственно из $N_1, N_2, \dots, N_k$ элементов, то

$$H(x_1, x_2, \dots, x_k) = H(x_1) + H(x_2) + \dots + H(x_k). \quad (1)$$

Для передачи количества информации I приходится употреблять

$$I' = \begin{cases} I, & \text{при I целом,} \\ \lfloor I \rfloor + 1, & \text{при I дробном} \end{cases}$$

двоичных знаков. Например, число различных «слов», состоящих из k нулей и единиц и одной двойки, равно $2^k(k+1)$.

Поэтому количество информации в такого рода сообщении равно $I = k + log(k + 1)$, т.е. для «кодирования» такого рода слов в чистой двоичной системе требуется $I' \approx k + log_2 k$ нулей и единиц.

При изложении теории информации обычно не задерживаются надолго на таком комбинаторном подходе. Но будет существенным подчеркнуть его логическую независимость от каких бы то ни было вероятностных допущений. Пусть, например, нас занимает задача кодирования сообщений, записанных в алфавите, состоящем из s букв, причем известно, что частоты

$$p_r = s_r / s \quad (2)$$

появления отдельных букв в сообщении длины n удовлетворяет неравенству $x = -\sum_{r=1}^{s} p_r \log_2 p_r \le h.$ (3)

Легко подсчитать, что при больших n двоичный логарифм числа сообщений подчиненных требованию (3), имеет асимптотическую оценку:

$$H = log_2 N \sim nh.$$

Поэтому при передаче такого рода сообщений достаточно употребить примерно nh двоичных знаков.

Универсальный метод кодирования, который позволит передавать любое достаточно длинное сообщение в алфавите из s букв, употребляя не многим более чем nh двоичных знаков, не обязан быть чрезмерно сложным, в частности, не обязан начинаться с определения частот p_r для всего сообщения. Чтобы понять это, достаточно заметить: разбивая сообщение S на m отрезков $S_1, S_2, \dots, S_m$, получим неравенство

$$x \geq n^{-1} (n_1 x_1 + n_2 x_2 + \ldots + n_m x_m), \quad (4)$$

Важно показать, что математическая проблематика, возникающая на почве чисто комбинаторного подхода к измерению количества информации, не ограничивается тривиальностями.

Вполне естественным является чисто комбинаторный подход к понятию «энтропия речи», если иметь ввиду оценку «гибкости» речи – показателя разветвленности возможностей продолжения речи при данном словаре и данных правилах построения фраз. Для двоичного логарифма числа N русских печатных текстов, составленных из слов, включенных в «Словарь русского языка» С. И. Ожегова и подчиненных лишь требованию «грамматической правильности» длины n, выраженной в «числе знаков» (включая «пробелы»), М. Ратнер и Н. Светлова получили оценку

$$H = (\log_2 N)/n = 1{,}9 \pm 0{,}1.$$

Это значительно больше, чем оценки сверху для «энтропии литературных текстов», получаемые при помощи различных методов «угадывание продолжений». Такое расхождение вполне естественно, так как литературные тексты подчинены не только требованию «грамматической правильности».

Труднее оценить комбинаторную энтропию текстов, подчиненных определенным содержательным ограничениям. Представляло бы, например, интерес оценить энтропию русских текстов, могущих рассматриваться как достаточно точные по содержанию переводы заданного иноязычного текста. Только наличие такой «остаточной энтропии» делает возможным стихотворные переводы, где «затраты энтропии» на следование избранному метру и характеру рифмовки могут быть довольно точно подсчитаны. Можно показать, что классический четырехстопный рифмованный ямб, с некоторыми естественными ограничениями на частоту «переносов» и т.п. требует допущения свободы обращения со словесным материалом, характеризуемой «остаточной энтропией» порядка 0,4 (при указанном выше условном способе измерения длины текста по «числу знаков, включая пробелы»). Если учесть, с другой стороны, что стилистические ограничения жанра, вероятно, снижают приведенную выше оценку «полной» энтропии с 1,9 до не более чем 1,1 – 1,2, то ситуация становится примечательной как в случае перевода, так и в случае оригинального поэтического творчества.

Посмотрим, в какой мере чисто комбинаторный подход позволяет оценить «количество информации», содержащееся в переменной *x* относительно связанного с ним переменного *y*. Связь между переменными *x* и *y*, пробегающими соответственно множества *X* и *Y*, заключается в том, что не все пары *x*, *y*, принадлежащие прямому множеству возможных пар *U* определяются при любом $a \in X$ множества Y_a тех y, для которых $(a, y) \in U$.

x	y			
	1	2	3	4
1	+	+	+	+
2	+	–	+	–
3	–	+	–	–

Естественно определить условную энтропию равенством

$H (y \mid a) = log_2 N(Y_x)$ (5)

(где $N(Y_x)$ – число элементов в множестве Y_x), а информацию в *x* относительно *y* – формулой

$I(x : y) = H(y) – H(y \mid x).$ (6)

Например, в случае, изображенном в таблице имеем

$I(x = 1 : y) = 0, \quad I(x = 2 : y) = 1,$

$I(x = 3 : y) = 2.$

Понятно, что $H(y \mid x)$ и $I(x : y)$ являются функциями от *x* (в то время как y входит в их обозначение в виде «связанного переменного»).

Без труда вводится в чисто комбинаторной концепции представление о «количестве информации, необходимом для указания объекта x при заданных требованиях к точности указания».

Очевидно

$H(x \mid x) = 0 \quad I(x : x) = H(x).$ (7)

Вероятностный подход

Возможности дальнейшего развития теории информации на основе определений (5) и (6) остались в тени ввиду того, что придание переменным x и y характера «случайных переменных», обладающих определенным совместимым распределением вероятностей, позволяет получить значительно более богатую систему понятий и соотношений. В параллель к введенным в комбинаторном подходе величинам имеем

$$H_W(x) = -\sum_x p(x)\log_2 p(x), \qquad (8)$$

$$H_W(y \mid x) = -\sum_y p(y \mid x)\log_2 p(y \mid x), \qquad (9)$$

$$I_W(x : y) = H_W(y) - H_W(y \mid x). \qquad (10)$$

По прежнему $H_W(y \mid x)$ и $I_W(x : y)$ являются функциями от x. Имеют место неравенства

$$H_W(x) \le H(x), \quad H_W(y \mid x) \le H(y \mid x), \qquad (11)$$

переходящие в равенства при равномерности соответствующих распределений (на X и X_x). Величины $I_W(x : y)$ *и* $I(x : y)$ не связаны неравенством определенного знака. Как и в §1,

$$H_W(x \mid x) = 0, \; I_W(x : x) = H_W(x). \qquad (12)$$

Но отличие заключается в том, что можно образовать математические ожидания $MH_W(x : y)$, а величина

$$I_W(x, y) = MI_W(x : y) = MI_W(y : x) \qquad (13)$$

Характеризует «тесноту связи» между x и y симметричным образом.

Стоит, однако, отметить и возникновение в вероятностной концепции одного парадокса: величина $I(x : y)$ при комбинаторном подходе всегда неотрицательна, как это и естественно при наивном представлении о «количестве информации», величина же $I_W(x : y)$ может быть и отрицательной. Подлинной мерой «количества информации» теперь становится лишь осредненная величина $I_W(x, y)$.

Вероятностный подход естествен в теории передачи по каналам связи «массовой» информации, состоящей из большого числа не связанных или слабо связанных между собой сообщений, подчиненных определенным вероятностным закономерностям. В такого рода вопросах практически безвредно и ускорившееся в прикладных работах смешение вероятностей и частот в пределах одного достаточно длинного временного ряда (получающее строгое оправдание при гипотезе достаточно быстрого «перемешивания»). Практически можно считать, например, вопрос об «энтропии» потока поздравительных телеграмм и «пропускной способности» канала связи, требующегося для своевременной и неискаженной передачи, корректно поставленным в его вероятностной трактовке и при обычной замене вероятностей эмпирическими частотами. Если здесь и остается некоторая неудовлетворенность, то она связана с известной расплывчатостью наших концепций, относящихся к связям между математической теорией вероятностей и реальными «случайными явлениями» вообще.

Алгоритмический подход

По существу, наиболее содержательным является представление о количестве информации «в чем-либо» *(x)* и о «чем-либо» *(y)*. Не случайно именно оно в вероятностной концепции получило обобщение на случай непрерывных переменных, для которых энтропия бесконечна, но в широком круге случаев

$$I_W(x,y)=\iint P_{xy}(dxdy)\log_2\frac{P_{xy}(dxdy)}{P_x(dx)P_y(dy)}$$

конечно. Реальные объекты, подлежащие изучению, очень сложны, но связи между двумя реально существующими объектами исчерпываются при более простом схематизированном их описании. Если географическая карта дает нам значительную информацию об участке земной поверхности, то все же микроструктура бумаги и краски, нанесенной на бумагу, никакого отношения не имеет к микроструктуре изображенного участка земной поверхности.

Практически нас интересует чаще всего количество информации в индивидуальном объекте х относительно индивидуального объекта у. Правда, уже заранее ясно, что такая индивидуальная оценка количества информации может разумное содержание лишь в случаях достаточно больших количеств информации. Не имеет, например, смысла спрашивать о количестве информации в последовательности цифр 0 1 1 0 относительно последовательности 1 1 0 0. Но если мы возьмем вполне конкретную таблицу случайных чисел обычного в статистической практике объема и выпишем для каждой ее цифры цифру единиц ее квадрата по схеме

0 1 2 3 4 5 6 7 8 9

0 1 4 9 6 5 6 9 4 1,

то новая таблица будет содержать примерно

$$(\log_2 10 - \frac{8}{10})n$$

информации о первоначальной (n – число цифр в столбиках).

В соответствии с только что сказанным предлагаемое далее определение величины $I_A(x : y)$ будет сохранять некоторую определенность. Разные равноценные варианты этого определения будут приводить к значениям, эквивалентным лишь в смысле $I_{A1} \approx I_{A2}$.

Будем рассматривать «нумерованную область объектов», т.е. счетное множество $X = \{x\}$, каждому элементу которого поставлена в соответствие в качестве «номера» $n(x)$ конечная последовательность нулей и единиц, начинающаяся с единицы. Обозначим через $l(x)$ длину последовательности $n(x)$. Будем предполагать, что

1) соответствие между X и множеством D двоичных последовательностей описанного вида взаимно однозначно;
2) $D \subset X$, функция $n(x)$ на D общерекурсивна [1], причем для $x \in D$ $l(n(x)) \leq l(x) + C$, где C – некоторая константа;
3) Вместе с x и y в X входит упорядоченная пара (x, y), номер этой пары есть общерекурсивная функция номеров x и y и $l(x, y) \leq C_x + l(y)$, где C_x зависит только от x.

Не все эти требования существенны, но они облегчают изложение. Конечный результат построения инвариантен по отношению к переходу к новой нумерации $n'(x)$, обладающей теми же свойствами и выражающейся общерекурсивно через старую, и по отношению к включению системы X в более обширную систему X' (в предложении, что номера n' в расширенной системе для элементов первоначальной системы общерекурсивно выражаются через первоначальные номера n). При всех этих преобразованиях новые «сложности» и количества информации остаются эквивалентными в смысле $\approx$.

«Относительной сложностью» объекта y при заданном x будем считать минимальную длину $l(p)$ программы р получения

y из x. Сформулированное так определение зависит от «метода программирования». Метод программирования есть не что иное, как функция $\varphi(p,x)=y,$ ставящая в соответствие программе p и объекту y.

В соответствии с универсально признанными в современной математической логике взглядами следует считать функцию φ частично рекурсивной. Для любой такой функции полагаем

$$K_{\varphi}(y\mid x)=\begin{cases}\min\limits_{\varphi(p,x)=y} l(p), \\ \infty, & \text{если нет такого } p, \text{ что } \varphi(p,x)=y.\end{cases}$$

При этом функция $\upsilon=\varphi(u)$ от $u\in X$ со значениями $\upsilon\in X$ называется частично рекурсивной функцией преобразования номеров

$$n(\upsilon)=\Psi[n(u)].$$

Для понимания определения важно заметить, что частично рекурсивные функции не являются всюду определенными. Не существует регулярного процесса для выяснения того, приведет применение программы p к объекту x к какому-либо результату или нет. Поэтому функция $K_{\varphi}(y\mid x)$ не обязана быть эффективно вычислимой (общерекурсивной) даже в случае, когда она заведомо конечна при любых x и y.

Практическая работа №1

тема: Количество информации. Вероятностный и объемный подходы к определению количества информации.

Задание 1. Разработайте программу на каком-либо языке программирования для подсчета количества информации, приходящейся на один символ. Проверьте работу программы на примере текста.

Указание: составьте таблицу (см. образец), определив вероятность каждого символа в тексте как отношение количества одинаковых символов каждого значения ко всему числу символов в тексте. Затем по формуле Шеннона подсчитайте количество информации, приходящейся на один символ.

i	Символ	P(i)	i	Символ	P(i)	i	Символ	P(i)
1	-	0,175	12	Л	0,035	23	Б	0,014
2	О	0,090	13	К	0,028	24	Г	0,012
3	Е	0, 072	14	М	0,026	25	Ч	0,012
4	Ё	0,072	15	Д	0,025	26	Й	0,010
5	А	0,062	16	П	0,023	27	Х	0,009

6	И	0,062	17	У	0,021	28	Ж	0,007
7	Т	0,053	18	Я	0,018	29	Ю	0,006
8	Н	0,053	19	Ы	0,016	30	Ш	0,006
9	С	0,045	20	З	0,016	31	Ц	0,004
10	Р	0,040	21	Ь	0,014	32	Щ	0,003
11	В	0,038	22	Ъ	0,014	33	Э	0,003
						34	Ф	0,002

Задание 2. Решите задачу с пояснениями:

Задание 3 (творческое). Определите собственную скорость восприятия информации при чтении вслух и «про себя». Текст (1 страница любого текста полностью заполненная) для чтения должен быть новым, но понятным, т.е. информативным. Сделайте вывод и пояснения.

Варианты исходных данных к **заданию 1**

1 Организационно-правовые формы предприятий в своей основе определяют форму их собственности, то есть кому принадлежит предприятие, его основные фонды, оборотные средства, материальные и денежные ресурсы. В зависимости от формы собственности в России в настоящее время различают три основные формы предпринимательской деятельности: частную, коллективную и контрактную.

2 Общая технологическая схема изготовления сплавного транзистора напоминает схему изготовления диода, за исключением того, что в полупроводниковую пластинку производят вплавление двух навесок примесей с двух сторон. Вырезанные из монокристалла германия или кремния пластинки шлифуют и травят до необходимой толщины.

3 С конца пятнадцатого столетия в судьбах Восточной Европы совершается переворот глубокого исторического значения. На сцену истории Европы выступает новая крупная политическая сила – Московское государство. Объединив под своей властью всю северо-восточную Русь, Москва напряженно работает над закреплением добытых политических результатов и во внутренних, и во внешних отношениях.

4 С любопытством стал я рассматривать сборище. Пугачёв на первом месте сидел, облокотясь на стол и подпирая черную бороду своим широким кулаком. Черты лица его, правильные и довольно приятные, не изъявляли ничего свирепого. Все обходились между собою как товарищи и не оказывали особенного предпочтения своему предводителю.

5 Новые данные о физиологической потребности организма человека в пищевых веществах и энергии, а также выяснение закономерностей ассимиляции пищи в условиях нарушенного болезнью обмена веществ на всех этапах метаболического конвейера позволили максимально сбалансировать химический состав диет и их энергетическую ценность.

6 Информацию следует считать особым видом ресурса, при этом имеется в виду толкование «ресурса» как запаса неких знаний материальных предметов или энергетических, структурных или каких-либо других характеристик предмета. В отличие от ресурсов, связанных с материальными предметами, информационные ресурсы являются неистощимыми и предполагают существенно иные методы воспроизведения и обновления, чем материальные ресурсы.

7 Онтологическое определение национального менталитета сводится к «способности нации абсолютно определить свою судьбу, реализуя эту способность как собственную, от своего имени, под свою ответственность, самостоятельно и для себя. Это одновременно и онтологическая потенция, составляющая основу национального менталитета». В данном определении менталитет предельно сближен с национальным самосознанием и поведением. Но национальная ментальность подразумевает нормативно-оценочную сторону сознания, которая вырабатывает национальные духовно-ценностные ориентиры в жизнедеятельности этноса.

8 Ментальность формируется внутри культуры, традиций, языка, образа жизни, религиозности. Любое посягательство на один из этих компонентов непременно приведет к смятению ментальной структуры, поскольку любые инновации вызывают психологические реакции народа. При этом эти реакции могут резко отличаться у различных слоев общества, вплоть до полного неприятия и конфронтации. Урегулирование ситуации должно сопровождаться определенными взаимными уступками, ущемляя порой общую ментальность этноса.

9 Процесс трансфера отражает общую направленность человеческой личности и является результатом выбора, не в том смысле, что люди могли бы самопроизвольно выбирать тот или иной объект трансфера, а в том, что его выбор направляется ценностной ориентацией членов этноса, а последняя может не быть единой для них всех.

10 В сознании каждого члена этноса, каждой группы внутри этноса, культурная традиция преломляется особым, неповторимым образом, но вместе с тем ни когда не представляет собой хаотичного набора разрозненных культурных представлений. Традиция, распределяется между членами этноса, и регулирует действия каждого из них в отдельности, и индивидов, и внутриэтнических групп.

11 Носитель личностного сознания может сознательно выйти из своего традиционного общества, может в нем сознательно оставаться: или для того, чтобы его изменить, или для того, чтобы его сохранить. Увеличение в обществе носителей личностного сознания связано с кризисным состоянием социума. Ибо без носителей личностного сознания процессы позитивных общественных трансформаций и смены объектов трансфера на более адекватные — невозможно.

12 Управление — одно из основных понятий кибернетики. Смысл управления состоит в таком изменении поведения системы, которое направлено на достижение цели ее взаимодействия с внешним миром, к достижению которой система стремится сознательно или бессознательно.

13 Сложная динамическая система — одно из важнейших понятий кибернетики. Под динамичностью системы понимают ее изменение во времени как с точки зрения количественных и качественных параметров, так и с точки зрения ее внутренней организации в целом, взаимосвязи и состояния элементов, изменение ее положения в пространстве.

14 Следует разграничить понятия «профильное обучение» и «профильная школа». Профильное обучение — средство дифференциации и индивидуализации обучения, позволяющее за счет изменений в структуре, содержании и организации образовательного процесса более полно учитывать интересы, склонности и способности учащихся, создавать условия для обучения старшеклассников в соответствии с их профессиональными интересами и намерениями в отношении продолжения образования. Профильная школа есть институциональная форма реализации этой цели.

15 Модель общеобразовательного учреждения с профильным обучением на старшей ступени предусматривает возможность разнообразных комбинаций учебных предметов, что и будет обеспечивать гибкую систему профильного обучения. Эта система должна включать в себя следующие типы учебных предметов: базовые общеобразовательные, профильные и элективные. Базовые общеобразовательные предметы являются обязательными для всех учащихся во всех профилях обучения.

16 В процессе информатизации образовательного учреждения «создаётся качественно новое информационно-образовательное пространство, в котором увеличивающийся информационный поток заставляет всех участников процесса переходить от модели накопления знаний к системе овладения навыками самообразования».

17 Запоминающие устройства типа "память только для чтения" хранят информацию, которая в зависимости от типа ЗУ либо вообще

	не изменяется, либо изменяется редко и в специальном режиме программирования. Программирование всех видов постоянной памяти заключается в том или ином размещении элементов связи между горизонтальными и вертикальными линиями матрицы запоминающих элементов.
18	Накопители информации представляют собой гамму запоминающих устройств с различным принципом действия физическими и технически – эксплуатационными характеристиками. Основным свойством и назначением накопителей информации является хранение и воспроизведение информации. Запоминающие устройства принято делить на виды и категории в связи с их принципами функционирования, эксплуатационно-техническими физическими, программными и др. характеристиками
19	При разработке сложного алгоритма целесообразно стараться выделить в нем последовательности действий, которые выполняют решение каких-либо подзадач и могут многократно вызываться из основного алгоритма. Такие алгоритмы называются **вспомогательными** и в алгоритмических языках программирования реализуются в форме **подпрограмм**, которые вызываются из основной программы.
20	Главным достоинством экспертных систем является возможность накопления знаний и сохранение их длительное время. В отличие от человека к любой информации экспертные системы подходят объективно, что улучшает качество проводимой экспертизы. При решении задач, требующих обработки большого объема знаний, возможность возникновения ошибки при переборе очень мала.

Варианты исходных данных к **заданию 2**

1	Сколько информации содержится в сообщении о том, что сумма очков на двух подброшенных игральных костях равна семи?
2	Имеется 12 монет. Одна из них фальшивая. Неизвестно, легче она или тяжелее остальных. За какое количество взвешиваний можно определить эту монету? Пронумеруйте монеты. Найдите решение.
3	Среди девушек нашего факультета 25% - блондинки, 75% - блондинок имеют голубые глаза. Всего же голубые глаза имеет половина всех девушек факультета. Пусть мы ищем девушку с голубыми глазами. Сколько дополнительной информации содержится в сообщении о том, что она блондинка?
4	Имеются две урны, содержащие по 20 шаров: 10 белых, 5 черных и 5 красных – в первой и 8 белых, 8 черных и 4 красных – во второй. Из каждой урны достают по одному шару. Исход какого из опытов следует считать более неопределенным? Исход какого из опытов сложнее предсказать?
5	Число символов алфавита M=5. Определить количество

	информации на символ сообщения, составленного из этого алфавита: а) если символы встречаются с разными вероятностями; б) если символы алфавита встречаются в сообщении с вероятностями 0.8, 0.15, 0.03, 0.015, 0.005. Как отличается информационная нагрузка на символ в первом и во втором случаях? В каком случае символы более нагружены?
6	Производится стрельба по двум мишеням: по первой мишени сделано два независимых выстрела, по второй – три. Вероятности попадания при одном выстреле соответственно равны ½ и 1/3. Исход стрельбы по какой мишени является более определенным?
7	В двух урнах по 15 шаров, причем в первой урне 5 красных, 7 белых и 3 черных, а во второй – соответственно 4, 4, и 7. Из каждой урны вынимается по одному шару. Определить, для какой из урн исход опыта является более определенным.
8	Найти минимальное количество бинарных вопросов, которое нужно задать, чтобы определить задуманное число от 1 до 8. Какое количество информации содержат в этой ситуации сообщения о том, что: а) задуманное число четное; б) больше двух; в) больше шести; г) делится на три?
9	Определить количество информации, которое содержится в сообщении о том, что сумма выпавших очков на двух игральных костях равна семи, если нужно угадать оба выпавших числа.
10	Какое количество информации в среднем получает человек, определяющий день рождения своего собеседника, когда последний сообщает ему: а) месяц в котором он родился, но не сообщает число; б) сообщает число, но не сообщает месяц?
11	Алфавит состоит из букв А, В, С, Д. Вероятности появления букв равны соответственно: 0.25, 0.25, 0.34 и 0.16. Определить количество информации на символ сообщения, составленного из такого алфавита.
12	Имеются 27 монет, одна из них фальшивая, отличающаяся от остальных меньшим весом. Сколько взвешиваний нужно произвести, чтобы найти эту монету с помощью чашечных весов без гирь?
13	После экзамена по информатике, который сдавали ваши друзья, объявляются оценки («2», «3», «4» или «5»). Какое количество информации будет нести сообщение об оценке учащегося А, который выучил лишь половину билетов, и сообщение об оценке учащегося В, который выучил все билеты.
14	В непрозрачном мешочке хранятся 10 белых, 20 красных, 30 синих и 40 зеленых шариков. Какое количество информации будет содержать зрительное сообщение о цвете вынутого шарика.
15	Какое количество бинарных вопросов необходимо задать, чтобы определить положение фигуры на шахматной доске (то есть указать

	вертикаль и горизонталь)? Определить количество информации, содержащееся в сообщении о том: а) фигура расположена в левой части доски; б)фигура находится на вертикали «f»; в) фигура занимает одну из угловых клеток.
16	Вероятность появления события при одном испытании равна p, вероятность непоявления q=1-p. При каком p результат испытания обладает наибольшей неопределенностью?
17	Сколько информации содержится в сообщении о том, что при бросаниях двух монет оба раза было сочетание «орел»/ «орешка»?
18	Мы отгадываем задуманное кем-то двузначное число: а) Какое количество информации требуется для отгадывания всего числа? б) Изменится ли требуемое количество информации, если будем отгадывать не все число сразу, а по очереди: сначала 1-ю цифру числа, а затем 2-ю?
19	Сколько информации содержится в сообщении о том, что при бросаниях двух монет оба раза было сочетание «орел»/ «орешка»?
20	Какое количество вопросов достаточно задать собеседнику, чтобы точно определить день и месяц его рождения?

2. ТЕОРИЯ КОДИРОВАНИЯ ИНФОРМАЦИИ

2.1. ПРЕДСТАВЛЕНИЕ ДАННЫХ

2.1.1 Абстрактный алфавит. Кодирование

Цифровая информация в действительности представляет собой частный случай так называемого *алфавитного способа* представления дискретной информации. Его основой является произвольный фиксированный упорядоченный конечный набор символов (букв) любой природы, называемый *абстрактным алфавитом* или просто *алфавитом*.

Набор символов алфавита всегда фиксирован. Например, русский алфавит содержит 33 буквы, алфавит английского языка – 26 букв. Язык математических и других научных текстов может включать, наряду с обычными буквами данного языка, буквы других языков (например, древнегреческого), а также различные специальные символы (например, символы цифр, арифметических операций и др.). Число различных понятий, выражаемых символами алфавита, не превышает их числа.

Для увеличения значений, представляемых символами алфавита, используются слова. *Словом* называется любая конечная последовательность символов в этом алфавите. Если считать в русском алфавите 32 буквы (Е и Ё одна буква) и использовать слова из одной, двух и трех букв, то можно образовать $N = 32^1 + 32^2 + 32^3 = 32 + 1024 + 32768 = 33824$ различных слова. Их вполне хватит для выражения большинства мыслей, хотя в орфографическом словаре русского языка содержится более 105 тысяч слов.

При обработке информации часто возникает необходимость представить средствами одного алфавита символы других алфавитов. Такое представление называется *кодированием*. Проблема решается просто, если требуется закодировать буквы алфавита *X* с меньшим числом букв, чем у кодирующего алфавита *Y*. Если, например, *X* – алфавит десятичных цифр, а *Y* – обычный русский алфавит, то для кодирования *X* в *Y* можно использовать подмножество алфавита *Y*, например,

X – 1 2 3 4 5 6 7 8 9 0

Y – к л м н о п р с т у.

При кодировании алфавитов с большим числом символов алфавитом с меньшим числом символов используются слова кодирующего алфавита. Так, буквы русского алфавита можно закодировать парами десятичных цифр:

а = 01, б = 02, … , к = 10, л = 11, … , я = 33.

2.2 Двоичный алфавит

Простейшим абстрактным алфавитом, достаточным для кодирования любого другого алфавита, является алфавит, состоящий из двух символов. Такой алфавит носит название *двоичного*, а его символы обычно обозначают 0 и 1. Величина, способная принимать лишь два различных значения, представляет собой *информационный атом*, получивший название *бит* (bit – Binaty uniT). Ввиду своей простоты двоичный алфавит широко распространен в технических информационных устройствах и ЭВМ.

На основе двоичного алфавита можно конструировать более сложные алфавиты. Алфавиты, которыми привык пользоваться человек, применяют слова двоичного алфавита длиной n символов. В настоящее время используются восьмиразрядные двоичные слова. Такое слово получило название *байт*. Число слов в такой кодировке равно $N = 2^8 = 256$, что оказывается вполне достаточным для представления строчных и прописных букв латинского и одного из национальных алфавитов, знаков препинания и управляющих символов для технических устройств. В настоящее время наметилась тенденция увеличить число разрядов двоичного слова до 16. Такая система кодирования символов получила название – UNICODE. Шестнадцать разрядов позволяют обеспечить уникальные коды для $N = 2^{16} = 65536$ различных символов, что достаточно для размещения в одной таблице символов большинства языков планеты.

2.3 Кодирование текстовых данных

Для кодирования текста естественного языка в устройствах обработки данных достаточно иметь коды символов этого языка в двоичном алфавите. В первые годы развития вычислительной техники использовались 5-, 6- и 7-разрядные двоичные коды, применяемые в средствах связи. Для английского языка, захватившего де-факто нишу международных средств общения, институт стандартизации США (ANSI) ввел в действие систему кодирования ASCII (American Standard Code for Information Interchange – стандартный код для информационного обмена США). Этот код использует восьмиразрядные двоичные слова. В системе ASCII закреплены две части таблицы кодирования – базовая и расширенная. Базовая таблица определяет 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, занимают так называемые управляющие коды, которым не соответствуют никакие символы. Они используются для управления передачей данных и управления печатью текста. Начиная с позиции 32 по код 127 размещены коды символов английского алфавита, знаков препинания, цифр, арифметических действий и некоторых вспомогательных символов (рис. 2.7).

Расширенная часть кодовой таблицы используется для представления символов одного национального языка. Для каждого языка требуется своя расширенная часть таблицы. Отсутствует единый стандарт для национальных языков, что приводит к множественности одновременно действующих кодировок. Только в

России можно указать три действующих стандарта кодировки и еще два устаревших.

Кодовая таблица CP-866 (альтернативная) используется в персональных компьютерах с операционной системой MS DOS (рис. 2.7). Другая таблица CP-1251 предложена для России фирмой Microsoft для операционных систем Windows (рис. 2.8).

В ЭВМ с операционной системой UNIX и сетях Интернет применяется кодировка KOI8-R (рис. 2.9). В ней коды русских букв не расположены в алфавитном порядке (наследие 7-разрядного кода). Эта кодировка используется в технике связи и применялась в мини-ЭВМ до эпохи персональных компьютеров.

	32	48	64	80	96	112	128	144	160	176	192	208	224	240
0		0	@	P	`	p	А	Р	а	░	└	╨	р	Е
1	!	1	A	Q	a	q	Б	С	б	▒	┴	╤	с	е
2	"	2	B	R	b	r	В	Т	в	▓	┬	╥	т	Є
3	#	3	C	S	c	s	Г	У	г	│	├	╙	у	є
4	$	4	D	T	d	t	Д	Ф	д	┤	─	╘	ф	Ї
5	%	5	E	U	e	u	Е	Х	е	╡	┼	╒	х	ї
6	&	6	F	V	f	v	Ж	Ц	ж	╢	╞	╓	ц	Ў
7	'	7	G	W	g	w	З	Ч	з	╖	╟	╫	ч	ў
8	(	8	H	X	h	x	И	Ш	и	╕	╚	╪	ш	°
9	)	9	I	Y	i	y	Й	Щ	й	╣	╔	┘	щ	●
10	*	:	J	Z	j	z	К	Ъ	к	║	╩	┌	ъ	·
11	+	;	K	[	k	{	Л	Ы	л	╗	╦	█	ы	√
12	,	<	L	\	l	|	М	Ь	м	╝	╠	▄	ь	№
13	-	=	M	]	m	}	Н	Э	н	╜	═	▌	э	¤
14	.	>	N	^	n	~	О	Ю	о	╛	╬	▐	ю	▪
15	/	?	O	_	o	Del	П	Я	п	┐	╧	▀	я	Del

Рис. 2.7 – Кодовая таблица CP-866 (ГОСТ-альтернативная)

	32	48	64	80	96	112	128	144	160	176	192	208	224	240
0		0	@	P	`	p	Ђ	ђ		°	А	Р	а	р
1	!	1	A	Q	a	q	Ѓ	‘	Ў	±	Б	С	б	с
2	"	2	B	R	b	r	‚	’	ў	I	В	Т	в	т
3	#	3	C	S	c	s	ѓ	“	J	i	Г	У	г	у
4	$	4	D	T	d	t	„	”	¤	ґ	Д	Ф	д	ф
5	%	5	E	U	e	u	…	•	Ґ	µ	Е	Х	е	х
6	&	6	F	V	f	v	†	–	¦	¶	Ж	Ц	ж	ц
7	'	7	G	W	g	w	‡	—	§	·	З	Ч	з	ч
8	(	8	H	X	h	x	€	�	Е	е	И	Ш	и	ш
9	)	9	I	Y	i	y	‰	™	©	№	Й	Щ	й	щ

10	*	:	J	Z	j	z	Љ	љ	Є	є	К	Ъ	к	ъ
11	+	;	K	[	k	{	‹	›	«	»	Л	Ы	л	ы
12	,	<	L	\	l	|	Њ	њ	¬	ј	М	Ь	м	ь
13	-	=	M	]	m	}	Ќ	ќ	-	Ѕ	Н	Э	н	э
14	.	>	N	^	n	~	Ћ	ћ	®	ѕ	О	Ю	о	ю
15	/	?	O	_	o	Del	Џ	џ	Ї	ї	П	Я	п	Del

Рис. 2.8 – Кодовая таблица Windows 1251

	32	48	64	80	96	112	128	144	160	176	192	208	224	240
0		0	@	P	`	p	Ђ	ђ		№	ю	п	Ю	П
1	!	1	A	Q	a	q	Ѓ	‘	ђ	Ђ	а	я	А	Я
2	"	2	B	R	b	r	‚	’	ѓ	Ѓ	б	р	Б	Р
3	#	3	C	S	c	s	ѓ	“	е	Е	ц	с	Ц	С
4	$	4	D	T	d	t	„	”	є	Є	д	т	Д	Т
5	%	5	E	U	e	u	…	•	ѕ	Ѕ	е	у	Е	У
6	&	6	F	V	f	v	†	–	і	І	ф	ж	Ф	Ж
7	'	7	G	W	g	w	‡	—	ї	Ї	г	в	Г	В
8	(	8	H	X	h	x	€	©	ј	J	х	ь	Х	Ь
9	)	9	I	Y	i	y	‰	™	љ	Љ	и	ы	И	Ы
10	*	:	J	Z	j	z	Љ		њ	Њ	й	з	Й	З
11	+	;	K	[	k	{	‹	»	ћ	Ћ	к	ш	К	Ш
12	,	<	L	\	l	|	Њ	®	ќ	Ќ	л	э	Л	Э
13	-	=	M	]	m	}	Ќ	«	ґ	Ґ	м	щ	М	Щ
14	.	>	N	^	n	~	Ћ	·	ў	Ў	н	ч	Н	Ч
15	/	?	O	_	o	Del	Џ	¤	џ	Џ	о	ъ	О	Del

Рис. 2.9 – Кодовая таблица KOI8-R

Международный институт стандартизации ISO (Interna-tional Standard Organization) выпустил стандарт ISO-8859-5 для кодировки символов русского алфавита. На практике эта таблица кодировки начинает применяться в новых программах (рис. 2.10).

	32	48	64	80	96	112	128	144	160	176	192	208	224	240
0		0	@	P	`	p	Ђ	ђ		А	Р	а	р	№
1	!	1	A	Q	a	q	Ѓ	‘	Е	Б	С	б	с	е
2	"	2	B	R	b	r	‚	’	Ђ	В	Т	в	т	ђ
3	#	3	C	S	c	s	ѓ	“	Ѓ	Г	У	г	у	ѓ
4	$	4	D	T	d	t	„	”	Є	Д	Ф	д	ф	є
5	%	5	E	U	e	u	…	•	Ѕ	Е	Х	е	х	ѕ
6	&	6	F	V	f	v	†	–	І	Ж	Ц	ж	ц	і

```
| 7 | ' | 7 | G | W | g | w | ‡ | — | Ї | З | Ч | з | ч | ї |
| 8 | ( | 8 | H | X | h | x | € | � | Ј | И | Ш | и | ш | ј |
| 9 | ) | 9 | I | Y | i | y | ‰ | ™ | Љ | Й | Щ | й | щ | љ |
| A | * | : | J | Z | j | z | Љ | љ | Њ | К | Ъ | к | ъ | њ |
| B | + | ; | K | [ | k | { | ‹ | › | Ћ | Л | Ы | л | ы | ћ |
| C | , | < | L | \ | l | | | Њ | њ | Ќ | М | Ь | м | ь | ќ |
| D | - | = | M | ] | m | } | Ќ | ќ | - | Н | Э | н | э | § |
| E | . | > | N | ^ | n | ~ | Ћ | ћ | Ў | О | Ю | о | ю | ў |
| F | / | ? | O | _ | o |Del| Џ | џ | Џ | П | Я | п | я |Del|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

Рис. 2.10 – Кодовая таблица ISO-8859-5

2.4 Кодирование чисел. Позиционные системы счисления

С появлением арабских цифр человечество использует особую компактную форму слов для кодирования числовых данных. Числа записываются кодом, состоящим из цифр, разделяющей целую и дробную часть числа точки (запятой) и знака, например: -7361.83.

В общем виде код числа обозначается словом:

$$a_{n-1}a_{n-2}...a_2a_1a_0.a_{-1}a_{-2}...a_{-m}.$$

Такая форма записи числа называется сокращенной. Чтобы узнать значение числа следует записать его в полной форме, т.е. вычислить выражение

$$N = \sum_{i=-m}^{n-m-1} a_i b_i ,$$

где n – всего числовых разрядов (позиций) в числе;

m – число разрядов дробной части числа;

a_i – разрядный коэффициент (значение цифры в разряде);

b_i – вес (значение) разрядного коэффициента.

Значения весов разрядных коэффициентов неодинаковы и определяются позицией, поэтому такой код числа называется *позиционным*. Чаще всего каждый следующий (левый) разряд имеет больший вес. Вообще веса разрядных коэффициентов b_i могут выбираться произвольно. В практике позиционного представления чисел вес каждого разряда определяется параметром p – *основанием системы счисления* и равен степени i основания системы, т.е. $b_i = p^i$.

Числовое значение кода числа с основанием p равно

$$N = \sum_{i=-m}^{n-m-1} a_i \, p^i .$$

Основанием системы счисления может быть любое число. Привычным для человека является p = 10 (десятичные числа, подумайте почему). Значения разрядных коэффициентов a = 0,1,2,3,4,5,6,7,8,9.

Для целых чисел $m = 0$ и

$$N = \sum_{i=0}^{n-1} a_i\, p^i = a_{n-1}\, p^{n-1} + a_{n-2}\, p^{n-2} + \ldots + a_i\, p^i + \ldots + a_1\, p^1 + a_0\, p^0.$$

Например, $258 = 2{\cdot}10^2 + 5{\cdot}10^1 + 8{\cdot}10^0$.

Заметим, что основание системы счисления в коде числа не указывается, оно предполагается.

В информатике применяют другие значения основания. При p=2 получаем двоичную систему счисления. В ней $a_i = 0{,}1$. Вес каждого последующего разряда увеличивается не в 10, а в 2 раза.

Найдем значение двоичного числа 101011:

$$101011_2 = 1\cdot 2^5 + 0\cdot 2^4 + 1\cdot 2^3 + 0\cdot 2^2 + 1\cdot 2^1 + 1^0\cdot 2^0 =$$
$$= 32 + 8 + 2 + 1 = 43_{10}.$$

2.5 Перевод чисел из одной системы счисления в другую

Значение целого числа в позиционной системе счисления вычисляется полиномом

$$N = \sum_{i=0}^{n-1} a_i\, p^i = a_{n-1} p^{n-1} + a_{n-2} p^{n-2} + \ldots + a_i\, p^i + \ldots + a_1 p^1 + a_0\, p^0.$$

Полином обычно вычисляется по схеме Горнера

$$N = ((\ldots((0)\cdot p + a_{n-1})\,p + a_{n-2})\,p + a_{n-3})\,p + \ldots + a_2)\,p + a_1)\,p + a_0,$$

$$258 = (((0)\cdot 10 + 2)10 + 5)10 + 8 = (20 + 5)10 + 8 = 250 + 8 = 258.$$

Схема Горнера дает возможность получить алгоритм перевода кода числа из одной системы счисления в другую.

Если вычисления проводятся в новой системе, следует записать значение основания системы счисления p и разрядных коэффициентов как числа в новой системе счисления q и вычислить полином по правилам новой системы счисления. Так мы переводили число 43 из двоичной в десятичную систему.

Еще один алгоритм перевода, если вычисления производятся в новой системе. Следует вычислять полином по схеме Горнера.

Например,

$$101011_2 = ((((((0\cdot 2 + 1)2 + 0)2 + 1)2 + 0)2 + 1)2 + 1) = 43_{10}.$$

Переведем $43_{10} = 101011_2$, используя действия в двоичной системе счисления. Заметим, что двоичные значения чисел 10, 4 и 3 есть

$$10_{10} = 1010_2;\ 4_{10} = 100_2;\ 3_{10} = 11_2.$$

Подставляем эти значения и вычисляем двоичное представление числа 43:

$$43_{10} = 100 \cdot 1010 + 11 \cdot 1 = 101000 + 11 = 101011_2.$$

Для перевода надо выполнять арифметические действия в двоичной системе счисления. Это неудобно для человека, привыкшего к десятичному счету.

Перевод числа из одной системы счисления в другую можно выполнить, используя вычисления в старой системе. Выведем соотношения для перевода числа из системы счисления p в систему счисления q. Запишем полиномы для полной формы числа в обоих случаях и приравняем их, а затем приведем второй полином к схеме Горнера:

$$\begin{aligned} N &= a_n p^n + a_{n-1} p^{n-1} + \ldots + a_1 p^1 + a_0 p^0 = \\ &= b_k q^k + b_{k-1} q^{k-1} + \ldots + b_1 q^1 + b_0 q^0 = \\ &= ((0)q + b_k)q + b_{k-1}) + \ldots + b_2)q + b_1)q + b_0). \end{aligned}$$

Теперь можно сформулировать алгоритм перевода. Следует разделить число N на основание новой системы счисления q, тогда остаток есть b_0 – младший разряд числа в новой системе счисления. Процесс деления следует продолжить над частным, отмечая остатки до получения нулевого частного. Затем написать остатки в обратном порядке.

Например, $43_{10} = 101011_2$

$$\begin{array}{lllllll} \underline{43\ |2\ } & & & & & & \\ \ 1 & \underline{21\ |2\ } & & & & & \\ & \ 1 & \underline{10\ |2\ } & & & & \\ & & \ 0 & \underline{5\ |2\ } & & & \\ & & & \ 1 & \underline{2\ |2\ } & & \\ & & & & \ 0 & \underline{1\ |2\ } & \\ & & & & & \ 1 & 0 \end{array}$$

Для дробной части числа надо делить на ½, т.е. умножать на 2 и отделять целые части, пока не получим нулевую дробную часть, а затем записать целые части в порядке их получения как дробные разряды нового числа. Например, для числа 0.1875:

$$\begin{array}{r} 0.1875 \\ \underline{\qquad 2} \\ 0.3750 \\ \underline{\qquad 2} \\ 0.7500 \\ \underline{\qquad 2} \\ 1.5000 \\ \underline{\qquad 2} \\ 1.0000 \end{array} \qquad\qquad 0.1875_{10}=0.0011_2$$

Не всегда перевод дробной части ограничивается целым числом разрядов, число может быть иррациональным. Перевод заканчивается, когда будет вычислено требуемое число разрядов дробной части.

Попробуйте для примера перевести 0.2 в двоичную систему.

2.6 Другие основания систем счисления, применяемые в информатике и вычислительной технике

Двоичная система хороша для ЭВМ, но приводит к длинному коду числа, что неудобно для обозрения. Часто двоичные числа записываются короче с использованием 8-ричных или 16-ричных чисел.

8-ричная система $P = 8$ $a = 0,1,2,3,4,5,6,7,$

16-ричная система $P = 16$ $a = 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F.$

Шестнадцатеричные цифры больше 9 обозначаются прописными буквами английского алфавита.

Правила перевода аналогичны:

$43_{10} \to 2B_{16} = 101011_2$

43 |<u>16</u>

$$\begin{array}{l} \underline{32} \quad 2\,|\underline{16} \\ 11 \quad 2 \;\; 0 \\ \downarrow \\ \quad B \end{array}$$

Перевод между двоичной, восьмеричной, шестнадцатеричной системами счисления можно значительно упростить, учитывая, что $8 = 2^3$ и $16=2^4$. Значит, три двоичных разряда соответствуют одному восьмеричному, аналогично 4 разряда двоичных образуют один 16-ричный.

$$\underbrace{0010}_{2}\underbrace{1011}_{B}{}_2=2B_{16};$$

$$\underbrace{101}_{5}\underbrace{011}_{3}{}_2 = 53_8;$$

$101011_2 = 53_8 = 43_{10} = 2B_{16}$.

2.7 Оптимальное основание системы счисления

Рассматривая представление чисел в различных системах счисления, можно обнаружить, что уменьшение значения основания p приводит к увеличению количества разрядов n числа, но набор разрядных коэффициентов, равных p, становится меньше. Реализация вычислительного устройства требует одного элемента схемы на один разрядный коэффициент. Должно существовать основание системы счисления, при котором требуется минимальное количество элементов вычислительного устройства для представления чисел от 0 до значения N.

Пусть основание равно *p*. Если имеется n-разрядное число, то максимальное его значение

$$N = p^n - 1 \approx p^n.$$

Для каждого разряда надо *P* элементов для представления разрядных коэффициентов 0..*P*-1. Для n-разрядного числа нужно k элементов

$$k = n \cdot p$$

$$N = p^{\frac{k}{p}}$$

Найдем *k*, логарифмируя это выражение

$$\log_a N = \frac{k}{p} \log_a p,$$

$$k = \log_a N \cdot \frac{p}{\log_a p} = A \cdot \frac{p}{\log_a p}.$$

Для минимума требуется, чтобы производная для *k* по переменной *P* стала равной нулю. Найдем производную:

$$\frac{dk}{dp} = A \cdot \left(\frac{1 \cdot \log_a p - \frac{1}{p} \log_a e \cdot p}{(\log_a p)^2} \right) = 0.$$

$$\log_a p = \log_a e.$$

Из уравнения получаем

Оптимальное значение основания системы счисления равно числу *e*.

$$p = e = 2.71828\ldots$$

Математически основание системы счисления необязательно должно быть целым, при нецелом основании коды целых чисел будут иметь дробную часть. Однако это неудобно для восприятия. В вычислительных устройствах выбирают *P* целым. Лучшей будет троичная система счисления, но она не получила широкого распространения, элемент с тремя состояниями сложнее элемента с двумя состояниями. Двоичная система счисления тоже близка к оптимальной, и двоичные элементы легко реализуются в электронных схемах.

Подсчитаем число разрядов *n* и количество элементов *k* для разных оснований *p*, если задано максимальное значение числа *N*.

$$n = \frac{\lg N}{\lg P}; k = n \cdot p.$$

Число разрядов и число элементов определяются формулами

Возьмем $N=10^{10}$. Ниже в таблице приведены не округленные до целого числа результаты вычислений *n* и *k*.

p	2	2.7128	3	5	8	10	16
n	33.2	23.1	21	14.3	11.1	10	8.31
K	66.43	62.65	62.89	71.53	88.60	100	132.88

2.8 Смешанные и непозиционные системы счисления

Смешанные системы. Со времен древнего Вавилона используется следующее обозначение угловой меры – 27°43'14". Это трехразрядное число, состоящее из градусов (°), минут (') и секунд ("). Вес минутного разряда равен 60, один градус равен 60 минутам или $60^2 = 3600$ секундам, а значения разряда градусов

$$N = 27 \cdot 60^2 + 43 \cdot 60^1 + 14 = 27 \cdot 3600 + 43 \cdot 60 + 14 = 99794.$$

меняется от 0 до 359. Полная форма числа и его значение будут:

Основание системы счисления здесь 60, но разрядные коэффициенты закодированы двузначными десятичными числами и имеют значения от 00 до 59. Для сложения разрядов углового числа можно использовать правила десятичной системы счисления, но перенос в старший разряд производится, когда значение разряда превысит 60. Попробуйте просуммировать

25° 46' 35" + 47° 23' 45" = 73° 10' 20".

Это пример кодирования числа в смешанной системе счисления.

Двоично-десятичное представление чисел. В вычислительной технике используется двоично-десятичное представление чисел, когда каждый разряд десятичного числа кодируется в двоичном виде, например

279 = 0010 0111 1001.

Для двоичного представления значений от 0 до 9 необходимо четыре двоичных разряда. То, что используется только десять комбинаций из 16, позволяет кодировать также знак числа: '+' = $1100_2 = C_{16}$; '–' = $1101_2 = D_{16}$. Такая система кодирования является основой упакованного формата десятичных чисел, где в одном байте размещаются два разряда десятичного числа. Современные ЭВМ имеют команды десятичной арифметики, которые оперируют с числами в упакованном формате.

Единичная система счисления. Система счисления с основанием P = 1 является вырожденной, поскольку в ней прибавление единицы вызывает перенос в следующий разряд. Значение числа равно количеству его разрядов, например **6 = 111111**. Эта система счисления представляет собой обычный счет предметов, когда наличие предмета обозначают знаком: крестиком, палочкой или другим символом. Единичная система счисления не имеет символа нулевого значения.

Непозиционные системы представления чисел. Мировое общество имеет примеры непозиционного кодирования числовых значений. Древние римляне использовали особые знаки для десятичных разрядов и их половин: **I = 1; V = 5; X = 10; L = 50; C = 100; D = 500; M = 1000**.

Натуральные числа записываются при помощи этих цифр. При этом, если большая цифра стоит впереди меньшей, то они складываются, если же меньшая цифра – впереди большей, то меньшая вычитается из большей. Например,**VI = 5 + 1; IV = 5 – 1; XIX = 10 + (10 – 1) = 19; XL = 50 – 10 = 40**.

Попробуйте расшифровать значение чисел MCMXCVIII, MM, XLIV.

Выполнение арифметических действий над многозначными числами в римской системе очень трудно. Попробуйте сложить числа

$$\begin{array}{r} \mathrm{XLVII} \\ +\ \mathrm{XVIII} \\ \hline \mathrm{LXV} \end{array} \qquad \begin{array}{r} 47 \\ +\ 18 \\ \hline 65 \end{array}$$

Римская система в настоящее время не применяется. В отдельных случаях римскими цифрами обозначают века, месяцы при указании дат, а также порядковые числительные.

2.9 Прямой, обратный и дополнительный коды числа

Для арифметических операций в ЭВМ применяют специальные коды для представления чисел, позволяющие упрощать определение знака результата операции и заменять вычитание сложением, при этом облегчается выработка признаков переполнения разрядной сетки. Кроме *прямого кода* (сокращенная форма записи числа, см. 2.4), используются обратный и дополнительные коды.

Обратный код – такой код числа, который, будучи суммирован с прямым кодом, дает все цифры, равные старшим значениям в системе счисления.

Дополнительный код – такой код числа, который, будучи сложен с прямым кодом, дает все нули и единицу переноса из старшего разряда, которая выходит за пределы разрядной сетки. Для получения дополнительного кода достаточно прибавить единицу младшего разряда к обратному коду. Например, для трехзначного десятичного числа:

прямой код	275	прямой код	275
обратный код	<u>724</u>	дополнительный код	<u>725</u>
сумма	999	сумма	<u>**1**</u> 000

Прямой код нагляден для представления чисел, но плох для проведения арифметических операций. Сложение чисел с одинаковыми знаками делается просто: складываются цифровые разряды, и к ним приписывается знаковый разряд:

$$25 + 53 = 78;$$
$$(-25) + (-53) = -(25 + 53) = -78.$$

Сложение чисел с разными знаками в прямом коде требует:

определить число с наибольшим модулем;
вычесть из большего меньшее число;
присвоить результату знак наибольшего числа,

например:

$$25 + (-53) = -(53 - 25) = -28;$$
$$(-25) + 53 = (53 - 25) = 28.$$

Операцию вычитания можно свести к операции сложения по правилу A – B = A + (–B):

$$25 - 53 = 25 + (-53) = -(53 - 25) = -28;$$
$$53 - 25 = 53 + (-25) = 53 - 25 = 28;$$
$$(-53) - (-25) = (-53) + 25 = -(53 - 25) = -28;$$
$$(-25) - (-53) = (-25) + 53 = 53 - 25 = 28;$$
$$25 - (-53) = 25 + 53 = 78;$$
$$(-25) - 53 = (-25) + (-53) = -(53 + 25) = -78;$$
$$53 - (-25) = (53 + 25) = 78;$$

$$(-53) - 25 = (-53) + (-25) = -(53 + 25) = -78.$$

Эти операции упрощаются, если использовать обратный и дополнительный код числа. Операция вычитания сводится к операции сложения дополнительного кода:

$$25 - 53 = 25 + 47_{доп} = 72_{доп} = -28_{пр};$$
$$53 - 25 = 53 + 75_{доп} = \underline{\mathbf{1}}\ 28_{пр};$$
$$(-25) - 53 = 75_{доп} + 47_{доп} = \underline{\mathbf{1}}\ 22_{доп} = -78_{пр};$$
$$(-53) - 25 = 47_{доп} + 75_{доп} = \underline{\mathbf{1}}\ 22_{доп} = -78_{пр}.$$

Если абсолютное значение вычитаемого больше абсолютного значения уменьшаемого, то в результате получается дополнительный код, иначе – прямой код. Отрицательные числа лучше хранить в дополнительном коде.

Особый смысл приобретают прямой, обратный и дополнительный коды для двоичных чисел со знаком. Знаковый разряд является старшим (левым) в коде числа, причем плюс кодируется двоичным нулем, а минус – единицей.

Прямой код двоичного числа N, представляемого в n-разрядной сетке со знаком, определяется формулой:

$$N_{пр} = \begin{cases} N, & N \ge 0; \\ 2^{n-1} + |N|, & N < 0. \end{cases}$$

Если считать знаковый разряд как числовой, то положительные числа представляются беззнаковыми кодами чисел $0 \le N_{пр} < A$, а отрицательные – $A \le N_{пр} < 2 \cdot A,$ где $A = 2^{n-1}$. Цифровые разряды прямого кода содержат модуль числа N, а знаковый (старший) разряд имеет 0 для положительных и 1 для отрицательных чисел.

Дополнительный код двоичного числа N, представляемого в n – разрядной сетке (со знаком)

$$N = \begin{cases} N, & N \ge 0 \\ C - |N|, & N < 0 \end{cases}$$

где C – величина, равная весу разряда, следующего за старшим (для целых чисел $C = 2^n$).

Для положительных чисел (как у прямого кода) диапазон $0 \le N < A$, а для отрицательных $0 < |N| \le A$, где $A = 2^{n-1}$.

Рассмотрим примеры. Будем для простоты считать n=8 (с разрядом знака). Знаковый разряд отделен чертой.

Для положительных чисел:

прямой код	0 | 0010110	прямой код	0 | 0010110
обратный код	0 | 1101001	дополнительный код	0 | 1101010
сумма	0 | 1111111	сумма	**1** 0 | 0000000

Для отрицательных чисел:

прямой код	1 | 0010110	прямой код	1 | 0010110
обратный код	1 | 1101001	дополнительный код	1 | 1101010
сумма	1 | 1111111	сумма	**1** 1 | 0000000

Обратный код двоичного числа получается заменой всех нулей на единицы, а единиц – на нули (без знакового разряда). Для получения дополнительного кода надо к обратному коду прибавить единицу.

Если все разряды кода числа нули – это нулевое число (нуль всегда положительный). Если в коде числа все единицы – это –1 в дополнительном коде:

дополнительный код	1 | 1111111
отнимаем	1
обратный код	1 | 1111110
инвертируем	1 | 0000001 – это прямой код числа –1.

Если в коде числа только единица в знаковом разряде и все нули в числовых разрядах, то это дополнительный код числа –128:

дополнительный код	1 | 0000000,
обратный код	1 | 1111111,
инвертируем	1 | 0000000.

В этом случае дополнительный код равен прямому. Поскольку в дополнительном коде не существует отрицательного нуля, то это дополнительный код числа –128. Значит, диапазон представления семиразрядных двоичных чисел со знаком с использованием дополнительного кода составляет от –128 до 127 (подумайте, почему положительная граница на единицу меньше).

При алгебраическом сложении *n*-разрядных двоичных чисел в дополнительном коде производится арифметическое суммирование этих кодов, включая разряды знаков. При возникновении переноса из разряда знака, единица переноса отбрасывается.

Рассмотрим примеры на числах 25 и 53.

25_{10} = 00011001_2 – прямой код; $–25_{10}$ = 11100111_2 – дополнительный код.

53_{10} = 00110101_2 – прямой код; $–53_{10}$ = 11001011_2 – дополнительный код.

Сложение:

25 + 53 = $00011001_{\text{пр}}$

$\underline{00110101_{\text{пр}}}$

Сумма 78 = $01001110_{\text{пр}}$

(–25) + (–53) = $11100111_{\text{доп}}$

$\underline{11001011_{\text{доп}}}$

сумма $\underline{\mathbf{1}}$ $10110010_{\text{доп}}$

–78 = $11001110_{\text{пр}}$

Если в знаковом разряде результата стоит 1, то это дополнительный код (отрицательное число), если в знаковом разряде 0, то это прямой код, а результат – положительное число.

При сложении положительных чисел результат получился в прямом коде, сложение двух дополнительных кодов дает результат в дополнительном коде, т.е. отрицательное число. Обратите внимание, что при суммировании дополнительных кодов были одновременно переносы из старшего числового разряда в знаковый и из знакового. Результат получился нормальный, переполнение отсутствует, хотя есть выход за числовую разрядную сетку.

Вычитание:

25 + (–53) = $00011001_{\text{пр}}$

$\underline{11001011_{\text{доп}}}$

сумма $11100100_{\text{доп}}$

–28 = $10011100_{\text{пр}}$

53 – (–25) = $00110101_{\text{пр}}$

$\underline{11100111_{\text{доп}}}$

78 = $\underline{\mathbf{1}}$ $00011100_{\text{пр}}$

Во втором примере мы снова встречаемся с переносами из старшего числового разряда и из знакового разряда.

Признак переполнения разрядной сетки при проведении действий в дополнительном коде. При алгебраическом сложении двух двоичных чисел с использованием дополнительного кода признаком переполнения является наличие переноса в знаковый разряд суммы при отсутствии переноса из ее знакового разряда (положительное переполнение) или наличие переноса из знакового разряда суммы при отсутствии переноса в ее знаковый разряд (отрицательное переполнение). При положительном переполнении результат операции положительный, а при отрицательном – отрицательный. Если и в знаковый, и из знакового разряда суммы есть переносы, то переполнение отсутствует.

Проверим это на числах:

$100_{10} = 01100100_2$ – прямой код; $-100_{10} = 10011100_2$ – дополнительный код.
$75_{10} = 01001011_2$ – прямой код; $-75_{10} = 10110101_2$ – дополнительный код.

$$100 + 75 = \begin{array}{r} 01100100_{пр} \\ \underline{01001011_{пр}} \\ \mathbf{\underline{1}}0101111_{пр} \end{array} \qquad 175 > 127$$

$$(-100) + (-75) = \begin{array}{r} 10011100_{доп} \\ \underline{10110101_{доп}} \\ \mathbf{\underline{1}}\ 01010001_{доп} \end{array} \qquad -175 < -128$$

В первом случае при сложении кодов есть перенос в знаковый разряд без переноса из знакового разряда и произошло положительное переполнение. Во втором – есть перенос из разряда знака числа без переноса в знаковый разряд и результат меньше минимально представимого в этом формате числа (отрицательное переполнение).

2.10 Числа с фиксированной и плавающей точкой

В математике применяются две формы представления чисел: с *фиксированной точкой* и с *плавающей точкой*.

В числе с фиксированной точкой положение точки, отделяющей целую и дробную часть, закрепляется в определенном месте относительно разрядов числа. Сама точка не содержится в коде числа.

Первые ЭВМ предполагали точку перед старшим разрядом, т.е. кодировались только дробные числа. В современных ЭВМ точка предполагается после младшего разряда, такой формат позволяет представлять только целые числа. Самый левый бит обычно считается знаковым.

Общая формула для n-разрядного числа со знаком

$$-2^{n-1} \le N \le 2^{n-1} - 1.$$

Возможны числа без знака (только положительные). Тогда диапазон значений будет

$$0 \le N \le 2^{n} - 1.$$

Числа с фиксированной точкой в персональных компьютерах представляются в нескольких форматах и занимают целое число байт: 1, 2, 4 (рис. 2.5).

Байт (n = 8) 0 ÷ 255 (без знака),
–128 ÷ 127 (со знаком).
Слово (2 байта, n = 16) 0 ÷ 65535 (без знака),
–32768 ÷ 32767 (со знаком).
Двойное слово (4 байта, n = 32) 0 ÷ 4294967295 (без знака),
–2147483648 ÷ 2147484647 (со знаком).

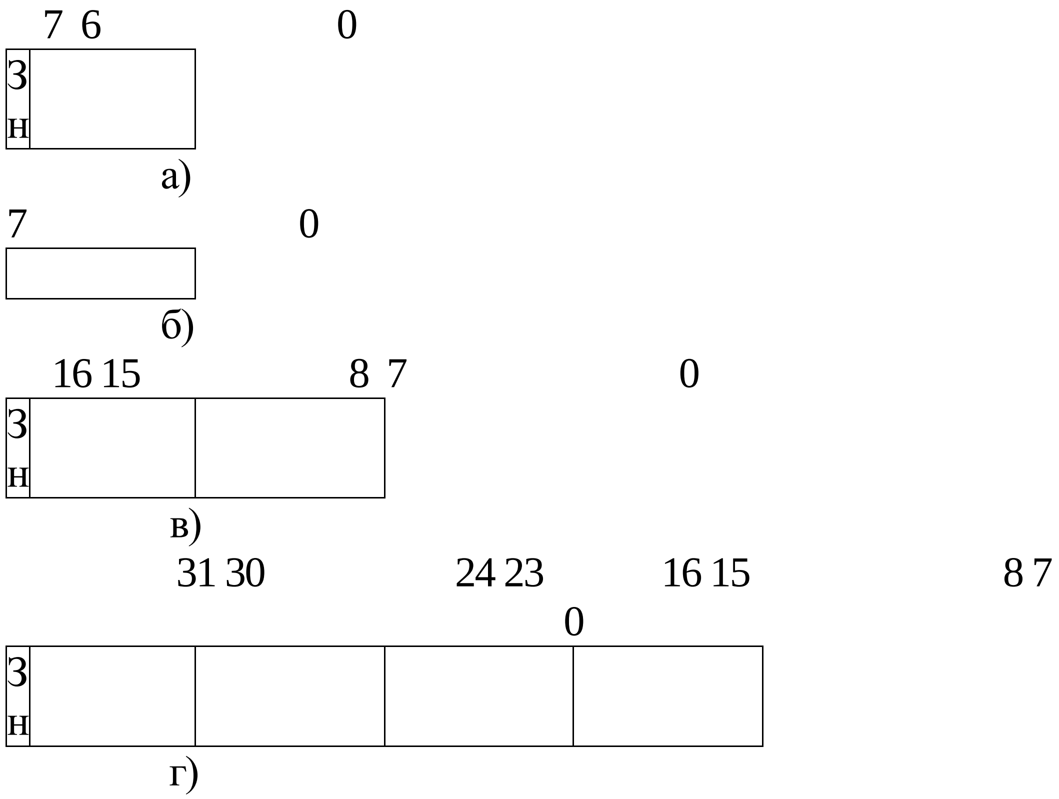


Рис. 2.5 – Представление чисел с фиксированной точкой:
а – формат байта со знаком; б – формат байта без знака;
в – формат слова со знаком; г – формат двойного слова
со знаком

Числа с плавающей точкой. Для чисел с дробной частью и реализации более широкого диапазона чисел используется научное представление числа – с плавающей точкой

$$x = q \cdot p^n,$$

где: q – мантисса числа;

p – основание системы счисления;

n – порядок числа.

Мантисса и порядок могут быть положительными и отрицательными. У числа с плавающей точкой четыре части: знак мантиссы; мантисса; знак порядка; порядок.

Например: $-1.257 \cdot 10^{-6} = -0.000001257$.

Здесь -1.257 – мантисса числа со знаком минус, 10 – основание системы счисления, -6 – порядок числа.

В ЭВМ используется P=2, мантисса и порядок кодируются в двоичной системе счисления. Мантисса обычно нормализуется, т.е. приводится к диапазону

$$\frac{1}{p} \le q < 1 \quad \text{или} \quad 1 \le q < p.$$

Арифметические действия над числами с плавающей точкой сложнее, чем с числами с фиксированной точкой.

Умножение:

сложить порядки;

перемножить мантиссы;

нормализовать мантиссу произведения;

скорректировать порядок.

Нормализация мантиссы – сдвиг ее на r разрядов влево и уменьшение порядка на r единиц (r – число нулей в старших разрядах мантиссы).

Например,

$$0.33 \cdot 10^{-3} * -0.2 \cdot 10^{2} = (0.33 * (-0.2)) \cdot 10^{(-3+2)} = -0.066 \cdot 10^{-1} =$$
$$= -0.66 \cdot 10^{((-1)+(-1))} = -0.66 \cdot 10^{-2}.$$

Деление:

из порядка делимого вычесть порядок делителя;

разделить мантиссы;

нормализовать мантиссу частного;

скорректировать порядок.

Например:

$$-0.33 \cdot 10^{-3} / 0.2 \cdot 10^{2} = -(0.33 / 0.2) \cdot 10^{((-3)-2)} = -1.65 \cdot 10^{-5} =$$

$= -0.165 \cdot 10^{(-5+1)} = -0.165 \cdot 10^{-4}$.

Сложение:

найти разность большего и меньшего порядков;

сдвинуть мантиссу меньшего числа вправо на разность порядков;

сложить мантиссы;

в случае переполнения прибавить 1 к порядку (нормализовать).

Сложим числа: $-0.2 \cdot 10^{2} + 0.33 \cdot 10^{-3}$:

а) разность порядков $2 - (-3) = 5$;

б) приведем мантиссу к старшему порядку $0.33 \cdot 10^{-3} = 0.0000033 \cdot 10^{2}$;

в) складываем мантиссы $-0.2000000 + 0.0000033 = -0.1999967$.

Ответ: $-0.1999967 \cdot 10^{2}$.

Для упрощения действий над порядками их сводят к действиям над целыми положительными числами, применяя для порядка смещенный код. Такой код называют *характеристикой* числа. К порядку n прибавляется смещение $A = 2^{k-1} - 1$, где k – число двоичных разрядов порядка. Характеристика $X = n + 2^{k-1} - 1$ всегда положительна.

Диапазон чисел с плавающей точкой зависит от основания системы счисления и количества разрядов порядка. Точность представления числа определяется количеством разрядов мантиссы. В нормализованной мантиссе старший разряд всегда 1, его можно не хранить в поле мантиссы, а предполагать.

В микропроцессорных ЭВМ используется представление чисел с плавающей точкой с двоичным основанием системы счисления. Есть три формата кодирования таких чисел (рис.2.6):

одинарной точности – 32 разряда (4 байта);

двойной точности – 64 разряда (8 байт);

расширенной точности – 80 разрядов (10 байт).

Мантисса нормализуется в пределах $1 <= q < 2$. Старший разряд мантиссы присутствует только в формате расширенной точности. Представление с расширенной точностью является внутренним для микропроцессора и используется в операндах арифметических команд с плавающей точкой.

31 30 23 22 0

Зн	Характеристика	Мантисса

а)

63 62 52 51 0

Зн	Хактеристика	Мантисса

б)

79 78 64 63 0

Зн	Характеристика	Мантисса

в)

Рис. 2.6 – Представление чисел с плавающей точкой
а – одинарной точности; б – двойной точности; в – расширенной точности

В таблице приведены параметры кодов чисел различных форматов, диапазоны возможных значений и обеспечиваемая ими точность представления мантиссы. Современные микропроцессоры обеспечивают астрономическую точность расчетов в расширенном формате.

Таблица – Параметры чисел с плавающей точкой в персональных компьютерах

Формат числа с плавающей точкой	Число двоичных разрядов		Значение порядка		Точность мантиссы		Диапазон значений числа с плавающей точкой
	Характеристика	Мантисса	Двоичный	Десятичный	Двоичная	Десятичная	
Одинарный	8	23+1	–128÷127	±38	2^{-24}	10^{-7}	$1.17\cdot10^{-38} \div 3.37\cdot10^{+38}$
Двойной	11	52+1	–1024÷1023	±308	2^{-53}	10^{-15}	$2.23\cdot10^{-308} \div 1.67\cdot10^{+308}$
Расширенный	15	64	–16384÷16383	±4932	2^{-64}	10^{-19}	$3.37\cdot10^{-4932} \div 1.2\cdot10^{+4932}$

Двоично-десятичные числа. Большинство современных ЭВМ имеют команды обработки десятичных чисел. Десятичные числа кодируются двоично-десятичным представлением, когда каждая десятичная цифра представляется четырехразрядным двоичным числом (см. 2.8), и могут быть в зонном и упакованном формате.

В упакованном формате в каждый байт записывается две соседние цифры числа (при нечетном количестве цифр слева к числу приписывается ноль). В зонном формате в каждом байте размещается только одна цифра, которая занимает младшие разряды байта, а в старшем полубайте (зоне) записывается постоянное двоичное число: нули, единицы или другой двоичный код. На рис. 2.7 приведены примеры кодирования чисел в упакованном и зонном формате.

Если в зонной части числа использовать код 0011, то зонный формат совпадает с символьным представлением числа (ANCII – числа). Микропроцессоры имеют команды для сложения и вычитания таких чисел. Десятичная арифметика удобна в программах обработки данных, когда для простых вычислений время, затраченное на преобразование чисел в двоичное представление, может быть больше, чем время арифметических операций.

цифра	цифра	цифра	цифра	цифра	цифра

0000	0101	1001	0110	0000	0111

0 5 9 6 0 7

а)

зона	цифра	зона	цифра	зона	цифра

0000	0111	0000	0011	0000	0100

7 3 4

б)

Рис. 2.11 – Кодирование десятичных чисел
а – упакованный формат; б – зонный формат

Лабораторная работа № 4
Кодирование числовой информации

Цель работы: овладение навыками кодирования числовой информации

Порядок выполнения задания:

1. Составьте блок-схему алгоритма преобразования $Z_p \rightarrow Z_1 \rightarrow Z_q$ и реализуйте его на каком-либо языке программирования.
2. Составьте блок-схему алгоритма преобразования $Z_{10} \rightarrow Z_q$, используя два способа:
- *первый способ* реализовать из соотношений:

$$\gamma_{i+1} = \gamma_i \text{ div } q \qquad b_i = \gamma_i \text{ mod } q$$

- *второй способ* реализовать из соотношений:

$$b_i = \delta_i \text{ div } q^i \qquad \delta_{i+1} = \delta_i \text{ mod } q^i$$

и реализуйте его на каком-либо языке программирования.
3. Разработайте алгоритм перевода $0,Y_{10} \rightarrow 0,Y_q$, основанный на выделении i первых цифр дроби, и реализуйте его на каком-либо языке программирования.
4. Разработайте алгоритм перевода:
а) $Z_2 \rightarrow Z_8$,
б) $Z_2 \rightarrow Z_{16}$,
в) $Z_8 \rightarrow Z_{16}$ и $Z_{16} \rightarrow Z_8$, используя промежуточный переход к двоичной системе счисления,
г) $Z_p \rightarrow Z_q$ без перехода к промежуточным системам счисления.
5. Постройте алгоритм перевода нормализованного числа из одной системы счисления в другую ($X_{10} \rightarrow X_2$) при $k_p<0$. Реализуйте его программным путем одновременно с алгоритмом для $k_p>=0$.
6. Постройте алгоритм перевода нормализованного числа из одной системы счисления в другую ($X_2 \rightarrow X_{10}$) при $k_p<0$. Реализуйте его программным путем одновременно с алгоритмом для $k_p>=0$.
7. Постройте алгоритм сложения и умножения двух целых чисел с основанием q. Реализуйте алгоритм на каком-либо языке программирования.

Состав отчета по лабораторной работе:
1. Листинги программ кодирования числовой информации (согласно номеру варианта) по пунктам 1-7.
2. Выводы по работе.

Варианты исходных данных

№	п.1	п.2	п.3	п.4	п.5	п.6	п.7
1	$37_8 \to Z_3$	$785 \to Z_5$	$953{,}25 \to Z_5$	**а-б)** 10110110101; **в)** $123_8 \to Z_{16}$; $14C_{16} \to Z_8$; **г)** $120123 \to Z$ p=3, q=9	523,25	$0{,}110001*2^{101}$	1153,2; 1147,32 q=9
2	$53_7 \to Z_5$	$860 \to Z_6$	$914{,}625 \to Z_6$	**а-б)** 11111011011; **в)** $416_8 \to Z_{16}$; $40F_{16} \to Z_8$; **г)** $AAA \to Z$ p=16, q=4	203,82	$0{,}110000*2^{100}$	40F,4; 160,4 q=16
3	$42_5 \to Z_3$	$270 \to Z_4$	$360{,}25 \to Z_4$	**а-б)** 1010010101; **в)** $335_8 \to Z_{16}$; $25E_{16} \to Z_8$; **г)** $12345678 \to Z$ p=9, q=3	286,16	$0{,}101011*2^{111}$	2023,5; 526,4 q=7
4	$26_8 \to Z_5$	$265 \to Z_3$	$712{,}375 \to Z_3$	**а-б)** 10110111111; **в)** $215_8 \to Z_{16}$; $1B1_{16} \to Z_8$; **г)** $122323 \to Z$ p=4, q=16	841,37	$0{,}110011*2^{100}$	1512,4; 1015,2 q=6
5	$55_6 \to Z_4$	$757 \to Z_7$	$300{,}375 \to Z_7$	**а-б)** 11110110111; **в)** $665_8 \to Z_{16}$; $30C_{16} \to Z_8$; **г)** $765437 \to Z$ p=9, q=3	800,31	$0{,}110101*2^{11}$	166,14; 143,2 q=8
6	$78_9 \to Z_6$	$945 \to Z_9$	$989{,}375 \to Z_9$	**а-б)** 10011110111; **в)** $251_8 \to Z_{16}$; $2CF_{16} \to Z_8$; **г)** $1EF \to Z$ p=16, q=4	269,75	$0{,}11101*2^{1000}$	12123,2; 32321,3 q=4
7	$64_7 \to Z_3$	$530 \to Z_4$	$652{,}625 \to Z_4$	**а-б)** 10110100001; **в)** $456_8 \to Z_{16}$; $1A1_{16} \to Z_8$; **г)** $210012 \to Z$ p=3, q=9	139,09	$0{,}100101*2^{10}$	251,432; 52,54 q=6
8	$77_8 \to Z_4$	$536 \to Z_3$	$725{,}625 \to Z_3$	**а-б)** 11110100111; **в)** $117_8 \to Z_{16}$; $19F_{16} \to Z_8$; **г)** $675432 \to Z$ p=9, q=3	339,25	$0{,}100100*2^{11}$	1060,52; 661,14 q=7

№	п.1	п.2	п.3	п.4	п.5	п.6	п.7
9	$73_8 \to Z_6$	$970 \to Z_7$	$673{,}5 \to Z_7$	**а-б)** 10000100101; **в)** $775_8 \to Z_{16}$; $3E0_{16} \to Z_8$; **г)** $3E7 \to Z$ p=16, q=4	160,57	$0{,}10000*2^{1111}$	1226,1; 266,7 q=8

10	$51_7 \to Z_4$	$618 \to Z_5$	$628{,}25 \to Z_5$	**а-б)** 10100111101; **в)** $711_8 \to Z_{16}$; $2D0C_{16} \to Z_8$; **г)** $765432 \to Z$ p=9, q=3	305,88	$0{,}10111*2^{1001}$	32B,D; 187,D8 q=15
11	$21_3 \to Z_7$	$772 \to Z_9$	$876{,}5 \to Z_9$	**а-б)** 10110110101; **в)** $123_8 \to Z_{16}$; $14C_{16} \to Z_8$; **г)** $120123 \to Z$ p=3, q=9	134,62	$0{,}10001*2^{101}$	FFFE; A01C q=16
12	$61_8 \to Z_9$	$284 \to Z_3$	$212{,}5 \to Z_3$	**а-б)** 10111110001; **в)** $101_8 \to Z_{16}$; $123F_{16} \to Z_8$; **г)** $123613 \to Z$ p=9, q=3	273,35	$0{,}11111*2^{1101}$	103254; 666 q=7
13	$87_9 \to Z_4$	$898 \to Z_8$	$667{,}125 \to Z_4$	**а-б)** 10111111101; **в)** $177_8 \to Z_{16}$; $F4C_{16} \to Z_8$; **г)** $FEC9 \to Z$ p=16, q=4	359,17	$0{,}10101*2^{1011}$	35A7B; 17A9 q=13
14	$78_9 \to Z_5$	$1933 \to Z_{11}$	$256{,}62 \to Z_{11}$	**а-б)** 10010101101; **в)** $736_8 \to Z_{16}$; $EA5_{16} \to Z_8$; **г)** $2210 \to Z$ p=3, q=9	99,7	$0{,}10100*2^{111}$	1210121; 1002101 q=3
15	$66_7 \to Z_9$	$339 \to Z_4$	$737{,}265 \to Z_4$	**а-б)** 11011001101; **в)** $601_8 \to Z_{16}$; $1111_{16} \to Z_8$; **г)** $13AC9 \to Z$ p=16, q=4	568,66	$0{,}11110*2^{1010}$	442301; 334210; q=5;
16	$54_9 \to Z_6$	$1949 \to Z_{12}$	$413{,}562 \to Z_{12}$	**а-б)** 10000110111; **в)** $107_8 \to Z_{16}$; $13FA1_{16} \to Z_8$; **г)** $FAF0 \to Z$ p=16, q=4	1345,2	$0{,}10101*2^{1101}$	AABA; BACC; q=14
№	п.1	п.2	п.3	п.4	п.5	п.6	п.7
17	$36_7 \to Z_2$	$744 \to Z_7$	$1005{,}375 \to Z_7$	**а-б)** 11110011101; **в)** $105_8 \to Z_{16}$; $AAA_{16} \to Z_8$; **г)** $22010 \to Z$ p=3, q=9	458,01	$0{,}101*2^{10011}$	101001; 110101; q=2
18	$65_9 \to Z_5$	$2700 \to Z_{13}$	$619{,}25 \to Z_{13}$	**а-б)** 10001100101; **в)** $166_8 \to Z_{16}$; $AE3F_{16} \to Z_8$; **г)** $78654 \to Z$ p=9, q=3	699,22	$0{,}1010*2^{10010}$	FFFFF; EEEEE; q=16
19	$43_5 \to Z_3$	$5453 \to Z_{14}$	$667{,}25 \to Z_{14}$	**а-б)** 10010111001; **в)** $737_8 \to Z_{16}$; $ABCD_{16} \to Z_8$; **г)** $137653 \to Z$ p=9, q=3	177,39	$0{,}10*2^{1001010}$	137852; 368858; q=9
20	$25_6 \to Z_3$	$5630 \to Z_{15}$	$339{,}25 \to Z_{15}$	**а-б)**	1001,5	$0{,}1111*2^{11111}$	553102;

				10110010111; **в)** $176_8 \to Z_{16}$; $EE21F_{16} \to Z_8$; **г)** $330312 \to Z$ p=4, q=16			221531; q=6

2. 2. Кодирование. Пропускная способность канала

2.2.1. Основные определения. Пропускная способность канала.

Модель системы передачи информации. Системой передачи информации (СПИ) называется совокупность технических средств, используемых для передачи информации в пространстве (связь, телекоммуникации). В сущности, те же закономерности характерны и для системы передачи информации во времени (хранение информации). Модель системы передачи информации представлена на рис.

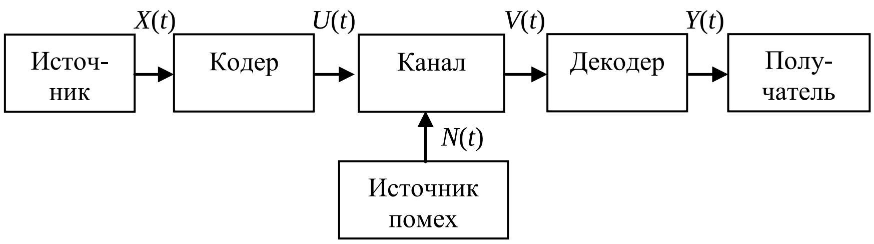


Рис.

Источником называется устройство или человек, генерирующие сообщение, подлежащее передаче, а *получателем* – устройство или человек, получающие информацию. *Каналом* называется совокупность физических объектов, используемых для переноса сигнала (кабель, проводная линия, лента магнитофона, среда, в которой распространяются волны, и т.п.). *Кодером* называется устройство, преобразующее сигнал на выходе источника в сигнал, пригодный для передачи по данному каналу. *Декодером* называется устройство, преобразующее сигнал на выходе канала к виду, пригодному для использования потребителем. Воздействие помех (шумов) на сигнал в процессе его распространения в канале отражено на схеме введением *источника помех*.

Источник информации. Источник информации называется дискретным или непрерывным в зависимости от вида созда-ваемых им сообщений. Источник называется стационарным, если сообщение на его выходе есть стационарная случайная функция (последовательность). Дискретный источник называ-ется источником без памяти, если символы в последова-тельности на его выходе независимы.

Канал. Канал называется дискретным, непрерывным, дискретно-непрерывным или непрерывно-дискретным в зависимости от вида сигналов на его входе и выходе. Если входной $A(t)$ и выходной $B(t)$ сигналы связаны взаимно-однозначно, то такой канал называется каналом без шума. В канале с шумом возможны случайные ошибки при преобразовании входного сигнала в выходной.

Дискретный канал с шумом называется каналом без памяти, если ошибки в отдельных символах выходной последователь-ности статистически независимы. Канал называется стационарным (постоянным), если условные вероятности перехода от $A(t)$ к $B(t)$ не зависят от начала отсчёта.

Кодер и декодер. Кодирующее устройство выполняет следующие операции:

а) согласование источника с каналом (перевод реальных сообщений в электрические сигналы, модуляция непрерывных сигналов, квантование непрерывных сообщений, представление s-ичного дискретного сообщения в m-ичном коде и т.п.);

б) экономное представление информации с минимальной избыточностью либо, наоборот, разумное введение избыточности в сигнал, передаваемый по каналу, с целью повышения его помехоустойчивости.

Функции декодера в значительной степени обратны функциям кодера. Кроме того, введение при кодировании избыточности в сигнал $A(t)$ часто представляет возможность обнаруживать и исправлять ошибки, возникающие в канале из-за влияния помех. Эта операция также выполняется декодирующим устройством.

Скорость передачи информации. Одной из важных характеристик источника является скорость создания информации. Для стационарного дискретного источника она равна энтропии последовательности символов на его выходе, вычисленной по формуле. Скорость создания информации непрерывным источником, вообще говоря, бесконечна. Поэтому в качестве скорости создания информации стационарным непрерывным источником принимают значение ε-энтропии непрерывной функции на его выходе, вычисленной при заданной точности воспроизведения этой функции.

Важнейшей характеристикой СПИ в целом является скорость передачи информации. Скоростью передачи информации называется средняя величина взаимной информации (в единицу времени или на отсчёт) между сигналом $X(t)$ на выходе источника и сигналом $Y(t)$, поступающим к потребителю.

Пропускная способность канала. Скорость передачи информации зависит в значительной степени от скорости её создания, способов кодирования и декодирования. Наибольшая возможная в данном канале скорость передачи информации называется его пропускной способностью C. Пропускная способность канала, по определению, есть скорость передачи информации при использовании «наилучших» для данного канала источника, кодера и декодера, поэтому она характеризует только канал.

Пропускная способность дискретного (цифрового) канала без помех

$$C = \log m \quad \text{бит/символ}, \qquad 4.1.1)$$

где m – основание кода сигнала, используемого в канале (см. 4.2.2). Скорость передачи информации в дискретном канале без шумов равна его пропускной способности, когда символы в канале независимы, а все m букв алфавита равновероятны (используются одинаково часто).

Пропускная способность дискретного канала с шумом без памяти

$$C = \max_{p(A)} \sum_{j=1}^{m} \sum_{k=1}^{m} p(a_j)\, p(b_k / a_j) \log \frac{p(b_k / a_j)}{p(a_j)} \qquad (4.1.2)$$

бит/симв.

При заданных вероятностях перехода $p(b_k / a_j)$ задача вычисления пропускной способности сводится к отысканию «наилучшего» распределения вероятностей $p(a_1),..., p(a_m)$ на входе канала. Скорость передачи информации в таком канале равна его пропускной способности, когда символы в канале независимы, а распределение вероятностей m входных букв равно найденному.

2.2.2. Задача кодирования информации

Ранее указывалось, что источник сообщения включает кодирующую систему, формирующую сигналы по известным получателю правилам. Ввиду независимости содержания сообщения от выбранной формы его представления возможно преобразование одного кода в другой с предоставлением правила обратного преобразования получателю сообщения. Целесообразность такого дополнительного кодирования сообщения на передающей стороне и соответствующего декодирования на приемной стороне возникает из-за избыточности алфавита сообщения и искажения сигналов действующими в канале связи помехами. Кодирование предшествует хранению и передаче информации.

Реализация основных характеристик канала связи, помимо разработки технических устройств, требует решения информационных задач - выбора оптимального метода кодирования.

Основными задачами кодирования являются:

1. Обеспечение экономичности передачи информации посредством устранения избыточности.

2. Обеспечение надежности (помехоустойчивости) передачи информации.

3. Согласование скорости передачи информации с пропускной способностью канала.

Соответствие между элементами дискретных сообщений и видом кодирования обеспечивается выбором:

1. Длительности сигналов.

2. Длины кодового слова.

3. Алфавита знаков и способа кодирования (побуквенного, блочного).

Сформулируем задачу кодирования.

Полагаем, что сообщение источника информации формируется из знаков a_i, $i=1, 2, ..N^a$ *внешнего (входного, первичного)* алфавита A объемом N^a. Сообщения представляют собой *слова,* образованные последовательностью n_r знаков: $A_r=a_1a_2...a_{nr}$. В кодирующем устройстве слово A_r преобразуется в *кодовое слово* $B_r=b_1b_2...b_{mr}$, составленное из m_r знаков b_j, $j=1, 2, \ldots N^b$ *внутреннего (выходного, вторичного)* алфавита B. Число знаков кодового алфавита называют *основанием кода.* Число знаков в кодовом слове называют *длиной кодового слова.* Отображение G множества слов в алфавите *A* на множество слов в алфавите B называют *кодирующим отображением* или *кодом.* Применение кодирующего отображения G к любому слову из входного алфавита называется *кодированием.* То есть код - это правило отображения знаков одного алфавита в знаки другого алфавита, кодирование - это преобразование одной формы сообщения в другую посредством указанного кода.

Различают *побуквенное* и *блочное* кодирование. При *побуквенном* кодировании каждому знаку внешнего алфавита ставится в соответствие кодовое слово из знаков внутреннего алфавита. При *блочном* кодировании слову из знаков внешнего алфавита ставится в соответствие кодовое слово из знаков внутреннего алфавита.

Слова из знаков внутреннего алфавита *В,* сопоставленные со словами из знаков внешнего алфавита *А* по правилу *G,* называются *кодовыми комбинациями.* Если $A_r \in A$ и $G(A_r) = B_r$, то говорят, что слову A_r соответствует кодовая комбинация B_r. Совокупность кодовых комбинаций, используемых для передач заданного, количества дискретных сообщений, называют *кодовым словарем.*

Процесс, обратный кодированию, заключается в восстановлении из кодовой комбинации $B_r=b_1b_2...b_{mr}$ слова $A_r=a_1a_2...a_{nr}$ из входного алфавита и называется *декодированием.* Если процесс кодирования осуществляется с использованием правила *G,* то процесс декодирования основан на применении правила G^{-1}, где G^{-1} есть отображение, обратное отображению G.

Операции кодирования и декодирования называют *обратимыми,* если их последовательное применение обеспечивает возврат к исходной форме сообщения без потери информации.

Пусть A_r - слово в алфавите А и $B_r = G(A_r)$ - слово в алфавите В. Код называется *обратимым,* если для любого слова $B_r = G(A_r)$ в алфавите *В* существует единственное преобразование $G^{-1}(B_r) = A_r$. То есть по слову B_r в алфавите В всегда однозначно восстанавливается слово A_r в алфавите А, **из** которого было образовано слово B_r.

Примером обратимого кодирование является представление знаков в телеграфном коде при передаче сообщений и восстановление их при приеме. Примером необратимого кодирования является перевод текста с одного естественного языка на другой. (Обратный перевод побуквенно обычно не соответствует исходному тексту.)

Чтобы код был обратимым, необходимо:

1) чтобы разным символам входного алфавита *А* были сопоставлены разные кодовые комбинации;

2) чтобы никакая кодовая комбинация не составляла начальной части какой-нибудь другой кодовой комбинации.

Наиболее простым правилом кодирования является сопоставление каждому символу входного алфавита *A* слова конечной длины в выходном алфавите *B*. Код может быть задан в форме таблицы, графа, аналитического выражения, то есть в тех же формах, что и отображение.

Пример. Цифры (0...9) являются знаками входного алфавита и могут кодироваться словами из знаков выходного двоичного алфавита. Бывает удобным при их обработке кодировать цифры так, чтобы двоичные слова минимально отличались друг от друга, например, лишь в одном бите. Коды, удовлетворяющие этому условию, называют **кодами Грея,** или **одношаговыми кодами**

0 1 2 3 4 5 6 7 8 9

(0000, 0001, 0011, 0010, 0110, 0111, 0101, 0100, 1100, 1101).

Пример. Декодировать сообщение bbaababba, закодированное в соответствии с приведенной кодовой таблицей:

x	y	z
a	ba	bba

Решение. Анализ таблицы показывает, что код удовлетворяет обоим свойствам обратимости, следовательно, сообщение может быть декодировано единственным образом:

bba a ba a bba

z x y x z

Критерий оценки эффективности кода

Рассмотрим случай формирования сообщений A_r, r=1, 2, … N^A из алфавита A объемом N^A Обозначим $p^A{}_r$ вероятность появления сообщения A_r Энтропия сообщений

$$H^A = -\sum_{i=1}^{N^A} p^A{}_i \log_2 p^A{}_i$$

Формируемые кодером кодовые комбинации $B_r = b_1 b_2 \ldots b_{mr}$

длиной m_r, состоят из знаков b_j, $j=1, 2,\ldots N^b$, вероятность появления которых $p^b{}_j$. Энтропия знаков равна

$$H^B{}_1 = -\sum_{i=1}^{N^b} p^b{}_i \log_2 p^b{}_i$$

Средняя длина кодовой комбинации на выходе кодера равна:

$$m_{cp} = \sum_{r=1}^{N^A} m_r p^A{}_r$$

Энтропия кодовых комбинаций на выходе кодера $H^B = m_{cp} \cdot H^b{}_1$.

Для передачи сообщений без потери информации необходимо выполнение условия

$$H^B = m_{cp} \cdot H^b{}_1 \geq H^A,$$

из которого следует необходимость выполнения условия для средней длины кодовой комбинации

$$m_{cp} \geq \frac{H^A}{H^b{}_1}.$$

Максимальная энтропия знаков алфавита В мощностью N^b достигается при $p^b{}_j = \frac{1}{N^b},\ j = 1, 2, \ldots N^b$. Тогда энтропия одного знака кодовой комбинации

$$H^b{}_{1\ \max} = \sum_{i=1}^{N^b} \left(\frac{1}{N^b}\right) \cdot \log_2 N^b = \log_2 N^b$$

Ввиду этого максимальная энтропия кодовых комбинаций передаваемых сообщений равна $H^B{}_{\max} = m_{cp} \cdot \log_2 N^b$

а минимально достижимая средняя длина кодовых комбинаций на выходе кодера равна

$$m_{\min} = H^A / \log_2 N^b$$

При $H^B{}_{\max} > H^A$ закодированное сообщение обладает избыточностью информации, из чего следует, что для кодирования используется больше знаков, чем это минимально необходимо. Для количественной оценки избыточности сообщения используют коэффициент избыточности К, определяемый по формуле:

$$K = \frac{H^B{}_{\max} - H^A}{H^B{}_{\max}} = \frac{m_{cp} \cdot \log_2 N^b - m_{\min} \cdot \log_2 N^b}{m_{cp} \cdot \log_2 N^b} = \frac{m_{cp} - m_{\min}}{m_{cp}}.$$

Под эффективным кодом понимается такой код, коэффициент избыточности которого равен нулю. На основании формулы (2.4.1) заключаем, что для эффективных кодов должно выполняться $m_{cp} = m_{min}$. При этом неравенство $H^B{}_{\max} > H^A$ должно перейти в

$$\sum_{r=1}^{N^A} m_{cp} \cdot p^A{}_r \cdot \log_2 N^b = -\sum_{r=1}^{N^A} p^A{}_r \cdot \log_2 p^A{}_r .$$

из которого следует

$$\sum_{r=1}^{N^A} p^A{}_i (m_{cp} \cdot \log_2 N^b + \log_2 p^A{}_r) = 0.$$

Предполагая, что множество A_r не содержит элементов, появление которых невозможно, приходим к заключению, что для выполнения (2.4.2) необходимо выполнение условия:

$$m_{cp} = -\frac{\log_2 p^A{}_r}{\log_2 N^b}$$ для всех $r=1, \ldots, N^A$.

Полученное отношение не всегда дает целочисленный результат и, следовательно, не для любого из набора сообщений A_r, $r=1,2,\ldots N^A$ с заданным распределением вероятностей $\{p^A{}_r\}$ можно построить эффективный код в смысле равенства нулю коэффициента избыточности К. Тем не менее всегда можно обеспечить выполнение неравенства

$$-\frac{\log_2 p^A{}_r}{\log_2 N^b} \le m_{cp} < -\frac{\log_2 p^A{}_r}{\log_2 N^b} + 1.$$

Умножив обе части неравенства на $p^A{}_r$ и суммируя по r, получим:

$$\frac{H^A}{\log_2 N^b} \le m_{cp} < \frac{H^A}{\log_2 N^b} + 1.$$

Полученное неравенство может служить критерием оценки эффективности реального кода.

Пусть сообщение источника содержит m_a знаков из алфавита A объемом N^a, и средняя информация на знак этого сообщения равна I_a. В результате кодирования формируется сообщение, содержащее m_b знаков из алфавита B объемом N^b, и средняя информация на знак этого сообщения равна I_b. Условие отсутствия потери информации при преобразовании:

$$m_a \cdot I^a \le m_b \cdot I^b$$

Из этого неравенства следует необходимость выполнения условия

$$I^A \le (m_b / m_a) I^b \quad \text{или} \quad m_b / m_a = K(A,B) \ge I^a / I^b$$

Величина $m_b / m_a = K(A,B)$, называемая *длиной кода,* характеризует среднее количество знаков закодированного сообщения, приходящееся на один знак исходного сообщения. Обычно для улучшения различимости знаков при приеме сигналов стремятся уменьшить количество знаков выходного алфавита B до N^b=2..4, тогда как число знаков исходного сообщения, как правило, велико (N^a ~ 150 при передаче символьной информации). Поэтому N^a>N^b и I^a> I^b, вследствие чего длина кода K(A, B) >1, то есть один знак входного сообщения, представляется несколькими знаками закодированного сообщения.

Эффективное кодирование должно обеспечить минимальную длину внутреннего кода для сокращения объема требуемой памяти запоминающего устройства и времени передачи сообщения (стоимости запоминающего устройства и времени передачи на единицу информации). Оценочное выражение, устанавливающее нижний предел длины кода, имеет вид:

$$K_{\min}(A,B) = I^a / I^b$$

Это выражение показывает два пути сокращения длины кода:

для уменьшения числителя следует уменьшать избыточность сообщения, учитывая различие частоты знаков и их сочетаний в сообщении $(I_{a0} > I_{a1} .. > I_{\infty})$

для увеличения знаменателя следует применять способ кодирования, обеспечивающий равные вероятности знаков внутреннего алфавита ($I_{b\max} = \log_2 N_b$).

Таким образом, с точки зрения экономичности критерием оптимальности кодирования является минимума длины кода, сокращение его до K_{min}.

Источники сообщений, при кодировании которых не учитывается вероятность появления сочетаний знаков внешнего алфавита, называют *источниками без памяти.*

Для источника без памяти наименьшая длина кода составляет $K_{\min}(A,B) = I_{a1} / \log_2 N^b$. В случае двоичного внутреннего алфавита среднее количество двоичных символов на один символ сообщения равно энтропии источника сообщений $K_{\min}(A,2) = I_{a1}$.

Относительная избыточность используемого кода определяется выражением:

$$R(A,B) = (K(A,B) - K_{\min}(A,B)) / K_{\min}(A,B) = K(A,B) / K_{\min}(A,B) - 1 = K(A,B) \cdot I^b / I^a - 1.$$

Для двоичных каналов передачи сообщений источников без памяти избыточность используемого кода равна $R(A,B) = K(A,B) / I_{a1} - 1.$

Первая теорема Шеннона

Выражение для нижнего предела длины кода не позволяет получить оценку возможного приближения к нему при реализации кодирования. Впервые теоретическое исследование решения проблемы устранения избыточности (эффективного кодирования) предпринял К.Шеннон. Для идеального канала связи (без помех) он доказал одно из основных положений теории информации.

В отсутствие помех и ограничений на задержку сообщения при любой "статистике знаков источника существует код, позволяющий получить среднее число кодовых символов на элемент сообщения, сколь угодно близкое к отношению средних информации на знак входного и выходного алфавитов.

Оптимальное кодирование для дискретного канала без помех сводится к предельному увеличению объема внутреннего алфавита канала, каждый символ (кодовая комбинация) которого соответствует некоторой отличной от других последовательности знаков внешнего алфавита. С увеличением длины кодируемых последовательностей уменьшается их вероятностная взаимосвязь, а значит, сокращается избыточность сообщений. Однако это связано с задержкой поступления сообщения получателю на *2*T+d*, где T - длительность кодируемой последовательности знаков внешнего алфавита, d - время выполнения технических операций и распространения сигналов в канале. Кодирование может начаться только после получения кодируемой последовательности длительностью T, декодирование может быть произведено только после принятия всей последовательности длительностью T.

Доказанная Шенноном теорема утверждает принципиальную возможность создания оптимального кода, но не указывает метод решения задачи.

Методы построения оптимальных кодов

Экономичное кодирование

В общем случае избыточность является полезным свойством сообщений, так как повышает помехоустойчивость. Благодаря ей, например, речь естественного языка достаточно устойчива к различного рода помехам и искажениям. Если в предложении текста исказить одну - две буквы, то, как правило, смысл фразы сохраняется.

Однако избыточность сообщения увеличивает время их передачи по каналам связи и требуемый объем памяти при хранении. Большая избыточность сообщений не всегда оправдана требованиями помехоустойчивости передачи и хранения информации. В таких случаях возникает задача устранения избыточности сообщений, получившая название *эффективного кодирования.*

Для передачи дискретных сообщений в системах связи используют, как правило, два алфавита. Один алфавит - *внешний*, имеющий достаточно большой объем, применяется для формирования сообщения на языке источника информации и адресата. Другой алфавит - *внутренний* - используется непосредственно для передачи информации по каналу связи. Он содержит небольшое количество символов, например, 0 и 1. (Словам, составленным из 0 и 1, при передаче могут соответствовать последовательности импульсов тока.)

Статистика символов внутреннего алфавита и, следовательно, их информационная нагрузка зависят от способа кодирования сообщения в системе. Разработка методов, позволяющих уменьшить избыточность сообщений, закодированных символами внутреннего алфавита, описана в большом количестве публикаций.

Очевидно, что для уменьшения избыточности кодовых слов над внутренним алфавитом следует выбирать возможно более короткие кодовые комбинации» Однако для полного устранения избыточности этого недостаточно. Как ранее было показано, необходимо учитывать в сообщениях вероятность появления и сочетаний знаков внешнего алфавита.

Равномерные коды

Каждая сформированная кодовая комбинация состоит из некоторого количества знаков, которые определяют ее размер.

Код называют *равномерным,* если все кодовые комбинации содержат одинаковое количество знаков равной длительности, а, значит, все комбинации имеют одинаковую длину. Основание кода определяется выбранной системой счисления. Оно определяет объем алфавита-множества знаков кода, посредством которых образуются кодовые комбинации. Коды, у которых основание равно двум, называют *двоичными,* или *бинарными.* При двоичном кодировании с постоянной длиной слов сообщение передает последовательность двоичных знаков (без стыков).

Примерами равномерного кода являются двоичное кодирование символов (буквы, цифр и других знаков), *телеграфный код Бодо* (предложен для передачи 56 символов 5-значным двоичным кодом. Использует дополнительную кодовую комбинацию (регистр), предшествующую знаковой *комбинации*) и код Трисиме, в котором знакам латинского алфавита ставятся в соответствие кодовые комбинации из одинакового количества 3-х знаков из алфавита объемом 3 знака: «I», «2», «З» (рис. 2.4.3).

A	111	J	211	S	311
B	112	K	212	T	321
C	113	L	213	U	313
D	121	M	221	V	321
E	122	N	222	W	322
F	123	O	223	X	323
G	131	P	231	Y	331
H	132	Q	232	Z	332

I 133	R 233	. 333

Рис. 2.4.3. Код Трисиме

Простые коды

Если дискретное сообщение образовано знаками входного алфавита объемом N, то для его передачи m знаками выходного алфавита объемом n необходимо $N_k = n^m > N$ кодовых комбинаций. Если $N_k = N$, то используются все комбинации и код является безизбыточным. Однако обычно точно такое равенство не выполняется ввиду некратности N степеням выбранной системы счисления. Но обязательно должно быть выполнено условие $N_k > N$, в результате чего $N_k – N$ кодовых комбинаций не используется.

Код, для которого выполняется условие $n^{m-1} < N < n^m$, называют *простым,* или *первичным,* кодом. Основное назначение первичных кодов - экономно представлять кодовыми комбинациями заданное множество дискретных величин, обычно счетное множество цифровых величин.

Простые коды подразделяют на *взвешенные (позиционные)*,старшим разрядам которых присваивается больший вес, чем младшим ($2^{m-1},2^{m-2},...,2^1,2^0$), и *невзвешенные (непозиционные)*, всем разрядам которых присваивают одинаковый вес.

Примером взвешенного когда является обычный двоичный код с весами знаков $2^{m-1},2^{m-2},...,2^1,2^0$. Примером невзвешенного кода является код Грея.

Взвешенный двоичный код прост в реализации и удобен при выполнении вычислительных операций в ЭВМ. Но ошибка в старшем разряде искажает сообщение более значимо, чем ошибка в младшем разряде.

В невзвешенных кодах ошибка в любом разряде имеет одинаковое значение – будет искажена вся кодовая комбинация. Перед обработкой невзвешенный код преобразуют в обычный двоичный код. Недостатком невзвешенных кодов является сложность декодирования и вычислений.

Двоично-десятичные коды

В некоторых случаях для представления чисел в памяти ЭВМ используется смешанная двоично-десятичная «система счисления», где для хранения каждого десятичного знака нужен полубайт (4 бита). Число, которое необходимо представить в двоично-десятичном коде, кодируется последовательно цифра за цифрой. Каждая из десятичных цифр от 0 до 9 представляет собой кодовую комбинацию, состоящую из четырех двоичных знаков - тетрад от 0000 до 1001. Например, число 1975, выраженное в двоично-десятичном коде, представляет собой последовательность тетрад 0001 1001 0111 0101. Упакованный десятичный формат, предназначенный для хранения целых чисел с 18-ю значащими цифрами и занимающий в памяти 10 байт (старший из которых знаковый), использует именно этот вариант. Кодирование десятичных чисел тетрадами уступает по экономичности кодированию обычным двоичным кодом. Для десятичного числа, содержащего m цифр при двоично-десятичном кодировании, требуется 4m двоичных знаков, а при обычном двоичном кодировании $m \cdot \log_2 10 \sim 3{,}3 \cdot m$ двоичных знаков.

Для однозначного декодирования равномерных кодов достаточно выполнение первого условия обратимости кодов. Их техническая реализация достаточно проста, и поэтому они получили широкое распространение. Однако они не учитывают различие вероятности появления знаков в сообщениях и не обеспечивают максимальную эффективность кодирования.

Неравномерные коды

Одним из первых кодов, учитывающих вероятность знаков в сообщении, является код Морзе, разработанный в 1938 году задолго до исследований относительной частоты появления различных букв в текстах. В этом коде каждой букве и цифре сопоставлена оригинальная последовательность кратковременных импульсов - точек и тире, разделенных паузами. Буквам, используемым чаще, присвоены короткие кодовые комбинации, редко используемым буквам - длинные. Морзе оценил относительную частоту букв английского языка подсчетом литер в ячейках типографской наборной машины. Наиболее часто используемой букве «Е» он присвоил наиболее короткий код «точка». Следующей по количеству литер букве он присвоил код несколько большей длительности и так далее. При составлении кода Морзе для букв русского алфавита учет относительной частоты букв не производился, и это повысило его избыточность.

Расчеты избыточности кода Морзе на основании проведенных исследований частоты появления букв показали, что для букв английского алфавита она составляет 19%, для букв русского алфавита 22%.

А	·–	Ж	···–	М	––	Т	–	Ш	––––	Я	·–·–	4	····–
Б	–···	З	––·	Н	–·	У	··–	Щ	––·–			5	·····
В	·––	И	··	О	–––	Ф	··–·	Ъ,Ь	–··–	0	–––––	6	–····
Г	––·	Й	·–––	П	·––	Х	····	Ы	–·––	1	·––––	7	––···
Д	–··	К	–·–	Р	·–·	Ц	–·–·	Э	··–··	2	··–––	8	–––·
Е	·	Л	·–··	С	···	Ч	–––·	Ю	··––	3	···––	9	–––––·

При неравномерном кодировании длина кодов и соответственно продолжительность передачи кодовой комбинации могут значительно отличаться. В этом случае задача оптимального кодирования состоит в выборе такой системы кодирования, при которой минимальна суммарная длительность сообщения. Ввиду этого подход к выбору кода заключается в том, чтобы более вероятным значениям знаков в сообщении присвоить более короткие кодовые комбинации, чем менее вероятным.

Схема двоичного кодирования текстов по Р. Фано

Предложенная американским специалистом Р. Фано схема двоичного кодирования сводится к выполнению следующих операций.

1) Составить список букв алфавита (исходное множество букв) в порядке убывания значений соответствующих им вероятностей.

2) Разбить этот список на два подсписка (подмножества букв) таким образом, чтобы значения вероятностей того, что наугад взятая **из** рассматриваемого текста буква окажется в первом или во втором из этих подмножеств, были бы по возможности близки.

3) Приписать произвольному одному из этих подмножеств (подсписков символ «0», а другому «1»).

4) Рассматривая каждое из этих подмножеств (подсписков) как исходное, применительно к каждому из них осуществить операции, указанные в пунктах (2) и (3).

5) Этот процесс продолжать до тех пор, пока в каждом из

очередных подмножеств не окажется по одной букве.

6) Каждой букве приписать двоичный код, состоящий из последовательности нулей и единиц, встречающихся на пути из исходного множества букв ко множеству, состоящему из одной этой буквы.

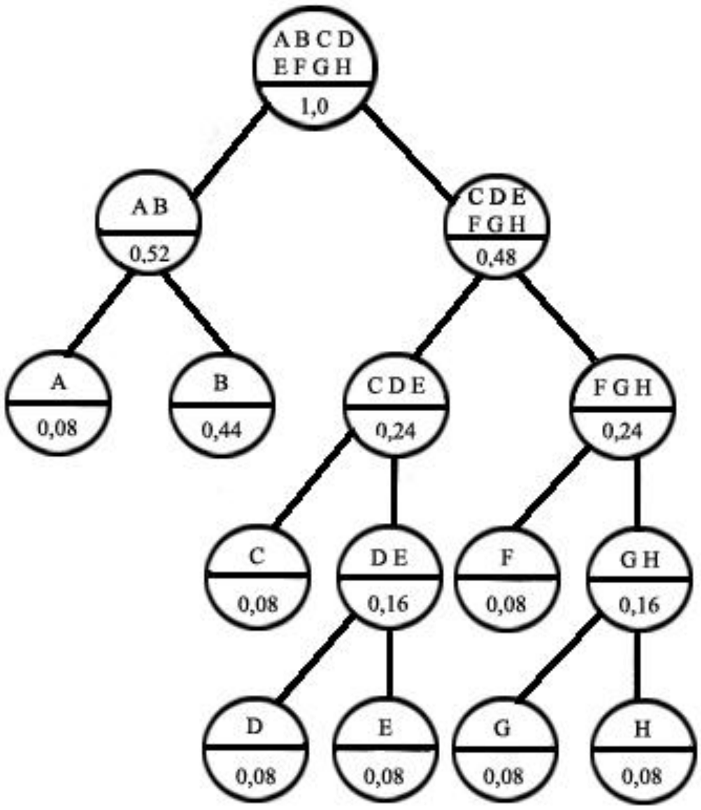
Пользуясь схемой Р. Фано (см. рис. 2.1) применительно к приведенному выше примеру, легко установить наборы двоичных символов, соответствующие буквам исходного текста:

Буква	Двоичный код	Буква	Двоичный код
A	00	E	1011
B	01	F	110
C	100	G	1110
D	1010	H	1111

Если обозначить через L_A=2, L_B =3, L_C=4,... числа двоичных символов в кодовых наборах, соответствующих буквам A, B, C, ..., то среднее число двоичных символов, отводимых под одну букву исходного алфавита, можно определить подформуле

$l=p(A)l_A+ p(B)l_B+\ldots+p(H)l_H=2{,}8$

Таким образом, с переходом к переменной длине кодовых наборов, отводимых под каждую букву исходного текста, удается



на 7% - 2,8 вместо трех символов на одну букву) сократить число двоичных символов в закодированном тексте. Правда, это связано с некоторым усложнением процедур кодирования и декодирования. Будучи достаточно эффективной, схема кодирования. Р. Фано не всегда гарантирует, что при заданном наборе значений вероятностей средняя длина кодовых наборов l окажется наименьшей возможной. Такую гарантию дает другая схема кодирования, предложенная американским математиком Д. Хаффмэном. Исходные соображения здесь те же, что и при рассмотрении схемы Р. Фано, тэднако, оперируя более тонким механизмом кодирования, Д. Хаффмэну удалось достичь наименьшего возможного при побуквенном кодировании значения средней длины кодовых наборов.

Схема двоичного кодирования текстов по Д. Хаффмэну

Предложенная Д. Хаффмэном схема кодирования заключается в следующем (см. рис. 2.2).

Формируется первый (k = 1) столбик, где все буквы алфавита записываются в порядке убывания значений вероятностей их встречаемости в исходном, подлежащем кодированию тексте. Здесь же, напротив каждой буквы, пишется соответствующее ей значение вероятности. Две буквы, занявшие в столбике предпоследнюю и последнюю позиции, с левой стороны снизу отмечаются двоичными символами соответственно "0" и "1". На рис. 2.2 эти буквы отделены от других пунктирными линиями.

2) При известном k - м столбике строится (k + 1)-й столбик по тому же принципу, что и предыдущий, с той лишь разницей, что буквы, отмеченные в предыдущем столбике двоичными символами, в последующем столбике отсутствуют. В новом столбике их "представляет" одна составная буква со значением вероятности, равным сумме вероятностей слагаемых букв.

3) Этот процесс продолжается до тех пор, пока в очередном столбике под номером k = n (число букв в алфавите) не окажется одна единственная составная буква, "представляющая" весь алфавит исходного текста со значением вероятности, равным единице. Этот последний столбик выполняет лишь контрольную функцию.

4) Для определения кодового набора, соответствующего интересующей нас букве, поступаем следующим образом.

Начиная с последнего столбика, переходя поочередно к предыдущим столбикам, рассматриваем каждый раз только те буквы, которые занимают в очередных столбиках последние два места (участки ниже пунктирных линий). Если в указанном участке очередного столбика оказывается интересующая нас буква или какая-либо составная буква, включающая эту букву, то в очередном разряде се кодового набора записываем двоичный символ, которым отмечена эта буква или включающая ее составная буква. Этот процесс продолжаем до тех пор, пока в очередном столбике, ниже пунктирной линии, не встретится интересующая нас буква в сольном варианте (не в составе какой-либо составной буквы).

Проследим, например, за получением кодового набора, соответствующего букве E. При движении от столбика с номером k = 8 к столбику с номером k = 7, в нем ниже пунктирной линии обнаруживаем составную букву $_0$(ACDEFGH), содержащую букву E. Это дает основание для записи в первом разряде кодового набора буквы E символа "0". которым отмечена составная буква $_0$(ACDEFGH). В следующий раз ниже пунктирной линии буква E встречается в составной букве $_0$(EFGH) в столбике под номером k = 6, поэтому во второй позиции кодового набора буквы E записываем символ "0". Ниже пунктирной линии буква E встречается также в пятом столбике, в составной букве $_1$(EF), что дает основание для того, чтобы в третьей позиции буквы E записать символ "1".

Далее буква $_0$E, ниже пунктирной линии и уже в сольном варианте встречается во втором столбике. Исходя из этого, четвертую позицию кодового набора заполняем символом "0" и на этом останавливаемся, т.е. в итоге букве E приписываем кодовый набор 0010. Поступая аналогичным образом, получим кодовые наборы, соответствующие остальным буквам алфавита:

k = 1	
B	0,44
A	0,08
C	0,08
D	0,08
E	0,08
F	0,08
$_0$G	0,08
$_1$H	0,08

k = 2	
B	0,44
(GH)	0,16
A	0,08
C	0,08
D	0,08
$_0$E	0,08
$_1$F	0,08

k = 3	
B	0,44
(GH)	0,16
(EF)	0,16
A	0,08
$_0$C	0,08
$_1$D	0,08

k = 4	
B	0,44
(GH)	0,16
(EF)	0,16
$_0$(CD)	0,16
$_1$A	0,08

k = 5	
B	0,44
(ACD)	0,24
$_0$(GH)	0,16
$_1$(EF)	0,16

k = 6	
B	0,44
0(EFGH)	0,32
1(ACD)	0,24

k = 7	
$_0$(ACDEFG)	0,56
$_1$B	0,44

k = 8	
(ABCDEFG)	1,00

Буква	Двоичный код	Буква	Двоичный код
A	011	E	0010
B	1	F	0011
C	0100	G	0000
D	0101	H	001

Среднее число двоичных символов, отводимых под одну букву исходного алфавита, здесь получается равным l = 2,6 = l_{min} (против l = 2,8 при схеме Р. Фано), что свидетельствует о том, что код Р. Фано хотя и экономный, но не оптимальный, так как, в отличие от схемы Д. Хаффмэна, схема кодирования по Р. Фано не всегда обеспечивает наименьшую возможную среднюю длину кодовых наборов.

Обе эти схемы ориентированы на то, чтобы ценою удлинения кодовых наборов менее вероятных букв достичь уменьшения длин кодовых наборов более вероятных букв. С этой задачей в общем-то справляются обе схемы кодирования, и поэтому с их помощью удается уменьшить среднее число двоичных символов, приходящихся на одну букву. Для того же, чтобы достичь наименьшую возможную средней длины кодового набора, требуется нечто большее, более тонкий механизм кодирования, нежели схема Р. Фано. Что же касается схемы Д. Хаффмэна. то в [1], например, приведено чрезвычайно простое и вместе с тем достаточно строгое доказательство того, что при любом наборе значений вероятностей эта схема обеспечивает наименьшую возможную при побуквенном кодировании среднюю длину кодовых наборов. В рассматриваемом смысле предложенная Д. Хаффмэном схема кодирования является оптимальной.

Следует особо подчеркнуть, что из оптимальности схемы Д. Хаффмэна вовсе не следует единственность варианта достижения этого оптимума. Рассмотрим, например, кодирование букв A, B, C, D и E при следующих значениях вероятностей их встречаемости:

P (A) = 0,49 P (D) = 0,16

P (B) = 0,17 P (E) = 0,01

P (C) = 0,17

С помощью схемы кодирования Д. Хаффмэна приходим к результату:

Буква	Двоичный	Буква	Двоичный
A	код	D	код
B	1	E	0010
C	01		0001
	000		

т.е. к значению средней длины кодовых наборов, равному l_{min} = 2,02 символа на букву.

Легко убедиться, что к такому же значению средней длины мы пришли бы при следующем варианте кодирования:

Буква	Двоичный	Буква	Двоичный
A	код	D	код
B	1	E	010
C	000		011
	001		

Несмотря на свою неоптимальность, схема Р. Фано тем не менее обеспечивает значения l. достаточно близкие или в точности совпадающие с l_{min} . В 1частности, легко убедиться в том, что в случае, когда все $p(i)$ (i = A, B,...) можно представить как $p(i)$=$2^{-\mathrm{m}}$, где m_{i} - натуральные числа, в результате кодирования по этим схемам непременно приходим к одинаковым значениям l. Более того. и этих случаях, независимо от того. используется ли схема Р. Фано или Д. Хаффмэна.число двоичных символов в коде каждой i-й буквы алфавита оказы вается равным $l_i = -\log_2 p(i) = m_i$, т.е. среднее число двоичных сим волов. приходящихся на одну букву, оказывается равным $l = l_{\min} = \sum_{i=1}^{n} p(i) l_i = -\sum_{i=1}^{n} p(i) \log_2 p(i)$.

Забегая вперед, отметим, что выражение, фигурирующее в правой части этою равенства, называется энтропией и имеет ключевое значение в теории информации. Это выражение остается в силе при произвольных значениях $p(i)$, вовсе не обязательно удовлетворяющих условию $p(i)$= $2^{-\mathrm{m}i}$. Более подробно понятие энтропии мы рассмотрим чуть позже, в следующем параграфе этой главы.

Говоря об оптимальности кода Р. Хаффмэна. следует запомнить, что здесь пока речь идет лишь о побуквенном кодировании, и поэтому в общем случае схема кодирования по Д. Хаффмэну в том виде, в каком мы ее рассматривали, также не является пределом компактного представления исходных текстов последовательностью двоичных символов. Если вслед за отказом от постоянства длин кодовых наборов, отказаться также от побуквенного кодирования, т.е. допустить кодирование сразу нескольких букв (комбинации букв), то в общем случае можно добиться значительно большего эффекта сжатия (компрессии) исходных текстов. Заметим при этом, что при заданных статистических характеристиках исходного текста чем длиннее комбинации букв, подвергающиеся кодированию, тем, при прочих равных условиях, большим получается эффект сжатия. Естественно, что с увеличением длины этих последовательностей, растет также время, расходуемое на их кодирование, и поэтому чрезмерно сильное сжатие исходных текстов не всегда оправдано, так как оно может быть связано с большими затратами машинного времени.

Блоковое кодирование. Выше были рассмотрены способы побуквенного кодирования, когда каждой букве на выходе источника приписывалось свое кодовое слово. Такой способ кодирования позволяет достигнуть минимальной средней длины кодового слова.

При кодировании блоками по k букв прежде всего строят вспомогательный алфавит, состоящий из $N=s^k$ букв, так что каждой букве этого алфавита соответствует своя комбинация (блок) из k букв алфавита источника. Вероятность буквы $z_{ij\ldots q}$ вспомогательного алфавита, соответствующей комбинации $x_i x_j \ldots x_q$, в простейшем случае вычисляется по формуле

$$p(z_{ij\ldots q}) = p(x_i)p(x_j)\ldots p(x_q)\,. \qquad (4.2.4)$$

Далее буквы вспомогательного алфавита располагаются в порядке убывания вероятностей, и осуществляется кодирование методом Шеннона–Фэно или Хафмана.

Таким образом, кодирующее устройство при блоковом кодировании разбивает последовательность букв на выходе источника на блоки длиной в k букв каждый и генерирует последовательность кодовых слов, соответствующих каждому из блоков.

Пример 4.2.2. Сообщение источника X составляется из статистически независимых букв, извлекаемых из алфавита A, B, C с вероятностями 0,7; 0,2; 0,1. Произвести двоичное кодирование по методу Шеннона–Фано отдельных букв и двухбуквенных блоков. Сравнить коды по их эффективности.

Решение. Производим побуквенное кодирование методом Шеннона–Фано.

1) Располагаем буквы алфавита источника в порядке убывания вероятностей.

2) Делим алфавит источника на две (m=2) примерно равновероятные группы. Всем сообщениям верхней группы (буква A) приписываем в качестве первого кодового символа 1, всем сообщениям нижней группы приписываем символ 0.

3) Производим второе разбиение на две группы (буквы B и C) и снова букве в верхней группе (B) приписываем символ 1, а в нижней (C) в качестве второго символа кодового слова приписываем 0. Так как в каждой группе оказалось по одной букве, кодирование заканчиваем. Результат приведен в табл. 4.2.2.

Таблица 4.2.2

x_j	$p(x_j)$	Разбиения	Кодовое слово
A	0,7	———	1
B	0,2	—	01
C	0,1		00

Оценим эффективность полученного кода. Энтропия источника

$$I = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i) = 1{,}1568$$

Средняя длина кодового слова

$$L = \sum_{i=1}^{n} l_i p(x_i) = 0{,}7\cdot 1 + 0{,}2\cdot 2 + 0{,}1\cdot 2 = 1{,}3$$

Видим, что $L>H(X)$, и коэффициент эффективности $\gamma_1 = 1{,}1568/1{,}3 = 0{,}8898,$ а избыточность R_1=0,1102.

Покажем, что кодирование блоками по 2 буквы (k=2) увеличивает эффективность кода. Строим вспомогательный алфавит из $N=3^2$ блоков. Вероятности блоков находим по формуле (4.2.4), считая буквы исходного алфавита независимыми. Располагаем блоки в порядке убывания вероятностей и осуществляем кодирование методом Шеннона–Фано. Все полученные двухбуквенные блоки, вероятности их и соответствующие кодовые обозначения сведены в табл. 4.2.3.

При блоковом кодировании средняя длина кодового слова на одну букву = 1,165

При этом коэффициент эффективности

$$\gamma_2 = \acute{I}\ (\tilde{O})/L_2 = 1,1568/1,165 = 0,9955.$$

Избыточность при двухбуквенном кодировании R_2=0,0045.

Получили $\gamma_2 > \gamma_1$, $R_2 << R_1$, что и требовалось показать.

Таблица 4.2.3

Двухбук-венные блоки	Вероят-ности	Разбиения	Кодовые слова
AA	0,49		1
AB	0,14		0 1 1
BA	0,14		0 1 0
AC	0,07		0 0 1 1
CA	00,7		0 0 1 0
BB	0,04		0 0 0 1
BC	0,02		0 0 0 0 1
CB	0,02		0 0 0 0 0 1
CC	0,01		0 0 0 0 0 0

Лабораторная работа № 2
Способы построения двоичных кодов

Цель - овладение навыками алфавитного неравномерного и равномерного двоичного кодирования.

Постановка задачи

Первичный алфавит *X* с символами $x_1, \ldots, x_8$ передается по дискретному двоичному каналу.

1. Требуется построить схему неравномерного кодирования, в которой суммарная длительность при передаче (или суммарное число кодов при хранении) данного сообщения была бы наименьшей.
2. Требуется построить схему равномерного кодирования.

Порядок выполнения задания

1. Закодируйте алфавит X с использование неравномерного алфавитного двоичного кода с разделителем знаков (предложите свой вариант).
2. Для цифр разработайте вариант байтового кодирования. Реализуйте процедуру кодирования программно (ввод – последовательность цифр, вывод последовательность двоичных кодов в соответствии с разработанной кодовой таблицей).
3. Разработайте программу декодирования к п.1 и п.2.

4. Выводы по работе.

Варианты исходных данных к п.1

Вариант	$P(x_1)$	$P(x_2)$	$P(x_3)$	$P(x_4)$	$P(x_5)$	$P(x_6)$	$P(x_7)$	$P(x_8)$
1	0,25	0,18	0,15	0,12	0,1	0,08	0,07	0,05
2	0,03	0,09	0,07	0,11	0,13	0,14	0,23	0,2
3	0,04	0,06	0,09	0,1	0,31	0,18	0,05	0,17
4	0,3	0,13	0,1	0,05	0,15	0,07	0,14	0,06
5	0,27	0,15	0,13	0,09	0,2	0,01	0,08	0,07
6	0,14	0,21	0,09	0,15	0,06	0,13	0,12	0,1
7	0,18	0,15	0,25	0,08	0,1	0,12	0,05	0,07
8	0,23	0,2	0,09	0,07	0,03	0,11	0,13	0,14
9	0,1	0,18	0,05	0,17	0,04	0,09	0,06	0,31
10	0,1	0,15	0,3	0,07	0,14	0,06	0,13	0,05
11	0,09	0,1	0,06	0,12	0,13	0,14	0,21	0,15
12	0,12	0,05	0,07	0,18	0,15	0,1	0,25	0,08
13	0,11	0,09	0,2	0,03	0,13	0,14	0,23	0,07
14	0,04	0,31	0,18	0,1	0,09	0,17	0,06	0,05
15	0,19	0,01	0,07	0,23	0,02	0,16	0,2	0,12
16	0,16	0,08	0,1	0,19	0,01	0,25	0,09	0,12
17	0,29	0,03	0,05	0,16	0,1	0,22	0,07	0,08
18	0,3	0,13	0,01	0,2	0,06	0,05	0,18	0,07
19	0,33	0,09	0,15	0,03	0,07	0,21	0,1	0,02
20	0,17	0,14	0,09	0,12	0,01	0,1	0,06	0,31

Варианты исходных данных к п.2

Вариант	Последовательность цифр
1	2857436325738755173873 78818
2	983287746652641445126377733
3	762635523441234123243512235
4	653254234546252516168126276
5	263623265425345343565857549
6	143412878758979809384653554
7	365767978917715678283839390
8	887374762516546154651212465
9	524413136526368736768745889
10	465776375798690860992651654
11	654574658745686770766526010
12	365456387478757574687598696
13	898724987876564234232435263
14	972368767654656867867768965
15	873673264876875986989568990

16	98187866765687688631256521 1
17	87456875685252628798749848 0
18	64876476875849686859687721 5
19	87584768567866521342413433 4
20	64534636152431328721765441 4

Лабораторная работа № 2
Статистическое кодирование информации

Цель работы: овладение навыками статистического кодирования по методам Шеннона-Фано и Хаффмана.

Постановка задачи:
Задано сообщение, состоящее из букв алфавита {a, b, c, d, e, f, g, h, i}. Требуется выбрать наилучший способ кодирования.

Порядок выполнения задания:

1. Разработать программу кодирования двоичного сообщения на каком-либо языке программирования следующими способами:
   - равномерным двоичным кодом;
   - кодом Шеннона-Фано;
   - кодом Шеннона-Фано с укрупнением (в качестве укрупненных символов использовать все возможные пары символов исходного сообщения);
   - кодом Хаффмана;
   - кодом Хаффмана с укрупнением (в качестве укрупненных символов использовать все возможные пары символов исходного сооообщения);
2. Найти избыточность построенных кодов. Выбрать наилучший способ кодирования. Сделать вывод.

Состав отчета по лабораторной работе:

1. Постановка задачи.
2. Листинги программ кодирования двоичного сообщения (согласно номеру варианта) следующими способами:
   - равномерным двоичным кодом;
   - кодом Шеннона-Фано;
   - кодом Шеннона-Фано с укрупнением (в качестве укрупненных символов использовать все возможные пары символов исходного сообщения);
   - кодом Хаффмана;
   - кодом Хаффмана с укрупнением (в качестве укрупненных символов использовать все возможные пары символов исходного сообщения).

3. Выводы по работе.

Варианты

Вариант	Сообщение
1	a a a b b b b b c c d d d d e e e e e e e a a b c c
2	a d a d bb f b a b c c d f d f a d d f b a d d c
3	a b c d e b b b c c d d d d d e e e b a a a b c d
4	i i a a b b b b b b b c c d d d d e e d c c a i i
5	a a b b b b b c c d d c d e c d e e e b b a c d e
6	f a a a b b f b b b c c d f d d d d c b a b c d a
7	e a a a e b b b b e c d d d d d d a c e b a c e b
8	g g a a b a b a c d c d d c g g c d b a e e e b c
9	b b i i i h h h h h h e e e i i i c c c c c c b b
10	b a b a c a c a f f f f f c a b a c b d d d d b b
11	f f f f f h h c h c h c h e e e d d d c c h d d f
12	d d d e e e e a a a c c c f f f f d f d e c e c e
13	c d c d e a b c a b e d c b c c a d d d e e d a a
14	a b c d e a b c d e a b c d e e e e e c d a c d a c
15	e f c d e f a f e c a f a c c c d d d c d a e e e
16	e g g e f a c a f f g g e e a c f g g g e e e f c
17	g h i i i i i a d d d a h d a h i d a a g g g h h
18	d f a c e a a a b f d b f d c e a a f d f d c c c
19	e c d h b e c d h b e c h b d e c d h b e c d h b
20	a g c d f a g c d f a g c d f f f d d d c c c g g

2.3. Помехоустойчивое кодирование

Лабораторная работа № 5
Помехоустойчивое кодирование двоичных сообщений с использованием кодов Хемминга

Цель работы: изучение и практическое освоение принципов помехоустойчивого кодирования дискретных двоичных сообщений с использованием кодов Хемминга.

Постановка задачи:

Двоичное дискретное сообщение, состоящее из 11 ($n_и$=11) информационных символов, закодировано кодами Хемминга и передано по каналу связи. В канале связи действуют помехи, приводящие к искажению одного передаваемого символа. Требуется обнаружить и исправить ошибки.

Порядок выполнения задания:

1. Исправить ошибку в передаваемой информации с использованием метода четности. Реализовать программным путем алгоритм кодирования по методу четности. Для этого необходимо:
- осуществить проверку на четность по каждой строке;
- осуществить проверку на четность по каждому столбцу;
- определить место возникновения ошибки и исправить ошибку.

2. Определить код Хемминга для двоичного дискретного сообщения с числом информационных символов $n_и$=11. Реализовать программным путем алгоритм восстановления кода Хемминга. Для этого необходимо:
- занести информационные биты в соответствующие элементы входной кодовой комбинации;
- определить число контрольных символов, обеспечивающих заданные требования по помехоустойчивости;
- установить на каких позициях кодовой комбинации следует разместить контрольные символы, и какие позиции займут информационные символы;
- определить значение каждого контрольного символа;
- сформировать кодовую комбинацию, включающую контрольные и информационные символы.

3. Провести проверку комбинации N=15 (количество контрольных бит равно 4, количество информационных бит равно 11), закодированной кодом Хемминга, на наличие одиночной ошибки. Реализовать программным путем алгоритм обнаружения одиночной ошибки. Для этого необходимо:
- ввести в одну из позиций, сформированной кодовой комбинации (задание 2), ошибку;
- в соответствии с алгоритмом проверки и исправления передаваемой последовательности бит в представлении Хемминга устранить ошибку.

Состав отчета по лабораторной работе:

4. Постановка задачи.
2. Листинги программ помехоустойчивого кодирования дискретных двоичных сообщений с использованием кодов Хемминга.
3. Выводы по работе.

Варианты исходных данных:

Вариант	п. 1	п. 2
1	11101010	10101111011

	01010000 10111101 11001100 01111011 1111000 *контрольный столбец 8, контрольная строка 6*	
2	00100001 00011000 11110101 10101010 00000110 0111000 *контрольный столбец 8, контрольная строка 6*	11001100111
3	1111110 0000000 1110001 1100111 1010101 010110 *контрольный столбец 8, контрольная строка 6*	10010101001
4	00011011 11010100 00011110 10101010 01010010 0010101 *контрольный столбец 8, контрольная строка 6*	11101101010
5	10001110 01101010 10000001 11101010 10000010 0100110 *контрольный столбец 8, контрольная строка 6*	10001111011
Вариант	п. 1	п. 2
6	1110001 1010100 0100100 1001011 101101 *контрольный столбец 7, контрольная строка 5*	11000110101
7	1000111 0110101 1100101 1110101	11111000111

	110000 *контрольный столбец 7, контрольная строка 5*	
8	1100111 0110101 1100101 1110101 110000 *контрольный столбец 7, контрольная строка 5*	10011111001
9	1000111 0110101 1100001 1110101 110000 *контрольный столбец 7, контрольная строка 5*	11110010101
10	1001101 0110101 1100101 1110101 110000 *контрольный столбец 7, контрольная строка 5*	10011111110
11	10001110 01101010 10000001 11101010 10000010 0100110 *контрольный столбец 8, контрольная строка 6*	11110000111
12	10001110 01101110 10000001 11101010 10000010 0100110 *контрольный столбец 8, контрольная строка 6*	10101010101
Вариант	п. 1	п. 2
13	00011011 11010110 00011110 10101010 01110100 0010111 *контрольный столбец 8, контрольная строка 6*	11001100110
14	00011000 11010110 00011110	10001111101

	10101010 01000001 0010101 *контрольный столбец 8, контрольная строка 6*	
15	11100111 01010101 11011010 11110011 01101100 0111011 *контрольный столбец 8, контрольная строка 6*	10011001101
15	11100111 01110101 11011010 11110011 01101100 0111011 *контрольный столбец 8, контрольная строка 6*	11110011100
17	11111111 10000001 11011010 11001100 01100110 0010111 *контрольный столбец 8, контрольная строка 6*	11111101010
18	11110000 11111110 01110111 10010011 01111000 1011001 *контрольный столбец 8, контрольная строка 6*	11011101101
19	00011111 11000011 11111111 10001101 10101010 0010010 *контрольный столбец 8, контрольная строка 6*	10101011111
20	01010101 11101100 11100111 10011001 10000111 0101000	11110011110

	контрольный столбец 8, контрольная строка 6	