

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РЕСПУБЛИКИ КАЗАХСТАН**

**Казахский национальный исследовательский технический
университет имени К. И. Сатпаева**

**Т. В. Яскевич
Б. Х. Айтчанов
Ш. М. Байматаева**

ОСНОВЫ ИНФОРМАЦИОННЫХ СИСТЕМ

**Рекомендовано Учебно-методическим советом университета
в качестве учебного пособия**

Алматы 2017

УДК 004.4(075)
ББК 32.973.202я73
Я 81

Рецензенты:

Биттеев Ш. Б., д-р техн. наук, проф., генеральный директор
НПО «Куат»;
Джурунтаев З., д-р техн. наук, ассистент-профессор каф.
«Информационная безопасность»;
Имангалиев Ш. И., канд. техн. наук, зав. каф.
«Информационные системы»;

Печатается по плану Министерства образования и науки
Республики Казахстан на 2017 год.

Яскевич Т. В.

Я 81 Основы информационных систем: Учеб. пособие / *Т. В. Яскевич, Б. Х. Айтчанов, Ш. М. Байматаева*. – Алматы: КазНИТУ имени К. И. Сатпаева, 2017. – 171 с. Ил. 21. Табл. 24. Библиогр. – 42 наим.

ISBN 978-601-323-081-8

В учебном пособии изложены основы информационных систем в аспекте их связи с общей теорией систем и теорией информации. Обсуждаются основные способы представления данных в компьютерных системах и технологии их обработки, история и перспективы развития информационных систем. Приведены примеры по каждой теме для закрепления материала.

Учебное пособие предназначено для студентов и магистрантов специальностей 5В070300 «Информационные системы», 5В070400 «Вычислительная техника и программное обеспечение».

ISBN 978-601-323-081-8 © Яскевич Т.В., Айтчанов Б.Х.,
Байматаева Ш. М., 2017

© КазНИТУ имени К. И. Сатпаева, 2017

ВВЕДЕНИЕ

Сегодня понятие «Информационная система» охватило все сферы жизнедеятельности человека. Идет он в поликлинику, там на него заведена электронная карта, в аптеку за лекарством – там его обслуживают с помощью информационной справочной системой об ассортименте и наличии лекарств; наконец, запускает в космос корабли и управляет ими с помощью современных сложных информационных систем.

Система, системный подход стоят в основе решения любой созидательной проблемы, любого научно-исследовательского проекта.

Понятие системы было введено австрийским биологом Л. фон Берталанфи¹ в 1950 г. («Общая теория систем») и повсеместно применяется на сегодняшний день для описания и преобразования окружающего мира.

Информационная система, по наиболее распространенному определению, является системой, включающей оборудование, программное обеспечение, персонал. Осуществляет сбор, хранение, передачу, обработку и представление пользователю информации. Служит для принятия решения и связана в первую очередь с наукой «Теория информации».

Теория информации как самостоятельная научная дисциплина, связанная с восприятием, передачей и переработкой, хранением и использованием информации, была основана американским ученым *К. Шенноном* в конце 40-х годов XX века. Предложенная Шенноном теория основывалась на фундаментальном понятии количественной меры неопределенности – *энтропии* и связанного с ней понятия *количества информации*.

¹Л. фон Берталанфи. Общая теория систем: критический обзор // Исследования по общей теории систем. – М.: Прогресс, 1969.

Другим фактором в становлении теории информации стало осознание того, что носитель информации – *сигнал* имеет случайную природу. На основе понятий энтропии и количества информации в теории информации введены важные характеристики сигналов и *информационных систем*.

Учебное пособие написано согласно типовому плану 2016 г. по дисциплине «Основы информационных систем». Материал пособия подготовлен для успешного усвоения материала студентами-бакалаврами. Содержит углубленный лекционный материал и множество примеров решения задач для успешного освоения материала практических занятий.

Программой (силлабусом) курса предусмотрено изучение следующих основных тем: фундаментальные положения теории информации, представление и хранение данных в компьютере, файловые операции и системы, количественная мера информации, эффективные и помехоустойчивые коды.

В первом разделе приводится основная информация по теории систем, которая используется при изучении и создании информационных систем.

Во втором разделе рассматриваются основные сведения об информационных системах и связь с теорией информации. Обсуждается история и перспективы развития информационных систем, рассматриваются технологии работы с данными.

Третий раздел посвящен представлению данных в информационных системах. Рассматривается конкретная файловая система как пример информационной системы.

В четвертом разделе рассматриваются вопросы определения основных информационных характеристик информационных систем (количества информации, пропускной способности и скорости передачи).

В пятом разделе рассматриваются основные технологии работы с информацией. К их числу относятся прежде всего эффективное и помехоустойчивое кодирование. В пособии приведены способы построения и использования эффективных и

помехоустойчивых (избыточных) кодов. Приводятся множество примеров решения задач.

Материалы учебного пособия могут быть использованы при изучении последующих специальных дисциплин: «Компьютерные сети» и «Проектирование информационных систем», а также студентами специальности 5В070200 «Автоматизация и управление» при изучении дисциплины «Прикладная теория информации».

1. ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕОРИЯ СИСТЕМ

1.1. Основные понятия теории систем

Информационные системы (ИС) прежде всего связаны с теорией систем и их изучение следует начинать с изучения основных понятий теории систем

Потребность в использовании понятия «система» возникала для объектов различной физической природы с древних времен: еще Аристотель обратил внимание на то, что целое (т. е. система) несводимо к сумме частей, его образующих. Сейчас ученые, изучающие общую теорию систем, провозгласили «*принцип эмерджентности*» [15].

Принцип эмерджентности заключается в том, что свойства целого не сводятся к простой сумме свойств составляющих его частей, а при объединении частей в целое образуется новое качество, не присущее отдельным частям.

Представьте себе кучу разрозненных деталей: материнская плата, клавиатура, процессор – это еще не компьютер. Они должны быть объединены. Но, компьютер – железо. Вычислительной системой он может стать только благодаря программному обеспечению. Тогда компьютер приобретает новое качество, которым не обладают отдельно компьютер и программное обеспечение.

Термин «система» и связанные с ним понятия комплексного, системного подхода исследуются и подвергаются осмыслению философами, биологами, психологами, кибернетиками, физиками, математиками, экономистами, инженерами различных специальностей. Потребность в использовании этого термина возникает в тех случаях, когда невозможно что-то продемонстрировать, изобразить, представить одним выражением и нужно подчеркнуть, что это будет большим, сложным, не полностью сразу понятным (с неопределенностью) и целым, единым. Например, «солнечная система», «система управления станком», «система организационного управления предприятием (городом, регионом и т. п.)», «экономическая система», «система кровообращения» и т. д.

В математике термин «система» используется для отображения совокупности математических выражений или правил: «система уравнений», «система счисления», «система мер» и т. п. Казалось бы, в этих случаях можно было бы воспользоваться терминами «множество» или «совокупность». Однако понятие системы подчеркивает упорядоченность, целостность, наличие определенных закономерностей.

Если попытаться дать общее определение для любых систем, то оно будет очень абстрактным и не удобным для практических целей, однако у всех систем, независимо от их физической природы, есть некоторые общие признаки.

Система – множество элементов, находящихся в отношениях и связях друг с другом, которое образует определенную целостность, единство.

Признаки системности:

- **структурированность**, т. е., возможность расчленения системы на составляющие компоненты; С одной стороны, система это целостное образование и представляет целостную совокупность элементов, а, с другой – в системе четко можно выделить ее элементы (целостные объекты).

- **взаимосвязанность** отдельных частей, т. е. наличие более или менее устойчивых связей (отношений) между элементами системы, превосходящих по своей силе (мощности) связи (отношения) этих элементов с элементами, не входящими в данную систему. В системах любой природы между элементами существуют те или иные связи (отношения). При этом с системных позиций определяющими являются не любые связи, а только лишь существенные связи (отношения), которые определяют интегративные свойства системы.

- **интегративность** системы, т. е. наличие единых целей, свойств, качеств, присущих системе в целом, но не присущих ее элементам в отдельности. Интегративные свойства системы обуславливает тот факт, что свойство системы, несмотря на зависимость от свойств элементов, не определяется ими полностью. Из этого следует, что простая совокупность элементов и связей между ними еще не система, и поэтому, расчленяя систему на отдельные части (элементы) и изучая каждую из них в

отдельности, нельзя познать все свойства нормально (хорошо) организованной системы в целом.

Понятие «система» широко использовалось в различных областях знаний, и на определенной стадии развития научного знания теория систем оформилась в самостоятельную науку.

Краткая история развития системных представлений

Зародились системные представления еще в древности. Из более поздних научных достижений можно назвать, во-первых, работы А. Ампера.

А. Ампер написал серию работ «Опыт о философии наук, или аналитическое изложение классификации человеческих знаний» (1834–1843). В них выделил науку об управлении государством и назвал ее «кибернетикой» (от греческого «*κίβηρ*» – кормчий, рулевой, управляющий чем-то).

Его идеи развил польский философ Болеслав Трентовский, который в 1843 г. опубликовал книгу «Отношение философии к кибернетике как искусству управления народом». Важный вклад в становление системных представлений внес в начале XIX в. А. А. Богданов.

А. Богданов издал фундаментальный труд – трехтомник «Всеобщая организационная наука (*тектология*)»² (1911–1925). Все явления являются процессами организации или дезорганизации. Все объекты имеют определенную степень организованности. Тектология призвана изучать общие закономерности организации. Что такое организованность? Чем уровень организованности выше, тем сильнее свойства целого отличаются от суммы свойств его частей. Богданов рассмотрел соотношение понятий «устойчивость» и «изменчивость», показал значение обратных связей, роль открытых систем.

Так, философия явилась источником возникновения обобщающего направления, названного теорией систем. Основоположителем этого направления считается биолог Л. фон Берталанфи³.

² Богданов А. А. Всеобщая организационная наука (тектология): В 2-х кн. – М.: Наука, 1989.

³Л. фон Берталанфи. Общая теория систем: критический обзор // Исследования по общей теории систем. – М.: Прогресс, 1969.

В 60-е годы при постановке и исследовании сложных проблем проектирования и управления довольно широкое распространение получил термин «*системотехника*».

Применительно к задачам управления в определенный период более широкое распространение получил термин «*кибернетика*», введенный М. А. Ампером, принятый для названия новой «науки об управлении в живых организмах и машинах» Н. Винером.

Из более поздних достижений следует выделить работу бельгийского ученого Ивана Пригожина «Порядок из хаоса» (1979). Материя – не пассивная субстанция, ей присуща спонтанная активность, вызванная наличием неравновесных состояний. Благодаря таким представлениям возникла так называемая «*теория катастроф*» (*теория бифуркаций*).

К числу задач, решаемых теорией систем, относятся:

- определение общей структуры системы;
- организация взаимодействия между подсистемами и элементами;
- учет влияния внешней среды;
- выбор оптимальной структуры системы;
- выбор оптимальных алгоритмов функционирования системы.

Рассматривая понятие системы более подробно следует выделить для любой системы множество входов, множество выходов, множество состояний, связанных оператором переходов (переходным отображением) и оператором выходов (функцией наблюдения):

$$S = (T, U, Y, X, \sigma, \eta),$$

где T – время; U – входы; Y – выходы; X – состояния; σ – оператор переходов; η – оператор выходов. Это определение учитывает все основные компоненты, рассматриваемые в кибернетике.

Основные проблемы теории систем:

Идентификация – описание связей между входами, выходами и состояниями (нахождение функции наблюдения η , переходного отображения σ).

Прогнозирование – предсказание выхода Y по входу U .

Управление – определение U для известного значения X .

Диагностирование – определение состояния X по известным входу U и выходу Y .

Распознавание – определение U по известному выходу Y . Исследование объекта как системы предполагает использование ряда систем представлений (категорий) среди которых основными являются:

1. **Структурное** представление, связанное с выделением элементов системы и связей между ними.

2. **Функциональное** представление систем – выделение совокупности функций (целенаправленных действий) системы и её компонентов, направленное на достижение определённой цели.

3. **Макроскопическое** представление – понимание системы как нерасчленимого целого, взаимодействующего с внешней средой.

4. **Микроскопическое** представление, основанное на рассмотрении системы как совокупности взаимосвязанных элементов. Оно предполагает раскрытие структуры системы.

5. **Иерархическое** представление, основанное на понятии подсистемы, получаемом при разложении (декомпозиции) системы, обладающей системными свойствами, которые следует отличать от её элемента – неделимого на более мелкие части (с точки зрения решаемой задачи). Система может быть представлена в виде совокупностей подсистем различных уровней, составляющую системную иерархию, которая замыкается снизу только элементами.

6. **Процессуальное** представление, предполагающее понимание системного объекта как динамического объекта, характеризующегося последовательностью его состояний во времени.

Рассмотрим основные понятия, характеризующие строение и функционирование систем.

1. **Элемент.** Под элементом принято понимать простейшую неделимую часть системы. Систему можно разделить на элементы различными способами в зависимости от формулировки цели и ее уточнения в процессе исследования. Элемент всегда является частью системы и вне ее не представляет смысла.

2. **Подсистема** – часть системы с некоторыми связями и отношениями. Система может быть разделена на элементы не сразу, а последовательным разделением на подсистемы, которые

представляют собой компоненты более крупные, чем элементы, и в то же время более детальные, чем система в целом.

3. Структура – это совокупность элементов и связей между ними. Она отражает наиболее существенные взаимоотношения между элементами и их группами (компонентами, подсистемами), которые мало меняются при изменениях в системе и обеспечивают существование системы и ее основных свойств. Структура может быть представлена графически, в виде теоретико-множественных описаний, матриц, графов и других языков моделирования структур. Структуру часто представляют в виде иерархии.

4. Связь – обеспечивает возникновение и сохранение структуры и целостных свойств системы. Это понятие характеризует одновременно и строение (статику), и функционирование (динамику) системы. Связь характеризуется направлением, силой и характером (или видом). По первым двум признакам связи можно разделить на направленные и ненаправленные, сильные и слабые, а по характеру – на связи подчинения, наследования, равноправные (или безразличные), связи управления.

Связи можно разделить также по месту приложения (внутренние и внешние), по направленности процессов в системе в целом или в отдельных ее подсистемах (прямые и обратные). Прямые связи предназначены для заданной функциональной передачи вещества, энергии, информации или их комбинаций – от одного элемента к другому в направлении основного процесса.

Обратные связи, в основном, выполняют осведомляющие функции, отражая изменение состояния системы в результате управляющего воздействия на нее.

5. Входы и выходы – материальные или информационные потоки входящие и выходящие из системы. Система осуществляет свою связь со средой следующим образом. Вход данной системы является в то же время выходом предшествующей, а выход данной системы – входом последующей. Таким образом, вход и выход располагаются на границе системы и выполняют одновременно функции входа и выхода предшествующих и последующих систем.

6. Состояние мгновенный «срез» системы, остановку в ее развитии. Его определяют либо через входные воздействия и выходные сигналы (результаты), либо через макропараметры, макросвойства системы. Таким образом, состояние – это множество существенных свойств, которыми система обладает в данный момент времени.

7. Поведение. Если система способна переходить из одного состояния в другое (например, $x_1 \rightarrow x_2 \rightarrow x_3$), то говорят, что она обладает поведением. Этим понятием пользуются, когда неизвестны закономерности переходов из одного состояния в другое. Тогда говорят, что система обладает каким-то поведением и выясняют его закономерности. Поведение каждой системы объясняется структурой систем низшего порядка, из которых состоит данная система, и наличием признаков равновесия (*гомеостаза*). В соответствии с признаком равновесия система имеет определенное состояние (состояния), которое является для нее предпочтительным. Поэтому поведение систем описывается в терминах восстановления этих состояний, когда они нарушаются в результате изменения окружающей среды.

8. Внешняя среда. Под внешней средой понимается множество элементов, которые не входят в систему, но изменение их состояния вызывает изменение поведения системы.

9. Модель. Под моделью системы понимается описание системы, отображающее определенную группу ее свойств. Углубление описания – детализация модели. Создание модели системы позволяет предсказывать ее поведение в определенном диапазоне условий.

10. Модель функционирования (поведения) системы – это модель, предсказывающая изменение состояния системы во времени, например: натурные (аналоговые), электрические, машинные на ЭВМ и др.

11. Равновесие – это способность системы в отсутствие внешних возмущающих воздействий (или при постоянных воздействиях) сохранить свое состояние сколь угодно долго.

12. Устойчивость. Под устойчивостью понимается способность системы возвращаться в состояние равновесия после того, как она была из этого состояния выведена под влиянием внешних возмущающих воздействий.

13. Под развитием обычно понимают: увеличение сложности какой-либо системы, улучшение приспособленности к внешним условиям. В результате возникает новое качество или состояние объекта.

14. Цель – образ несуществующего, но желаемого, с точки зрения задачи или рассматриваемой проблемы, состояния среды, т. е. такого состояния, которое позволяет решать проблему при данных ресурсах. Это описание, представление некоторого наиболее предпочтительного (с точки зрения поставленной цели и доступных ресурсов) состояния системы.

Классификация систем. Классификацией называется разбиение на классы по наиболее существенным признакам. Под классом понимается совокупность объектов, обладающие некоторыми признаками общности. Признак (или совокупность признаков) является основанием (критерием) классификации.

Система может быть охарактеризована одним или несколькими признаками и соответственно ей может быть найдено место в различных классификациях, каждая из которых может быть полезной при выборе методологии исследования. Обычно цель классификации ограничить выбор подходов к отображению систем, выработать язык описания, подходящий для соответствующего класса.

По содержанию различают *реальные* (материальные), объективно существующие, и *абстрактные* (концептуальные, идеальные), являющиеся продуктом мышления.

Реальные системы делятся на *естественные* (природные системы) и *искусственные* (антропогенные).

Естественные системы: системы неживой (физические, химические) и живой (биологические) природы.

Искусственные системы: создаются человечеством для своих нужд или образуются в результате целенаправленных усилий.

Искусственные делятся на *технические* (техно-экономические) и *социальные* (общественные).

Техническая система спроектирована и изготовлена человеком в определённых целях.

К социальным системам относятся различные системы человеческого общества.

Выделение систем, состоящих из одних только технических устройств почти всегда условно, поскольку они не способны вырабатывать своё состояние. Эти системы выступают как части более крупных, включающих людей – *организационно-технических систем*.

Организационная система, для эффективного функционирования которой существенным фактором является способ организации взаимодействия людей, называются *активными* [33]. Система, где рассматривается взаимодействие человека с технической подсистемой, называется *человеко-машинной* системой.

Таким образом, под *техническими системами* понимают единую конструктивную совокупность взаимосвязанных и взаимодействующих объектов, предназначенных для целенаправленных действий с задачей достижения в процессе функционирования заданного результата.

Отличительными признаками технических систем по сравнению с произвольной совокупностью объектов или по сравнению с отдельными элементами является конструктивность (практическая осуществляемость отношений между элементами), ориентированность и взаимосвязанность составных элементов и целенаправленность.

Для того чтобы система была устойчивой к воздействию внешних влияний, она должна иметь устойчивую структуру. Выбор структуры практически определяет технический облик как всей системы, так её подсистем, и элементов. Вопрос о целесообразности применения той или иной структуры должен решаться исходя из конкретного назначения системы. От структуры зависит также способность системы к перераспределению функций в случае полного или частичного отхода отдельных элементов, а, следовательно, надёжность и живучесть системы при заданных характеристиках её элементов.

Абстрактные системы являются результатом отражения действительности (реальных систем) в мозге человека.

Абстрактные системы разделяют на системы непосредственного отображения (отражающие определённые аспекты реальных систем) и системы генерализирующего (обобщающего) отображения. К первым относятся математические и эври-

стические модели, а ко вторым – концептуальные системы (теории методологического построения) и языки.

Приведем основные способы классификации (возможны и другие критерии классификации систем).

1. По отношению системы к окружающей среде:

- открытые (есть обмен ресурсами с окружающей средой);
- закрытые (нет обмена ресурсами с окружающей средой).

Для закрытой системы, любой её элемент имеет связи только с элементами самой системы.

Разумеется, закрытые системы представляют собой некоторую абстракцию реальной ситуации, так как, строго говоря, изолированных систем не существует. Однако, очевидно, что упрощение описания системы, заключающиеся в отказе от внешних связей, может привести к полезным результатам, упрощающим исследование системы.

2. По описанию переменных системы:

- с качественными переменными (имеющими лишь содержательное описание);
- с количественными переменными (имеющими дискретно или непрерывно описываемые количественным образом переменные);
- смешанного (количественно-качественного) описания.

3. По типу описания закона (законов) функционирования системы:

- типа «Черный ящик» (неизвестен полностью закон функционирования системы; известны только входные и выходные сообщения);

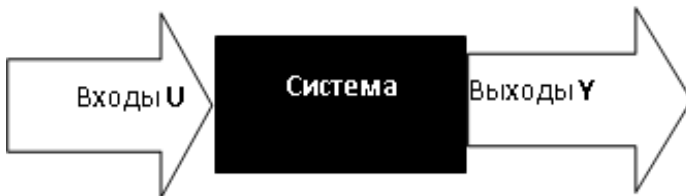


Рис. 1. Модель черного ящика

- не параметризованные (закон не описан; описываем с помощью хотя бы неизвестных параметров; известны лишь некоторые априорные свойства закона);
- параметризованные (закон известен с точностью до параметров и его возможно отнести к некоторому классу зависимостей);
- типа "Белый (прозрачный) ящик" (полностью известен закон функционирования системы). В «*белом ящике*» указываются все элементы системы, все связи между элементами внутри системы и связи определенных элементов с окружающей средой (входы и выходы системы).

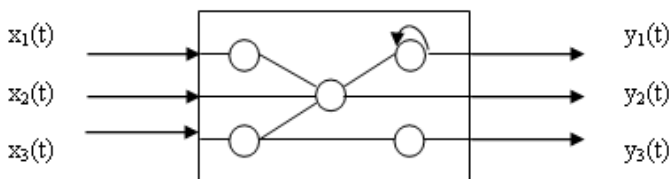


Рис. 2. Модель белого ящика

4. По способу управления системой (в системе):

- управляемые извне системы (без обратной связи, регулируемые, управляемые структурно, информационно или функционально);
- управляемые изнутри (самоуправляемые или саморегулируемые – программно управляемые, регулируемые автоматически, адаптируемые – приспособляемые с помощью управляемых изменений состояний, и самоорганизующиеся – изменяющие во времени и в пространстве свою структуру наиболее оптимально, упорядочивающие свою структуру под воздействием внутренних и внешних факторов);
- с комбинированным управлением (автоматические, полуавтоматические, автоматизированные, организационные).

Пример:

1. Опишите в виде «черного ящика» транслятор.

Решение: Система «*транслятор*» есть программа. Вход транслятора – программа на языке верхнего уровня u_1 , выход – объектный модуль y_1 .

2. Опишите в виде черного ящика систему компоновщик.

Решение: Система «компоновщик» есть программа. Вход компоновщика – множество объектных модулей $u_1, u_2, \dots, u_n$. Выход – исполняемый файл y_1 .

Свойства систем, связанных с целями и функциями:

1) синергичность – однонаправленность (или целенаправленность) действий компонентов усиливает эффективность функционирования системы;

2) приоритет интересов системы более широкого (глобального) уровня перед интересами её компонентов;

3) эмерджентность (лат.: «выбывающий», англ.: «возникновение нового») – цели (функции) компонентов системы не всегда совпадают с целями (функциями) системы;

4) мультипликативность – как позитивные, так и негативные эффекты функционирования компонентов в системе обладают свойством умножения, а не сложения;

5) целенаправленность;

6) альтернативность путей функционирования и развития;

Связанные со структурой:

1) целостность – первичность целого по отношению к частям;

Для того чтобы глубже понять свойство целостности, необходимо, прежде всего, учитывать две его стороны [2]:

2) свойства системы (целого) Q_s не является простой суммой свойств составляющих ее элементов (частей) q_i :

$$Q_s \neq \sum_{i=1}^n q_i$$

3) свойства системы (целого) зависят от свойств составляющих ее элементов (частей): $Q_s = f(q_i)$.

Кроме этих двух основных сторон, следует иметь в виду, что объединенные в систему элементы, как правило, утрачивают часть своих свойств, присущих им вне системы, т. е. система как бы подавляет ряд свойств элементов. Но, с другой стороны, элементы, попав в систему, могут приобрести новые свойства.

Обратимся к закономерности, двойственной по отношению к закономерности целостности. Ее называют физической *аддитивностью, независимостью, суммативностью, обособленностью*. Свойство физической аддитивности проявляется у системы, как бы распавшейся на независимые элементы. В этом случае становится справедливым

$$Q_s = \sum_{i=1}^n q_i .$$

В этом крайнем случае и говорить о системе уже нельзя.

Свойство целостности связано с целью, для выполнения которой предназначена система.

1) неаддитивность – принципиальная несводимость свойств системы к сумме свойств составляющих её компонентов;

2) структурность – возможна декомпозиция системы на компоненты, устанавливание связей между ними;

3) иерархичность – каждый компонент системы может рассматриваться как система (подсистема) более широкой глобальной системы. Иерархичность как закономерность заключается в том, что закономерность целостности проявляется на каждом уровне иерархии. Благодаря этому на каждом уровне возникают новые свойства, которые не могут быть выведены как сумма свойств элементов. При этом важно, что не только объединение элементов в каждом узле приводит к появлению новых свойств, которых у них не было, и утрате некоторых свойств элементов, но и что каждый член иерархии приобретает новые свойства, отсутствующие у него в изолированном состоянии.

В зависимости от структуры и пространственно-временных свойств системы делятся на **простые, сложные и большие**.

Сложные системы объединяет более обширную группу систем, то есть большие – подкласс сложных систем.

Часто различают понятия «*большая*» и «*сложная*» система. Большая система или малая определяется количеством входящих в нее элементов, например:

малые системы ($10 \dots 10^3$ элементов),
большие ($10^4 \dots 10^7$ элементов),
очень большие ($10^7 \dots 10^{30}$ элементов),
супербольшие ($10^{30} \dots 10^{200}$ элементов).

Является ли система сложной или простой определяется связями между элементами.

	Сложные	Простые
Большие	Мозг, экономика, живой организм	Телефонный справочник, словарь, шифрозамок для вора
Малые	Неисправный бытовой прибор для пользователя	Исправный бытовой прибор, шифрозамок для хозяина

Четкой границы, отделяющей простые системы от сложных, нет. Деление это условное и возникло из-за появления систем, имеющих в своем составе совокупность подсистем с наличием функциональной избыточности. Простая система может находиться только в двух состояниях: состоянии работоспособности (исправном) и состоянии отказа (неисправном). При отказе элемента простая система либо полностью прекращает выполнение своей функции, либо продолжает ее выполнение в полном объеме, если отказавший элемент резервирован. Сложная система при отказе отдельных элементов и даже целых подсистем не всегда теряет работоспособность, зачастую только снижаются характеристики ее эффективности. Это свойство сложных систем обусловлено их функциональной избыточностью и, в свою очередь, затрудняет формулировку понятия «отказ» системы.

Основополагающими при анализе и синтезе больших и сложных систем являются процедуры *декомпозиции* и *агрегирования*.

Декомпозиция – разделение систем на части, с последующим самостоятельным рассмотрением отдельных частей.

Очевидно, что декомпозиция представляет собой понятие, связанное с моделью, так как сама система не может быть расчленена без нарушений свойств. На уровне моделирования разрозненные связи заменяются соответственно эквивалентами, либо модели систем строятся так, что разложение её на отдельные части при этом оказывается естественным.

Применительно к большим и сложным системам декомпозиция является мощным инструментом исследования.

Агрегирование является понятием, противоположным декомпозиции. В процессе исследования возникает необходимость

объединения элементов системы с целью рассмотреть её с более общих позиций.

Декомпозиция и агрегирование представляют собой две противоположные стороны подхода к рассмотрению больших и сложных систем, применяемые в диалектическом единстве.

Чаще всего системы делят соответственно следующим характеристикам:

1. Детерминированные и стохастические системы

Если внешние воздействия, приложенные к системе (управляющие и возмущающие) являются определенными известными функциями времени $u = f(t)$. В этом случае состоянии системы описываемой обыкновенными дифференциальными уравнениями, в любой момент времени t может быть однозначно описано по состоянию системы в предшествующий момент времени. Системы, для которых состояние системы однозначно определяется начальными значениями и может быть предсказано для любого момента времени называются детерминированными.

Стохастические системы – системы, изменения в которых носят случайный характер. Например, воздействие на энергосистему различных пользователей.

Статистические свойства случайной величины определяют по ее функции распределения или плотности вероятности.

2. Хорошо организованные, плохо организованные (диффузные), самоорганизующиеся системы.

Хорошо организованные системы. Описание объекта в виде хорошо организованной системы применяется в тех случаях, когда можно предложить детерминированное описание и экспериментально доказать правомерность его применения, адекватность модели реальному процессу.

Плохо организованные системы. При представлении объекта в виде *«плохо организованной или диффузной системы»* не ставится задача определить все учитываемые компоненты, их свойства и связи между ними и целями системы. Подход к отображению объектов в виде диффузных систем широко применяется при: описании систем массового обслуживания, определении численности штатов на предприятиях и учреждениях,

исследовании документальных потоков информации в системах управления и т. д.

Самоорганизующиеся системы. Отображение объекта в виде самоорганизующейся системы – это подход, позволяющий исследовать наименее изученные объекты и процессы. Самоорганизующиеся системы обладают признаками диффузных систем: стохастичностью поведения, нестационарностью отдельных параметров и процессов. К этому добавляются такие признаки, как непредсказуемость поведения; способность адаптироваться к изменяющимся условиям среды, изменять структуру при взаимодействии системы со средой, сохраняя при этом свойства целостности; способность формировать возможные варианты поведения и выбирать из них наилучший и др.

3. Динамические и статические

Динамические системы характеризуются тем, что их выходные сигналы в данный момент времени определяются характером входных воздействий в прошлом и настоящем (зависит от предистории). В противном случае системы называют статическими.

Различают системы линейные и нелинейные. Для линейных систем реакция на сумму двух или более различных воздействий эквивалентна сумме реакций на каждое возмущение в отдельности, для нелинейных – это не выполняется.

4. Стационарные и нестационарные

Если параметры систем изменяются во времени, то она называется нестационарной, противоположным понятием является понятие стационарной системы.

5. Дискретные и непрерывные

Если вход и выход системы измеряется или изменяется во времени дискретно, через шаг Δt , то система называется *дискретной*. Противоположным понятием является понятие непрерывной системы.

6. Открытые и закрытые. В закрытой системе какой-либо обмен энергией, веществом и информацией с окружающей средой отсутствует. Любой элемент закрытой системы имеет связи только с элементами самой системы. Открытые – это системы, обменивающиеся энергией, веществом и информацией с окружающей средой. В открытых системах могут происходить явле-

ния самоорганизации, усложнения или спонтанного возникновения порядка. Все реальные системы являются открытыми;

Вопросы и задания

1. Опишите в виде черного ящика систему шифрования.
2. Упорядочьте в порядке возрастания организованности системы: Файловая система. СУБД.
3. Упорядочьте в порядке возрастания организованности системы: Набор предопределенных выражений языка программирования (ВЯ), Набор операторов языка программирования (ОЯ), Программа для ЭВМ (П), Библиотека процедур и функций (ПФ), Набор допустимых символов языка программирования (СЯ).
4. Что такое «эмерджентность»?
5. Что такое «элемент»?
6. Чем отличается модель «белого ящика» от модели «черного ящика»?
7. Назовите противоположные по смыслу классы систем:
линейная – нелинейная
детерминированная – ...
большая – ...
динамическая – ...
стационарная – ...
закрытая – ...
8. Дайте определения понятиям «входы», «выходы» и «состояния» системы.

2. ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕОРИЯ ИНФОРМАЦИИ

Из всего многообразия современных технических систем информационные системы составляют особую группу. Их назначение – передача, преобразование и хранение информации (связные, локационные, навигационные, телевизионные системы, электронно-вычислительные и системы информационно-измерительной техники, автоматизированные системы управления и контроля и т. п.).

Информационные системы существенно отличаются от энергетических устройств и систем (двигателей, электрических генераторов, линий электропередач), основой функционирования которых являются процессы передачи и преобразования энергии.

Главным критерием качества работы информационной системы служит их способность передавать, накапливать или преобразовать максимальное количество информации в единицу времени при допустимых искажениях и затратах, а не к. п. д. энергетических установок [2,7].

В информационных системах энергетические состояния играют второстепенную роль, а сама энергия является характеристикой сигнала, который используется в качестве транспортного средства.

Информационная система является, как правило, составной частью системы управления и поэтому ее уточненное определение – управляющая информационная система – система сбора, хранения, обновления, передачи, обработки и выдачи информации, необходимой для управления объектом.

Задачами управляющей информационной системы являются [1, 8, 10]:

- 1) обеспечение информацией о происходящем внутри системы, а также за ее пределами;
- 2) обеспечение информацией о правилах решения;
- 3) регистрация, анализ и оценка данных о среде объекта;
- 4) наблюдение за ходом управляемых процессов, получение, анализ и оценка данных о реальном состоянии управляемой системы;

- 5) обеспечение ввода данных и обработка данных;
- 6) извлечение из обработанных данных в нужной форме и необходимом объеме информации для принятия решений;
- 7) получение административной и мотивационной информации для принятия решения;
- 8) контроль за выполнением принятых решений.

Любая информационная система имеет следующие типы основных подсистем:

- подсистема информационного обеспечения (данных);
- подсистема интеллектуального обеспечения (информации, знаний);
- подсистема технического обеспечения (аппаратуры);
- подсистема технологического обеспечения (технологии);
- подсистема коммуникативного обеспечения (интерфейса);
- подсистема анализа и проектирования;
- подсистема оценки адекватности и качества, верификации;

Информационная среда – это среда (т. е. система и ее окружение) из взаимодействующих информационных систем, включая и информацию, актуализируемую в этих системах.

Информационная система управления – система, предназначенная для управления, как другой системой, так и внутри системы (т. е. в качестве управляющей подсистемы).

Различают 6 основных типов информационных систем управления (тип системы определяется целью, ресурсами, характером использования и предметной областью):

1. Диалоговая система обработки запросов (Transaction Processing System) – для реализации текущих, краткосрочных, тактического характера, часто рутинных и жестко структурируемых и формализуемых процедур, например, обработки накладных, ведомостей, бухгалтерских счетов, складских документов и т. д.

2. Система информационного обеспечения (Information Provision System) – для подготовки информационных сообщений краткосрочного (обычно) использования тактического или стратегического характера, например, с использованием данных из базы данных и структурированных, формализованных процедур.

3. Система поддержки принятия решений (Decision Support System) – для анализа (моделирования) реальной формализуемой ситуации, в которой менеджер должен принять некоторое решение, возможно, просчитав различные варианты потенциального поведения системы (варьируя ее параметры); такие системы используются как в краткосрочном, так и в долгосрочном управлении тактического или стратегического характера в автоматизированном режиме.

4. Интегрированная, программируемая система принятия решения (Programmed Decision System) предназначена для автоматического управления, в соответствии с программно реализованными в системе, структурированными и формализованными критериями оценки, отбора (выбора) решений; используются как в краткосрочном, так и в долгосрочном управлении тактического (стратегического) характера.

5. Экспертные системы (Expert System) – информационные консультирующие и (или) принимающие решения системы, которые основаны на структурированных, часто плохо формализуемых процедурах, использующих опыт и интуицию, т. е. поддерживающие и моделирующие работу экспертов, интеллектуальные особенности; системы используются как в долгосрочном, так и в краткосрочном оперативном прогнозировании, управлении.

6. Интеллектуальные системы, или системы, основанные на знаниях (Knowledge Based System) – поддерживают задачи принятия решения в сложных системах, где необходимо использование знаний в достаточно широком диапазоне, особенно в плохо формализуемых и плохо структурируемых системах, нечетких системах и при нечетких критериях принятия решения; эти системы наиболее эффективны и применимы для сведения проблем долгосрочного, стратегического управления к проблемам тактического и краткосрочного характера, повышения управляемости, особенно в условиях многокритериальности. В отличие от экспертных систем, в системах, основанных на знаниях, следует по возможности избегать экспертных и эври-

стических процедур и прибегать к процедурам минимизации риска. Здесь более существенно влияние профессионализма персонала, ибо при разработке таких систем необходимо сотрудничество и взаимопонимание не только разработчиков, но и пользователей, менеджеров, а сам процесс разработки, как правило, происходит итерационно, итерационными улучшениями, постепенным переходом от процедурных знаний (как делать) – к непроцедурным (что делать).

2.1. Основные понятия и определения информационной системы

Производственные процессы, а также процессы живой природы связаны с получением, передачей, преобразованием, накоплением и хранением информации. Существуют различные определения информации. Обычно под информацией понимают новые сведения об окружающем мире, которые получены в результате взаимодействия с ним [8].

Информация – это сведения, являющиеся объектом хранения, передачи и преобразования. Информация – это, прежде всего сведения, которые должны быть использованы.

Рассмотрим основные понятия и определения информационной системы на примере системы передачи информации (рис. 3).

Система передачи информации в качестве информационной системы включает следующие блоки: отправитель информации, линия связи, получатель [10].

Информация поступает в систему в форме сообщений. Сообщение – информация, выраженная в определенной форме и подлежащее дальнейшему преобразованию, т. е. сообщение – форма представления информации (текст телеграмм, речь оратора, показания прибора).

Сообщение для передачи его в соответствующий адрес должно быть предварительно преобразовано в сигнал.

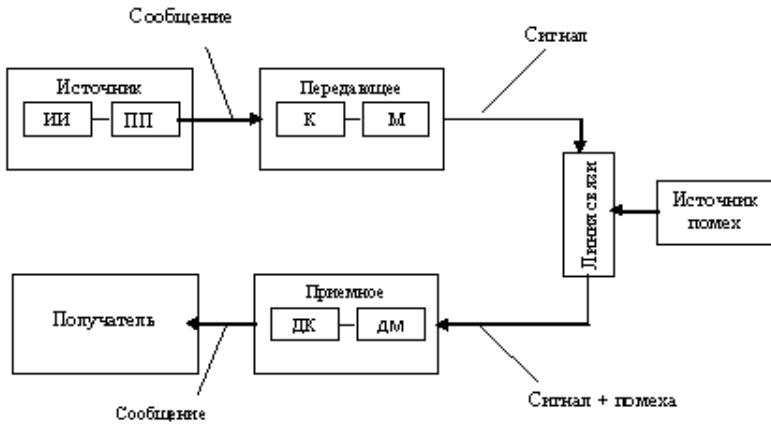


Рис. 3. Система передачи информации

Сообщение – совокупность знаков или первичных сигналов, содержащих информацию. Источник сообщений – совокупность источника информации (ИИ) и первичного преобразователя (ПП) датчика, человека оператора и т. д.

В процессе преобразования сообщений в сигнал происходит кодирование сообщений. В широком смысле кодирование – преобразование сообщений в сигнал, удобный для передачи по данному каналу, а в узком – это отображение дискретных сообщений сигналами в виде определенных сочетаний символов. Осуществляются преобразования сообщений в сигналы удобные для передачи по конкретным линиям связи (либо для хранения в запоминающих системах).

При передаче осуществляется воздействие на один или несколько параметров выбранного носителя, изменяя его в соответствии с передаваемой информацией. Такой процесс называют *модуляцией*, моделируемые параметры называют *информативными*.

При передаче на сигналы воздействуют помехи. Под помехами понимают любые мешающие внешние воздействия возмущения (атмосферные и промышленные помехи; влияние посторонних источников), а также искажение сигналов в самой аппаратуре (аппаратные помехи), вызывающие случайные от-

ключения принятого сигнала от передаваемого сигнала. Эффект воздействия помех стараются учесть эквивалентным изменением характеристик линий связи.

На приемнике осуществляется обратная операция – демодуляция и декодирование, т. е. восстановление по принятому сигналу переданного сообщения.

2.2. Виды информации. Этапы обращения информации

Сообщения различают по характеру измерения во времени – на непрерывные и дискретные [7, 27]. Непрерывные по времени сообщения отображаются непрерывной функцией времени, дискретные – дискретной функцией времени (поступают в определенный момент времени). Так как сообщения носят случайный характер, то непрерывные сообщения описываются случайной функцией времени, а дискретные сообщения – как цепь случайных событий.

Сообщения различают по множеству – непрерывные и дискретные. Непрерывные по множеству сообщения характеризуются тем, что функция их описывающая может принимать множество значений в некотором интервале. Дискретные – могут быть описаны с помощью конечного набора чисел или дискретных значений некоторых функций.

Дискретность по времени и множеству не связана друг с другом. Поэтому возможны следующие виды сообщений (рис. 4):

- а) непрерывные по множеству и по времени;
- б) непрерывные по множеству, дискретные во времени;
- в) дискретные по множеству и непрерывные по времени;
- г) дискретные по множеству и по времени.

Информацию различают на следующие виды:

- 1) по областям знаний (биологическая, техническая, экономическая и т. д.);
- 2) по физической природе восприятия (зрительная, слуховая, вкусовая);
- 3) по структурно–метрическим свойствам.

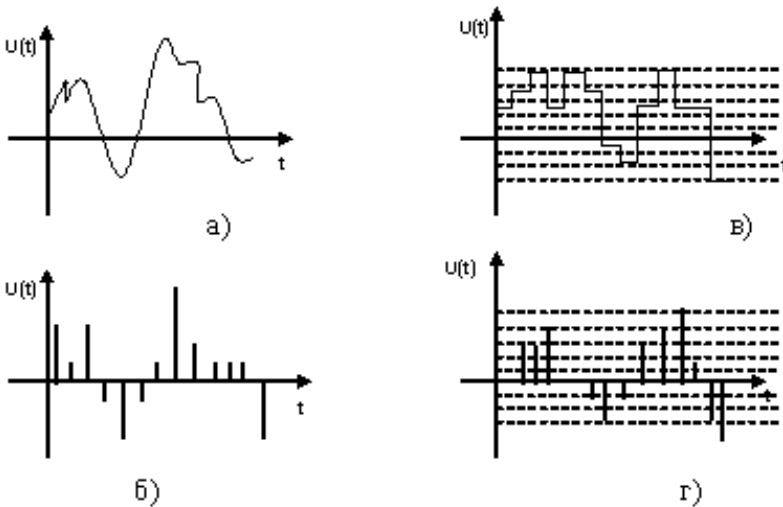


Рис. 4. Виды сообщений

Таблица 1

Структура информации

Виды информации	Обозначения	Форма представления информации		
		Топологическая	Абстрактная	Лингвистическая
Событие	Φ^0	Точка	суждение	Знак
величина	Φ^1	линия	понятие	буква
функция	Φ^2	поверхность	образ	слово
комплекс	Φ^3	объем	система	предложение
.....				
Поле	Φ^n	Пространство	Универсум	Фонд

Особый интерес представляет по этому виду параметрическая информация. Параметрическая информация – наборы численных оценок значений каких-либо параметров, результаты количественной оценки при исследовании.

Структуру информации можно изменять, переходя из одного вида информации к другому.

По размерности информации структура информации: информация нулевого порядка, соответствующая мощности точки, информация первого порядка – мощности линии и т. д.

Этапы обращения информации. В информационных системах существуют следующие этапы обращения информации (рис. 5):

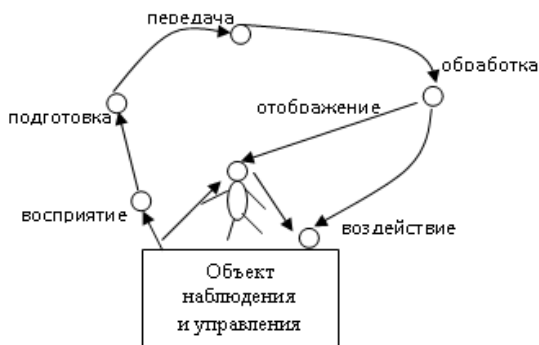


Рис. 5. Этапы обращения информации

1. Этап восприятия информации.
2. Этап подготовки информации.
3. Этап передачи и хранения информации.
4. Этап обработки информации.
5. Этап отображения информации.
6. Этап воздействия.

1. На этапе восприятия осуществляется целенаправленное извлечение и анализ информации о каком-либо объекте, в результате чего формируется образ объекта, проводятся его опознание и оценка.

2. На этапе подготовки информации проводятся такие операции, как нормализация, аналого-цифровое преобразование, шифрование. Иногда этот этап рассматривается как вспомогательный на этапе восприятия. В результате восприятия и подготовки получается сигнал в форме, удобной для передачи или обработки.

3. На этапах передачи и хранения информация пересылается либо с одного места в другое, либо от одного момента времени до другого. Поскольку теоретические вопросы, возникающие на этих этапах, близки друг к другу, этап хранения часто в самостоятельный этап не выделяется. При этом передача информации получает более широкое толкование. Для передачи на расстояние используются каналы различной физической природы, самые распространенные – электрические и электромагнитные, наиболее перспективен – оптический. Для хранения информации используются в основном полупроводниковые и магнитные носители.

4. На этапах обработки информации выявляются ее общие и существенные взаимозависимости, представляющие интерес для системы. Преобразование информации на этапе обработки осуществляется средствами информационной техники, либо человеком. Если процесс обработки формализуем, он выполняется техническими средствами. (ЭВМ, микропроцессоры), если не формализуем и требует творческого подхода, обработка информации осуществляется человеком. В системах управления целью обработки является решение задачи выбора управляющих воздействий (этап принятия решений).

5. Этап отображения информации должен предшествовать этапам, связанным с участием человека. Цель этапа отображения – предоставить человеку нужную ему информацию с помощью устройств, способных воздействовать на его органы чувств.

6. На этапе воздействия информация используется для осуществления необходимых изменений в системе. Воздействие – сигналы, проводят регулирование или управление, вызывая изменения в самом объекте.

В наиболее общем случае типовые программные компоненты, входящие в состав информационной системы, включают [11]:

- диалоговый ввод-вывод;
- логику диалога;
- прикладную логику обработки данных;
- логику управления данными;
- операции манипулирования файлами и (или) базами данных.

2. 3. История и перспективы развития информационных систем

Согласно наиболее распространенному определению информационных систем можно перечислить основные ее составляющие:

1 – человек (персонал) в (автоматических системах отсутствует);

2 – данные;

3 – технологии обработки данных для достижения определенных человеком цели

4– инструменты и материалы, необходимые для реализации принятой технологии обработки данных (сбор, хранение, передача, обработку и представление пользователю результатов).

Исходя из обобщенного определения информационных систем, история возникновения информационных систем начинается с тех пор, когда у человека появилась необходимость обмениваться информацией (передавать данные) между себе подобными.

Наскальные рисунки есть технологии обработки передаваемой информации (данных) для ее передачи другим. Для реализации этой технологии необходим материал – камень и инструмент для обработки камня (изготовления надписей-сообщений).

Криптография также является технологией информационных систем, где преобразовывается текст послания (материал), с помощью криптографического кодирования (инструмента).

Простейшей наиболее распространенной информационной системой в средние века, конечно же, являлась почта. Где письма в конвертах – источники информации они же сообщения (материал) доставлялись потребителям с помощью транспортных средств и с участием человека (инструмент).

Наступает индустриальный век появляется телеграф – и информационная система «Почта» приобретает новую технологию – телеграф.

Наконец, эпоха компьютеризации и сетевых технологий преобразует нашу систему «Почта» до неузнаваемости – что-

бы переслать письмо на огромные расстояния, теперь достаточно нескольких минут.

Эпоха компьютеризации явилась огромным толчком в развитии технологий информационной системы по работе с данными.

Основные достижения в развитии технологий информационной системы по работе с данными отражены в появлении:

1 – файловых систем (подразумевается способ хранения данных и программы, реализующие работу с данными согласно этому способу хранения);

2 – баз данных и СУБД;

3 – новой технологии, которую называют *Большие данные*.

Предшественницей баз данных являются файловые системы

Файловые системы – это набор программ, которые выполняют для пользователей некоторые операции, например, создание отчетов. Каждая программа определяет свои собственные данные и управляет ими. Прежде всего, конечно, файлы применяются для хранения текстовых данных: документов, текстов программ и т. д. Такие файлы обычно образуются и модифицируются с помощью различных текстовых редакторов. Структура текстовых файлов обычно очень проста: это либо последовательность записей, содержащих строки текста, либо последовательность байтов, среди которых встречаются специальные символы (например, символы конца строки). Файлы с текстами программ, объектные модули. С точки зрения файловой системы, такие файлы также обладают очень простой структурой – последовательность записей или байтов. Аналогично обстоит дело с файлами, содержащими графическую и звуковую информацию. Одним словом, файловые системы обычно обеспечивают хранение слабо структурированной информации, оставляя дальнейшую структуризацию прикладным программам.

Важным понятием и технологическим элементом файловых систем, который затем перешел в базы данных и СУБД являются понятия: **данные и метаданные**.

Напомним, что метаданные – это информация о том, как расположены данные при хранении. Здесь опять можно загля-

нуть в далекое прошлое XVIII–XIV в., когда функционирование склада нуждалось в журналах, где фиксировалось наличие и месторасположение хранимых материалов.

Современная информационная система обязательно основана на базе данных, использует СУБД – программный комплекс, функции которого состоят в поддержке надежного хранения информации в памяти компьютера, выполнении различных преобразований информации и/или вычислений, предоставлении пользователям удобного и легко осваиваемого интерфейса. Обычно объемы информации, с которыми приходится иметь дело таким системам, достаточно велики, а сама информация имеет достаточно сложную структуру. Классическими примерами информационных систем являются банковские системы, системы резервирования авиационных или железнодорожных билетов, мест в гостиницах и т. д.

Параллельно с развитием аппаратной части информационных систем на протяжении последних лет происходит постоянный поиск новых, более удобных и универсальных, методов программно-технологической реализации информационных систем.

Можно выделить три наиболее существенных новшества, оказавших колоссальное влияние на развитие информационных систем в последние годы:

- *новый подход к программированию*: с начала 90-х годов объектно-ориентированное программирование фактически вытеснило модульное; до настоящего времени непрерывно совершенствуются методы построения объектных моделей. Благодаря внедрению объектно-ориентированных технологий программирования существенно сокращаются сроки разработки сложных информационных систем, упрощаются их поддержка и развитие;
- *благодаря развитию сетевых технологий* локальные информационные системы повсеместно вытесняются клиент-серверными и многоуровневыми реализациями;
- *развитие сети Интернет* принесло большие возможности работы с удаленными подразделениями, открыло широкие перспективы электронной коммерции, обслуживания покупателей через Интернет и многое другое. Более того, определенные

преимущества дает использование Интернет-технологий в интрасетях предприятия (так называемые интранет-технологии).

Наконец, появились технологии поддержки проектирования ИС (баз данных, СУБД) – это прежде всего, CASE-средства, под которыми понимаются программные средства, поддерживающие процессы создания и сопровождения информационных систем, включая анализ и формулировку требований, проектирование прикладного программного обеспечения и баз данных, генерацию кода, тестирование, документирование, обеспечение качества, конфигурационное управление и управление проектом, а также другие процессы.

Современные СУБД предоставляют развитые средства для работы с информацией, хранящейся в сети Интернет. Для обеспечения доступа клиентов сети Интернет к информации базы данных, находящейся на сервере используются возможности Интернет-технологий, размещающих на web-страницах запрашиваемые данные. Таким образом, объединение Интернет технологий и технологий СУБД имеют определенные преимущества одним из которых является обеспечение доступа из корпоративных интрасетей и обеспечение информационного обмена между СУБД, работающими на разных платформах (операционных системах и аппаратных средствах) [36].

Эффективность функционирования ИС зависит от ее архитектуры. В настоящее время широко применяется архитектура «клиент – сервер». Такая архитектура предполагает наличие компьютерной сети и распределенной базы данных, включающей корпоративную базу данных и персональную базу данных. Достоинством организации такой информационной системы является возможность централизованного хранения, обслуживания и коллективного доступа к общей информации. Архитектура клиент – сервер допускает различные варианты реализаций. Исторически первыми появились информационные системы с применением файл-сервера. В таких ИС использовались файлы базы данных. Далее появились структуры баз данных с использованием сервер базы данных, где производилось выполнение основного объема обработки данных. Корпоративная база дан-

ных функционирует под управлением сервера БД, например, Microsoft SQL Server или Oracle Server.

Для расширения возможностей клиентской части и серверной части создаются программы расширения. При организации такого взаимодействия могут использоваться следующие средства:

- языки сценариев;
- элементы управления ActiveX;
- динамические страницы;
- активные серверные страницы ASP и др.

Использование архитектуры клиент – сервер дает возможность постепенного наращивания информационной системы предприятия. Важным достоинством применения БД в ИС является обеспечение независимости данных от прикладных программ. Это дает пользователям не заниматься проблемами представления данных на физическом уровне: размещения данных в памяти, методов доступа к ним и др.

Будущее развитие информационных систем связано с развитием новых технологий, которые называются *Большие данные*.

Большие данные (*bigdata*) в информационных технологиях – совокупность подходов, инструментов и методов обработки структурированных и неструктурированных данных огромных объёмов для получения воспринимаемых человеком результатов, эффективных в условиях непрерывного прироста, распределения по многочисленным узлам вычислительной сети. К категории Большие данные (*bigdata*) относится информация, которую невозможно обрабатывать традиционными способами, в том числе структурированные данные, медиа- и случайные объекты [42].

Большие данные призваны совершать три операции. Во-первых, обрабатывать большие объёмы данных. Во-вторых, уметь работать с быстро поступающими данными в очень больших объёмах, т. е. данных не просто много, а их постоянно становится все больше и больше. В-третьих, они должны уметь работать со структурированными и плохо структурированными данными параллельно в разных аспектах. Таким образом, большие данные предполагают, что, на входе имеется поток не все-

гда структурированной информации и что из него можно извлечь больше чем одну идею.

Таким образом, при работе с большими данными процессы сбора, хранения, распределения, анализа данных становятся трудоемкими. Объемы больших данных постоянно растут, изменяясь от нескольких десятков терабайтов до петабайтов данных.

В качестве примера больших данных можно отметить информацию, поступающую с различных физических установок, например, датчиков температуры, давления или расхода. Эти датчики в режиме реального времени измеряют соответствующую физическую величину и измеренные данные поступают на пульт оператора по управлению технологическим процессом. Поступившие данные используются для решения задачи выбора управляющих решений, которые представляют собой сам процесс обработки информации в ИС.

Для того чтобы эффективно обрабатывать большие объёмы данных, необходимы особые технологии. К таким технологиям относятся средства массово-параллельной обработки неопределённо структурированных данных, прежде всего, решениями категории NoSQL, алгоритмами MapReduce, программными каркасами и библиотеками проекта Hadoop.

Термин «*большие данные*» широко распространился с 2009 г., а к 2010 г. появились продукты и решения, относящиеся к этому термину. Ожидается, что в будущем применение технологий больших данных окажет большое влияние на информационные технологии в производстве, здравоохранении, торговле, государственном управлении.

Вопросы и задания

1. Что такое информационная среда?
2. Дайте определение информационной системе управления?
3. Назовите 6 основных типов информационных систем?
4. Для чего служит диалоговая система обработки запросов?
5. Объясните на каких составляющих основаны экспертные системы?

6. Чем отличаются интеллектуальные системы от экспертных?
7. Перечислите основные составляющие системы передачи информации в качестве информационной системы.
8. Дайте определение понятию «сообщение».
9. Как различают сообщения по характеру измерения во времени?
10. Как различают сообщения по множеству?
11. Перечислите возможные виды сообщений.
12. На какие виды различают информацию?
13. Какие существуют этапы обращения информации?
14. Перечислите типовые программные компоненты, входящие в состав информационной системы.
15. Перечислите основные составляющие информационной системы.
16. Перечислите основные технологии информационных систем.
17. Что такое СУБД?
18. Что предполагает архитектура «клиент – сервер»?
19. Для чего создаются создаются программы расширения?
20. Назовите новые технологии информационных систем.
21. Приведите примеры информационных систем.
22. Что такое метаданные?
23. Что такое CASE-средства ?
24. Для чего используются элементы управления ActiveX?
25. Что такое NoSQL?

3. ПРЕДСТАВЛЕНИЕ ДАННЫХ В ИНФОРМАЦИОННЫХ СИСТЕМАХ

3.1. Системы счисления

Программно-аппаратный комплекс ИС включает :

- 1) информационную базу (хранилища информации),
- 2) процедуры, позволяющие накапливать, хранить, корректировать, осуществлять поиск, обработку и выдачу информации.

Передача информации происходит с помощью сообщений, передача или хранение которых в информационной системе сводится к передаче или хранению чисел [7, 13]. Числа можно выразить в какой-либо системе счисления.

Для позиционной системы счисления число определяется следующим образом:

$$Q = \sum_{i=1}^l a_i m^{i-1} = a_l m^{l-1} + a_{l-1} m^{l-2} + \dots + a_2 m^1 + a_1 m^0,$$

где m – основание системы счисления; i – номер разряда данного числа; l – количество разрядов; a_i – множитель, принимающий любые целочисленные значения в пределах от 0 до $m - 1$ и показывающий, сколько единиц i -го разряда содержится в числе.

Например, для десятичного ($m = 10$) числа 345 его полное значение рассчитывается следующим образом:

$$3 \cdot 10^2 + 4 \cdot 10^1 + 5 \cdot 10^0 = 345.$$

Римские числа являются примером полупозиционной системы образования числа: так, в числах IX и XI знак I обозначает в обоих случаях единицу (признак непозиционной системы), но, будучи расположенным слева от знака X (обозначающего десять), вычитается из десяти, а при расположении справа – прибавляется к десяти. В первом случае полное значение числа равно 9, во втором – 11.

В современной информатике используются в основном три системы счисления (все – позиционные): двоичная, шестнадцатеричная и десятичная.

Двоичная система счисления используется для кодирования дискретного сигнала, потребителем которого является вычислительная техника. Такое положение дел сложилось исторически, поскольку двоичный сигнал проще представлять на аппаратном уровне. В этой системе счисления для представления числа применяются два знака – 0 и 1.

Шестнадцатеричная система счисления используется для кодирования дискретного сигнала, потребителем которого является хорошо подготовленный пользователь – специалист в области информатики. В такой форме представляется содержимое любого файла, затребованное через интегрированные оболочки операционной системы, например, средствами Norton Commander в случае MS DOS. Используемые знаки для представления числа – десятичные цифры от 0 до 9 и буквы латинского алфавита – A, B, C, D, E, F.

Десятичная система счисления используется для кодирования дискретного сигнала, потребителем которого является так называемый конечный пользователь – неспециалист в области информатики (очевидно, что и любой человек может выступать в роли такого потребителя). Используемые знаки для представления числа – цифры от 0 до 9.

Соответствие между первыми несколькими натуральными числами всех трех систем счисления представлено в таблице перевода (табл. 2).

Таблица 2

Таблица перевода

Десятичная система	Двоичная система	Шестнадцатеричная система
0	0	0
1	1	1
2	10	2

Продолжение табл. 2.

3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10

Для различения систем счисления, в которых представлены числа, в обозначение двоичных и шестнадцатеричных чисел вводят дополнительные реквизиты:

для двоичных чисел – нижний индекс справа от числа в виде цифры 2 или букв В либо b (binary – двоичный), либо знак В или b справа от числа. Например, $101000_2 = 101000b = 101\ 000B$;

- для шестнадцатеричных чисел – нижний индекс справа от числа в виде числа 16 или букв Н либо h (hexadecimal – шестнадцатеричный), либо знак Н или h справа от числа. Например, $3AB_{16} = 3ABH = 3ABh$.

Для перевода чисел из одной системы счисления в другую существуют определенные правила. Они различаются в зависимости от формата числа – целое или правильная дробь. Для вещественных чисел используется комбинация правил перевода для целого числа и правильной дроби.

Правила перевода целых чисел. Результатом перевода целого числа всегда является целое число.

Перевод из десятичной системы счисления в двоичную и шестнадцатеричную:

а) исходное целое число делится на основание системы счисления, в которую переводится (на 2 – при переводе в двоич-

ную систему счисления или на 16 – при переводе в шестнадцатеричную); получается частное и остаток;

б) если полученное частное меньше основания системы счисления, в которую выполняется перевод, процесс деления прекращается, переходят к шагу в. Иначе над частным выполняют действия, описанные в шаге а;

в) все полученные остатки и последнее частное преобразуются в соответствии с таблицей перевода в цифры той системы счисления, в которую выполняется перевод;

г) формируется результирующее число: его старший разряд – полученное последнее частное, каждый последующий младший разряд образуется из полученных остатков от деления, начиная с последнего и кончая первым. Таким образом, младший разряд полученного числа – первый остаток от деления, а старший – последнее частное.

Пример: Выполнить перевод числа 19 в двоичную систему счисления:



Таким образом, $19 = 10011_2$.

Пример: Выполнить перевод числа 19 в шестнадцатеричную систему счисления:

$$\begin{array}{r}
 \begin{array}{|l} 16 \\ \hline 1 \end{array} \\
 -19 \\
 \hline
 16 \\
 \hline
 3
 \end{array}
 \begin{array}{l}
 \diagup \\
 \diagdown
 \end{array}
 \begin{array}{l}
 1 \\
 3
 \end{array}
 \begin{array}{l}
 \diagdown \\
 \diagup
 \end{array}
 \begin{array}{l}
 1 \\
 3
 \end{array}$$

1 3 – результирующее число

Таким образом, $19 = 13_{16}$.

Пример: Выполнить перевод числа 123 в шестнадцатеричную систему счисления:

$$\begin{array}{r}
 \begin{array}{|l} 16 \\ \hline 7 \end{array} \\
 -123 \\
 \hline
 112 \\
 \hline
 11
 \end{array}
 \begin{array}{l}
 \diagup \\
 \diagdown
 \end{array}
 \begin{array}{l}
 7 \\
 B
 \end{array}
 \begin{array}{l}
 \diagdown \\
 \diagup
 \end{array}
 \begin{array}{l}
 7 \\
 B
 \end{array}$$

7 B – результирующее число.

Здесь остаток 11 преобразован в шестнадцатеричную цифру B (см. табл.2) и после этого данная цифра вошла в число. Таким образом, $123 = 7B_{16}$.

3.2. Представление информации в компьютере

Минимальной единицей информации в компьютере является бит. Бит может принимать значения 0 или 1.

Группа из восьми бит представляет собой байт. Биты или разряды в байте пронумерованы от 0 до 7 справа налево:

Номера битов: 7 6 5 4 3 2 1 0.

Каждый бит имеет свой вес: 2^i , соответственно для нулевого разряда- 2^0 , для 1-го – 2^1 и т. д. до 2^7 .

Таким образом, максимальное число (без знака), которое можно записать в байт, формируется при установке всех бит в единицу: 11111111. Оно равно $1 * 2^0 + 1 * 2^1 + 1 * 2^2 + 1 * 2^3 + 1 * 2^4 + 1 * 2^5 + 1 * 2^6 + 1 * 2^7 = 255$. Соответственно минимальное число в байте – это 0000 0000. Остальные числа образуются комбинацией из единиц и нулей в различных битах:

00000001 – 1;

00000010 – 2;

00000011 – 3 и т. д.

Таким образом, с помощью 8 бит можно запомнить или представить любое из 256 различных целых чисел без знака от 0 до 255.

Два байта или шестнадцать бит представляют собой слово.

Биты в слове пронумерованы от 0 до 15 справа налево:

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

Например: 10101010 1 1 1 1 0 0 0 0.

В этом слове байт с разрядами от 8 до 15 является старшим байтом и имеет значение 10 101 010, байт с разрядами от 0 до 7 является младшим байтом и имеет значение 11 110 000. (Для процессоров Intel в оперативной памяти расположен сначала младший байт, а потом старший).

Все вычисления в компьютере производятся в двоичной системе счисления, т. е. в двоичном коде. Шестнадцатеричная форма представления чисел является краткой записью чисел в двоичном коде. Для этого четыре двоичных разряда записываются одним шестнадцатеричным разрядом. Кроме этих двух форматов представления чисел могут использоваться десятичный, двоично-десятичный (BCD) и ASCII-коды.

Двоично-десятичные числа (BCD). Двоично-десятичные числа записываются шестнадцатеричными цифрами. Двоично-десятичные числа могут быть представлены в упакованном или распакованном формате. Упакованные BCD-числа, вероятно, больше распространены, они сохраняют две десятичные цифры в каждом байте — одну цифру в старших четырех битах и другую — в младших четырех битах непосредственно в двоичном виде. Поскольку 4 бит могут представлять двоичную величину с диапазоном от 0 до 15, использование 4 бит для представления величин с диапазоном от 0 до 9 оставляет немного места в каждом байте. (При другом рассмотрении можно представить, что BCD-байт может содержать величины в диапазоне от 0 до 99, в то время как двоичный байт может нормально представлять значения от 0 до 255.)

Неупакованные BCD-числа обычно используются как промежуточная форма для преобразования упакованных BCD-чисел в ASCII-символы и обратно.

Пример: десятичное число 311_{10} будет записано в двоичной системе счисления в двоичном коде как $1\ 0011\ 0111_2$, а в двоично-десятичном коде как $0011\ 0001\ 0001_{\text{BCD}}$.

ASCII – код

Ввод и вывод информации в компьютере производится в виде символов. Для обмена данными между МП и принтером, клавиатурой или дисплеем используются ASCII коды. ASCII код (американский стандартный код для обмена информацией) используется для кодирования алфавитно-цифровой информации в компьютере.

Например, цифры от 0 до 9 имеют ASCII коды от 48 до 57 (30h до 39h). Заглавные буквы от «А» до «Z» имеют ASCII коды от 65 до 90 (от 41h до 5Ah), прописные буквы от «a» до «z» – от 97 до 122 (от 61h до 7Ah). Для вывода результата на дисплей нужно его представить в ASCII коде. Например: требуется вывести на дисплей число 17h. Для этого сначала его нужно распаковать (разложить проще говоря на составляющие символы, а для шестнадцатеричной системы это 1 2 3 4 ABCDEF, а затем преобразовать в ASCII код (для 1 – это 31h, для 2 – 32h и т. д.).

Unicode – это универсальная международная кодировка, которая предусматривает выделение для набора символов каждого языка определенной непрерывной последовательности двоичных чисел. Символы Unicode хранятся в памяти компьютера в виде 16-разрядных чисел (в отличие от символов ASCII-кода, которые хранятся в виде 8-разрядных чисел), что позволяет представить свыше 60 тысяч различных символов; расплатой за универсальность Unicode является снижение плотности хранения информации (на каждый символ расходуется два байта памяти вместо одного).

Набор символов латинского алфавита (т. е. символов английского языка) и математические символы считаются в Unicode основными, обозначаются в документации как BasicLatin и размещаются в диапазоне 0020h–007Eh. Преобразование латинских символов из формата Unicode в ASCII-код сводится, таким образом, к простому отсечению старшего байта.

Символы русского языка (Cyrillic) размещаются в диапазоне 0410h–044Fh, как показано в табл. 3 (буква «Ё» вне этого диапазона, в наборе дополнительных символов кириллицы).

Таблица 3

Представление латинских и математических символов
в формате Unicode

Младшая цифра	Старшие цифры					
	002	003	004	005	006	007
0		0	@	P	~	P
1	!	1	A	Q	A	Q
2	“	2	B	R	B	R
3	#	3	C	S	C	S
4	\$	4	D	T	D	T
5	%	5	E	U	E	U
6	&	6	F	V	F	V
7	`	7	G	W	G	W
8	(	8	H	X	H	X
9	)	9	I	Y	I	Y
A	*	:	J	Z	J	Z
B	+	;	K	[	K	{
C	,	<	L	\	1	
D	-	=	M	]	M	}
E	.	>	N	^	N	-
F	/	?	0	_	O	

Таблица 4

Представление символов русского алфавита
в формате Unicode

Младшая цифра	Старшие цифры					
	002	003	004	005	006	007
0		A	P	A	P	
1	Ё	Б	С	Б	С	Ё
2		В	Т	В	Т	
3		Г	У	Г	У	

Продолжение табл. 4

4		Д	Ф	Д	Ф	
5		Е	Х	Е	Х	
6		Ж	Ц	Ж	Ц	
7		З	Ч	З	Ч	
8		И	Ш	И	Ш	
9		Й	Щ	Й	Щ	
А		К	Ь	К	Ъ	
В		Л	Ы	Л	Ы	
С		М	Ъ	М	Ь	
Д		Н	Э	Н	Э	
Е		О	Ю	О	Ю	
Ф		П	Я	П	Я	

3. 3. Представление отрицательных чисел и графической информации

Отрицательные числа в компьютере представляются в дополнительном коде (получение дополнительного кода: инвертировать все биты и прибавить 1). Так, для числа 65, имеющему представление в двоичном коде 01000001, дополнительный код имеет вид 10 111 111, это и есть – 65. Знаковый – самый старший разряд. Таким образом, для байтов со знаком минимальное число –128, а максимальное +127.

Графическая информация (в отличие от текстовой) представляется на экране в виде растрового изображения, которое формируется из определенного количества строк, содержащих определенное количество точек (*пикселей*), имеющих свой цвет, заданный специальным кодом. В процессе кодирования изображения производится его пространственная дискретизация – построение изображения из большого количества отдельных цветных точек. Качество кодирования изображения определяется разрешением изображения и глубиной цвета. *Разрешение изоб-*

ражения (количество пикселей в единице длины или в устройстве) может принимать различные значения, например 300 точек на дюйм у фотографий. *Глубина цвета* (количество бит, используемых для кодирования цвета пикселя) влияет на количество различных оттенков цветов. Наиболее распространены четыре глубины цвета:

- 8 бит — 256 (2^8) цветов;
- 16 бит (режим *High color*) – 65 536 (2^{16}) цветов;
- 24 бит (режим *True color*) – 16 777 216 (2^{24}) цветов;
- 32 бит — (режим *True color*) – 16 777 216 (2^{24}) цветов.

(Заметим, что человеческий глаз больше, чем 2^{24} цветов не различает). Иначе кодируется звуковая информация. Любая звуковая волна имеет непрерывно меняющиеся частоту и амплитуду. При увеличении амплитуды сигнала усиливается громкость звука, а при увеличении его частоты повышается тональность. Для обработки в ЭВМ звуковая (аналоговая) информация кодируется в виде последовательности цифровых импульсов в процессе *дискретизации звука*. *Дискретизация* – это преобразование непрерывных звуков (или изображений) в набор дискретных значений в закодированной форме. Качество кодирования звука определяется глубиной кодирования, частотой дискретизации и количеством звуковых каналов. Современные звуковые системы обеспечивают 16-битную *глубину кодирования* звука, при этом количество различных уровней сигнала составляет 65 536 (2^{16}). *Частота дискретизации* (количество изменений уровня сигнала в единицу времени) может лежать в диапазоне от 8 до 48 кГц. Частота 8 кГц соответствует качеству радиотрансляции, а частота 44,1 кГц – качеству звукового компакт-диска. *Количество звуковых каналов* может лежать в диапазоне от 1 (монозвук) или 2 (стереозвук) до 6 (кинотеатральный звук). Например, стереозвук с аудио-CD продолжительностью 1 с в закодированном виде будет занимать 176 400 байт:

$$2 \text{ (стерео)} \cdot 16 \text{ (бит)} \cdot 44\,100 \text{ (Гц)} = 1\,411\,200 \text{ (бит)}.$$

3. 4. Организация информации в компьютере

3.4.1. Файлы

Вся информация, обрабатываемая ВС, содержится в ней на устройствах: ОП, ВП, устройства ввода-вывода. При этом на любом из устройств информация хранится в виде длинной битовой строки, т. е. в виде последовательности нулей и единиц. Длина одной такой битовой строки может составлять многие сотни Гбит ($1Г = (1024)^3$). Так как работать с такой длинной строкой чрезвычайно неудобно, то она разделяется на поименованные части разной длины, называемые *файлами*. Более точное определение: *файл* – часть пространства носителя ВП (разрывная или непрерывная), которой присвоено имя, уникальное для данной ЭВМ.

Информация на носителе делится на части – файлы по смысловому принципу. Например, один файл может содержать текст исходной программы, второй – ее объектный модуль, а третий – загрузочный модуль. Кроме того, в большинстве современных ОС каждое устройство ввода-вывода также считается файлом. Что касается ОП, то информация в ней делится не на файлы, а на сегменты.

Файл является достаточно крупной единицей информации. Для удобства работы с ним битовая строка, образующая файл, делится на части, называемые *записями файла*. При этом файл может рассматриваться как несвязанный линейный список, расположенный на конкретном носителе информации. Заметим, что разные программы могут использовать разное разбиение одного и того же файла на записи.

В реальной ВС одновременно существуют сотни или даже тысячи файлов. Для того чтобы ориентироваться в этом, пользователь прибегает к помощи *файловой структуры системы*.

3.4.2. Файловые системы

Физический файл существует в ВС не обособленно, а как часть более крупной информационной единицы, называемой *файловой системой* (ФС). Эту систему образуют файлы и таб-

лицы, расположенные на конкретном носителе ВП или расположенные на его части (в разделе или на логическом диске).

Несмотря на большое количество существующих типов ФС, между этими системами много общего. Во-первых, кроме самих физических файлов любая ФС содержит информацию о размещении этих файлов на носителе. Во-вторых, для каждого файла ФС содержит его свойства, называемые *атрибутами файла*. Основные различия между ФС сводятся к различиям характеристик:

- структура ФС;
- состав атрибутов файла и место их хранения;
- размещение элементов ФС на носителе;
- место хранения информации о размещении файла;
- длина простого имени файла.

Как правило, все современные ОС имеют соответствующие системы управления файлами. Назначение *системы управления файлами* – организация более удобного доступа к данным, организованным как файлы в соответствии с конкретной файловой системой.

Во-первых, через систему управления файлами связываются по данным все системные обрабатывающие программы.

Во-вторых, с помощью этой системы решаются проблемы централизованного распределения дискового пространства и управления данными. В-третьих, благодаря использованию той или иной системы управления файлами пользователям предоставляются следующие возможности:

- создание, удаление, переименование (и другие операции) именованных наборов данных (именованных файлов) из своих программ или посредством специальных управляющих программ, реализующих функции интерфейса пользователя с его данными и активно использующих систему управления файлами;
- работа с не дисковыми периферийными устройствами как с файлами;
- обмен данными между файлами, между устройствами, между файлом и устройством (и наоборот);
- работа с файлами с помощью обращений к программным модулям системы управления файлами (часть API ориентирована именно на работу с файлами);

– защита файлов от несанкционированного доступа.

Таким образом, термин «*файловая система*» определяет, прежде всего, принципы доступа к данным, организованным в файлы. А термин «*система управления файлами*» следует употреблять по отношению к конкретной реализации файловой системы, т. е., это комплекс программных модулей, обеспечивающих работу с файлами в конкретной операционной системе. (Есть версия системы управления файлами с принципами FAT и для Windows и для Linux т. д.).

При работе с файлами, как правило, вводятся механизмы структурирования, прежде всего, организуются иерархические отношения. Для этого вводится понятие *каталога* (directory). Каталог содержит информацию о данных, организованных в виде файлов. Другими словами, в каталоге должны содержаться *дескрипторы файлов*. Если файлы организованы на блочном устройстве, то именно с помощью каталога система управления файлами будет находить адреса тех блоков, в которых размещены искомые данные. Причем очевидно, что каталогом может быть не только специальная системная информационная структура, которую часто называют *корневым каталогом*, но и сам файл. Такой *файл-каталог* должен иметь специальное системное значение; система управления файлами должна его выделять на фоне обычных файлов. Файл-каталог часто называют *подкаталогом* (subdirectory). Если файл-каталог содержит информацию о других файлах, то, поскольку среди них также могут быть файлы-каталоги, мы получаем возможность строить почти ничем не ограниченную иерархию.

Более того, введение таких файловых объектов, как файлы-каталоги, позволяет не только структурировать файловую систему, но и решить проблему ограниченного количества элементов в корневом каталоге. Ограничений на количество элементов в файле-каталоге практически нет, поэтому можно создавать каталоги чрезвычайно большого размера).

Принцип (метод) доступа определяет конкретную организацию доступа к данным, хранящимся на диске. С позиции

определения правил распределения файловой информации на носителе различают

- связанные распределения файлов;
- несвязные распределения файлов.

При связанном распределении информационные элементы файла (записи) размещаются на физическом носителе непрерывно по адресуемым участкам (для диска – это сектора). Такой способ характерен для непрерывных файлов, характеризуется малым временем доступа и эффективным использованием дискового пространства.

Главные недостатки – трудности в расширении файла и доступе к отдельным его элементам

Ведение учета записей и использование дискового пространства ведется на уровне секторов. Более эффективным является стратегия блочного распределения файлов, в том числе и непрерывного. В этом случае смежные сектора объединяются в блоки фиксированного или переменного размера.

При несвязном распределении файлу выделяется память элементами на произвольных участках носителя. При этом выделение может быть секторами или блоками. В этом случае система управления ведет список свободных блоков или секторов, которые выделяются по мере необходимости. При каждом выделении проводится исключение из блока свободных и привязка выделенного сектора (блока) запрашивающему файлу.

Способы блочной организации файлов.

1. Блочно-последовательная структура (цепочки блоков).

Признаки (особенности):

- 1 – адресная часть связана с набором данных,
- 2 – в связи с этим для нахождения адреса следующего блока его необходимо считать в оперативную память.
- 3 – простота манипуляции.

Главным недостатком способа является отсутствие возможности прямого обращения к блоку внутри последовательности блоков

Для исключения указанного недостатка используется способ распределения блочно-индексного распределения

2. Блочно-индексное распределение файлов. (Цепочки индексных блоков)

Адресация к каждому блоку происходит путем указания номера индексной таблицы и индекса в этой таблице.

Достоинства:

1. Разнесение адресов и наборов данных, что позволяет манипулировать адресами независимо от блоков данных (перемешать их между дисковой и оперативной памятью).

2. Возможность последовательного размещения файла в пределах выделенной индексной таблицы;

3. Прямая адресация или обращение к отдельным блокам данных (записям файла).

Распределение по карте файлов (таблице поблочного отображения)

Здесь правило выделения блока следующее:

Файлу выделяется первый блок, и номер этого блока заносится в директорию (каталог). По мере расширения файлу выделяются дополнительные блок из числа свободных, причем один и тот же блок не используется разными файлами.

3.4. 3. *Файловая система FAT*

3.4.3.1. *Исследование структуры логического диска файловой системы FAT*

Файловая система FAT (FAT12) (FileAllocationTable) была разработана Биллом Гейтсом и Марком МакДональдом в 1977 г. и первоначально использовалась в операционной системе 86-DOS. В дальнейшем 86-DOS была приобретена Microsoft. FAT была предназначена для работы с гибкими дисками размером менее 1 Мбайта, и вначале не предусматривала поддержки жестких дисков. FAT32 – усовершенствованная версия файловой системы VFAT, поддерживающая жесткие диски объемом до 2 терабайт. Впервые файловая система FAT32 была включена в состав ОС Windows 95 OSR 2. В FAT32 были расширены атрибуты файлов, позволяющие теперь хранить время и дату создания, модификации и последнего доступа к файлу или каталогу.

FAT(FAT32) является наиболее распространенной файловой системой и ее в той или иной степени поддерживают большинство современных ОС. Благодаря своей универсальности FAT может применяться на томах, с которыми работают разные операционные системы.

Логический диск (том) файловой системы типа FAT(FAT12,FAT16,VFAT,FAT32) состоит из четырех основных областей распределенных в следующем порядке:

1. Резервная область.
2. Область таблиц размещения файлов FAT 1 и FAT 2.
3. Область корневого каталога (не существует в FAT 32).
4. Область файлов и каталогов.

3.4.4. Организация данных на логических дисках системы

В первом секторе логического диска (сектор № 1) – загрузочная программа (Bootstrap, она же Bootsector (BS)) и блок параметров BIOS(BIOS Parametor – Block (BPB)). Начальный участок загрузочного сектора для всех типов FAT одинаковый (описание в табл. 5).

Таблица 5

№ п/п	Наименование элемента	Смещение	Размер	Описание
1	BS_jmpBoot	00h	3	Инструкция перехода (jmp) на загрузочный код
2	BS_OEM Name	03h	8	Текстовая строка-фирма, № версии ОС
3	BPB_bytsPerSec	0Bh	2	Число байт в секторе
4	BPB_SecPerClus	0Dh	1	Число секторов в кластере
5	BPB_RSVD SecCnt	0Eh	2	Число резервных секторов в резервной области раздела
6	BPB_NumFats	10h	1	Число таблиц копий FAT (в разделе всегда две)

Продолжение табл. 5.

№ п/п	Наименование элемента	Смещение	Размер	Описание
7	BPB_RootEntCnt	11h	2	Для FAT 12, FAT 16: количество 32-байтных дескрипторов файлов в корневом каталоге. При исполнении FAT 16 = 512 записей, для FAT 32 = 0
8	BPB_Totsec 16	13h	2	Общее число секторов в разделе. Если данное поле имеет значение 0, то число секторов задается в поле BPB_TotSec 32
9	BPB_Media	15h	1	Тип носителя информации
10	BPB_FAT Sz 16	16h	2	Для FAT 12, FAT 16 – количество секторов, занимаемых одной копией FAT
11	BPB_SecPerTrk	18h	2	Число секторов на дорожке
12	BPB_NumHeads	1Ah	2	Число головок
13	BPB_MiddSec	1Ch	4	Число скрытых секторов, предшествующих разделу, содержащему данный том. Ноль для носителей, не подлежащих разбиению на разделы.

Продолжение табл. 5.

№ п/п	Наименование элемента	Смещение	Размер	Описание
14	BPB_TotSec	20h	4	Общее число секторов в разделе. Если в разделе свыше 65 535 секторов, то используется данный элемент.

Таблица 6

Структура загрузочного сектора, начиная со смещения
24h для FAT 12 и FAT 16

№ п/п	Наименование элемента	Смещение	Размер	Описание
1	BS_DrvNum	24h	1	Номер дисководов для гибких от 0 до 3-х, для жестких с 20 h
2	BS_Reserved	25h	1	Зарезервирован для Windows NT
3	BS_BootSig	26h	1	Признак расширенной загрузочной записи. Показывает, что следующие 3 поля присутствуют (29h)
4	BS_Vol ID	27h	4	Номер логического диска
5	BS_VolLab	2Bh	11	Метка диска
6	BS_FilsysType	36h	8	Тип файловой системы

Таблица 7

Структура загрузочного сектора, начиная со смещения
24h для FAT 32

№ п/п	Наименование элемента	Смещение	Размер	Описание
1	BPB_FAT Sz 32	24h	4	Количество секторов, занимаемых одной копией FAT
2	BPB_ExtFlags	28h	2	Номер активной FAT
3	BPB_FsVer	2Ah	2	Номер версии FAT 32
4	BPB_BootClus	2Ch	4	Номер кластера для первого кластера корневого каталога (2)
5	BPB_FSINFO	30h	2	Номер сектора структуры FS INFO в резервной области логического диска
6	BPB_BK BootSec	32h	2	Номер сектора с резервной копией загрузочного сектора
7	BPB_Reserved	34h	12	Зарезервировано
8	BS_DrvNum	40h	1	Номер дискового
9	BS_Reserved1	41h	1	
10	BS_BootSig	42h	1	
11	BS_Vol ID	43h	4	Номер логического диска
12	BS_VolLab	47h	11	Метка
13	BS_FileSysType	52h	8	Тип файловой системы
14	...	1FE 1FF		Код 55 h Код AAh сигнатура

Заметим, что загрузочная запись для диска с FAT 32 занимает не один сектор как в FAT 12, а три.

3.4.5 Структура FS INFO

Структура FSINFO (только в FAT 32) размещается в первом после резервной области секторе и содержит информацию о количестве свободных кластеров на диске и номере первого свободного кластера в таблице FAT.

Таблица 8

№ п/п	Наименование элемента	Смещение	Размер	Описание
1	FSI_lead sig	000h	4	Сигнатура 416152 52h – признак FS INFO
2	FSI_Reserved1	004h	480	Зарезервировано
3	FSI_Strucsig	1E4h	4	614172 72h – сигнатура
4	FSI_FreeCount	1E8h	4	Текущее число свободных кластеров. Если это число 0FFFFFFF, то количество свободных кластеров неизвестно, его нужно вычислять
5	FSI_NxtFree	1ECh	4	Содержит номер кластера, с которого дисковый драйвер должен начинать поиск свободных кластеров. Если 0FFFFFFF, то поиск с кластера номер 2
6	FSI_Reserved 2	1F0h	12	Поле зарезервировано
7	FSI_Trail Sig	1FCh	4	Сигнатура – признак конца FS INFO

3.4.6. Назначение и внутренняя организация таблиц размещения файлов

При использовании FAT область данных логического диска разделена на участки одинакового размера – *кластеры*.

Число секторов в кластере должно быть кратно 2^N (1, 2, ..., 64). Размер кластера зависит от типа используемой системы FAT и объема логического диска.

Кластеры нумеруются, начиная с 2-х. Каждому кластеру поставлен в соответствие элемент таблицы FAT.

Первые 2 элемента FAT – резервные.

Таблица 9

Специальный код элементов FAT

Значение кода	FAT 12	FAT 16	FAT 32
Свободный кластер	0	0	0
Дефектный кластер	FF7h	FFF7h	FFFFFF7h
Последний кластер в списке	FF8h – FFF	FFF8h - FFFF	FFFFFF8h – FFFFFFFh

3.4.7. Каталог (Directory)

Каталог – это таблица, в которой каждому файлу на диске, включая подкаталог, соответствует одна запись размера 32 байта.

Корневой каталог (Root directory) – главный каталог диска, с которого начинается дерево подкаталогов.

В FAT 12, FAT 16 - специальное место с фиксированным размером (16 Кб) рассчитанное на хранение 512 элементов

В FAT 32 – файл произвольного размера.

Таблица 10

Структура элемента каталога

№ п/п	Наименование поля	Смещение	Размер	Описание
1	DIR_Name	00h	11	Краткое имя файла
2	Dir_Attr	0Bh	1	Атрибуты файла
3	DIR_NT Res*	0Ch	1	Зарезервировано для NT
4	DIR_CrtTimeTenth*	0Dh	1	Поле уточнения времени создания
5	DIR_CrtTime	0Eh	2	Время создания
6	DIR_CrtDate*	10h	2	Дата создания
7	DIR_LstAccDate*	12h	2	Дата последнего обращения
8	DIR_FST Class HI*	14h	2	Старшее слово номера первого кластера файла
9	DIR_Wrt	16h	2	Время выполнения последней операции записи в файл
10	DIR_WrtDate	18h	2	Дата выполнения последней операции записи в файл
10	DIR_FstClusterLo	1Ah	2	Младшее слово номера 1-го кластера файла
11	DIR_FileSize	1Ch	4	Размер файла в байтах

* – только для FAT 32

1-й байт короткого имени DIR_Name[0] выполняет функции признака занятости элемента каталога.

Если DIR_Name[0] = E5h, то элемент каталога свободен.

Если $\text{DIR_Name}[0] = 00$, то является началом чистой области каталога.

3.4.8. Длинное имя файла (LFN)

Записывается в каталог первым. За ним – стандартный описатель файла, содержащий укороченный по специальному алгоритму вариант этого имени. Признак длинного имени файла – это наличие единиц в разрядах (0–3) байта атрибута.

Порядок записи в каталог описания файла с длинным именем

(Начало каталога)
. . .
Последний фрагмент длинного имени
. . .
Второй фрагмент длинного имени
Первый фрагмент длинного имени
Стандартный описатель файла
. . .
Конец каталога

Таблица 11
Структура элемента каталога, хранящего фрагмент
длинного имени

№ п/п	Наименование поля	Смещение	размер	Описание
1	DIR_Counter	00h	1	Номер фрагмента
2	Dir_Lname 1	01h	10	Первый уча- сток фраг- мента длин- ного имени
3	DIR_Attr	0Bh	1	Атрибут файла

Продолжение табл.11.

№ п/п	Наименование поля	Смещение	размер	Описание
4	Dir_Flagf	0Ch	1	Байт флагов
5	DIR_ChkSum	0Dh	1	Контрольная сумма корот- кого имени
6	DIR_Lname 2	0Eh	12	Второй уча- сток фраг- мента длин- ного имени
7	DIR_First	1Ah	2	Номер перво- го кластера
8	DIR_Lname 3	1Ch	4	Третий уча- сток фраг- мента длин- ного имени

Задача. На рис. 6 представлена область данных, состоящая из 81h, 82 h, 83 h, 84 h, 85 h, 86 h, 87 h и 88 h кластеров.

Показано, что запись файла начинается с кластера 87 h, вторая часть содержится в кластере 88 h, третья – в кластере 81h , четвертая – в кластере 84 h, пятая – в кластере 82 h. Кластер 85 h испорчен. Кластер 86 h свободен. Требуется определить номера элементов FAT32 и их содержимое, отражающее данное заполнение области данных.

81 h		82 h		83 h		84 h	
3-я Часть		5-я часть		Пу- стой		4-я Часть	
85 h		86 h		87 h		88 h	
Испорчен		Пустой		1-я часть		2-я часть	

Рис. 6. Исходные данные

На рис.7 представлен правильный ответ на данный тест.

81 h	82 h	83 h	84 h
0000084 h	FFFFFFF h	0000000 h	0000082 h
85 h	86 h	87 h	88 h
FFFFFFF7 h	0000000 h	0000088 h	0000081 h

Рис.7. Номера элементов FAT32 и их содержимое

3.4.9. Логическая структура жесткого диска. Организация данных на жестком диске

Пространство на жестком диске может быть организовано в виде одного или нескольких разделов. Разделение всего дискового пространства на разделы полезно по следующим причинам. Во-первых, это структурирует хранение данных. Например, выделение отдельного раздела под операционную систему и программное обеспечение и другого раздела под данные пользователей позволяет отделить последние от системных файлов и не только повысить надежность системы, но и сделать более удобным ее обслуживание. Во-вторых, на каждом разделе может быть организована своя файловая система, что иногда бывает необходимо. Наконец, в ряде случаев на компьютере может потребоваться установка более одной операционной системы.

В самом первом секторе НЖМД, т. е. в секторе с координатами 0–0–1 располагается структура данных, несущая информацию о логической организации диска. Эта структура получила название *главной загрузочной записи* (MasterBootRecord, MBR). MBR содержит программу, с помощью которой можно проанализировать логическую организацию диска, а также найти и загрузить в оперативную память программу загрузки операционной системы. Программа, расположенная в MBR, носит название *внесистемного загрузчика* (Non-SystemBootstrap, NSB). По смещению 0x1 BE в MBR находится таблица разде-

лов,

(PartitionTable, PT), которая занимает 64 байта и содержит четыре элемента. Каждый элемент этой таблицы описывает один раздел.

Последние два байта MBR имеют значение 55AA_(h), то есть чередующиеся значения 0 и 1. Эта сигнатура выбрана для того, чтобы проверить работоспособность всех линий передачи данных. Значение 55AA_(h), присвоенное последним двум байтам, имеется во всех загрузочных секторах.

Таблица 12

Внутренняя структура 1-го сектора жесткого диска.

Смещение	Размер	Описание
000 – 18Dh	446	Загрузочная запись (NSB)
1BE – 1CDh	16	Элемент 1-го раздела
1CE – 1DDh	16	Элемент 2-го раздела
1DE – 1EDh	16	Элемент 3-го раздела
1EE – 1FDh	16	Элемент 4-го раздела
1FE – 1FFh	2	Сигнатура 0AA55h

Структура каждого элемента PT

Первым байтом в элементе таблицы разделов идет флаг *активности раздела* BootIndicator (значение 0 – не активен, 128 (80_(h)) – активен). Он позволяет определить, является ли данный раздел системным загрузочным. В результате процесс загрузки операционной системы осуществляется путем загрузки первого сектора с такого активного раздела и передачи управления на расположенную в нем программу, которая и продолжает загрузку. Активным может быть только один раздел, и это обычно проверяется программой NSB, расположенной в MBR. За флагом активности раздела следует байт номера головки, с которой начинается раздел. За ним следуют два байта, означающие соответственно номер сектора и номер цилиндра загрузочного сектора, где располагается первый сектор загрузчика операционной системы. Затем следует кодовый идентификатор SystemID (длиной в один байт), указывающий на принадлежность данного раздела к той или иной операционной системе и на установку в этом разделе соответствующей файловой системы.

Таблица 13

Формат элемента таблицы раздел

Название записи элемента PartitionTable	Смещение	Длина (байт)
Флаг активности раздела	00 h	1
Номер головки начала раздела	01 h	1
Номер сектора и номер цилиндра загрузочного сектора раздела	02 h	2
Кодовый идентификатор операционной системы	04 h	1
Номер головки конца раздела	05 h	1
Номер сектора и номер цилиндра последнего сектора раздела	06 h	2
Младшее и старшее слово относительно номера начального сектора	08 h	4
Младшее и старшее слово относительно размера раздела в секторах	0C h	4

Если в поле кода раздела записан 0, то описание считается пустым, т. е., он не определяет на диске никакого раздела.

Разделы диска могут быть двух типов: *первичные* (primary) и *расширенные* (extended). Максимальное число первичных разделов равно четырем. Если первичных разделов несколько, то только один из них может быть активным. Именно загрузчику, расположенному в активном разделе, передается управление при включении компьютера с помощью внесистемного загрузчика. Для DOS-систем и иных операционных систем, использующих спецификации DOS, остальные первичные разделы в этом случае считаются невидимыми (hidden). Так ведут себя и операционные системы Windows 9x.

Согласно принятым спецификациям на одном жестком диске может быть только один расширенный раздел, который, в свою очередь, может быть разделен на большое количество подразделов – *логических дисков* (logical disks). Расширенный раздел содержит вторичную запись MBR (Secondary MBR, SMBR), в состав которой вместо таблицы разделов входит ана-

логичная ей *таблица логических дисков* (Logical Disks Table, LDT). Таблица LDT описывает размещение и характеристики раздела, содержащего единственный логический диск, а также может специфицировать следующую запись SMBR. Следовательно, если в расширенном разделе создано k логических дисков, то он содержит k экземпляров SMBR, связанных в список. Каждый элемент этого списка описывает соответствующий логический диск и ссылается (кроме последнего) на следующий элемент списка.

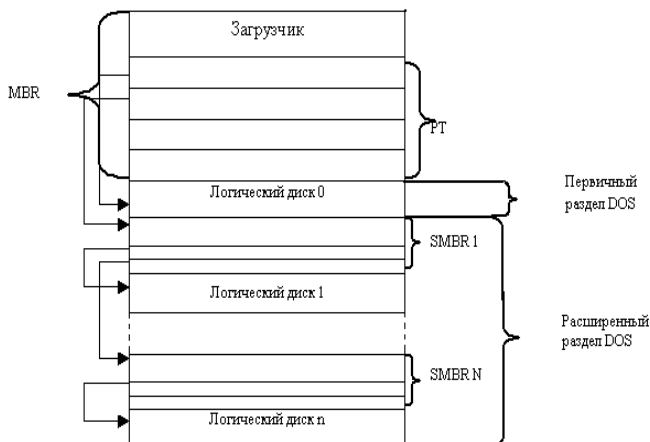


Рис. 8. Разбиение диска на разделы

Таким образом, описатель первичного раздела указывает сразу на загрузочный сектор логического диска (в первичном разделе он только один). Описатель расширенного раздела указывает на вторичную MBR (Secondary MBR – SMBR)

Вопросы и задания

1. Какое максимальное число (без знака) можно записать в 5 бит?
2. Сколько разных чисел можно записать в 10 битах?
3. Сколько бит требуется выделить для того, чтобы записать в них 7 десятичных чисел без знака. Известно, что значение чисел будет лежать в интервале от 0 до 15.

4. Какое число бит потребуется для записи числа 2586 в двоичном и упакованном BCD)?
5. Сколько различных символов можно представить с помощью Unicode?
6. Каковы недостатки BCD?
7. Что такое файл?
8. Что называется файловой системой?
9. К различиям каких характеристик сводятся основные различия между файловыми системами?
10. В чем разница между файловой системой и системой управления файлами?
11. Определите связанные распределения файлов.
12. Определите несвязные распределения файлов.
13. Перечислите разновидности блочной организации файлов.
14. Какова структура главной загрузочной записи (Master Boot Record, MBR)?
15. В чем разница понятий логический диск (логический раздел) и логическая структура диска?

Задача

Дана часть области данных 41h–48h кластеры, в которой записан файл при этом, кластер 41h свободен, в кластере 42h файл начинается, вторая часть файла записана в 43h-м, третья – в 44 h-м, 45h и 43h кластеры свободны 47h кластер испорчен, в 48h файл заканчивается. Требуется определить номера элементов FAT32 и их содержимое, отражающее данное заполнение области данных.

Исходные данные:

41	42	43	44
Испорчен	записана 1-я часть	Пустой	записана 2-я часть
45	46	47	48
Пустой	записана 3-я часть	испорчен	конец файла

4. ОСНОВНЫЕ ПОНЯТИЯ, СВЯЗАННЫЕ С ИНФОРМАЦИЕЙ. ИНФОРМАЦИОННЫЕ ХАРАКТЕРИСТИКИ, ОПРЕДЕЛЯЮЩИЕ КАЧЕСТВО ИНФОРМАЦИОННЫХ СИСТЕМ

Важным вопросом теории информации является установление меры количества и качества информации [1, 7, 8].

Информационные меры измерения отвечают трем основным направлениям в теории информации:

1) Структурная теория рассматривает дискретное строение массивов и их измерение простым подсчетом информационных элементов (квантов) или комбинаторным методом, предполагающим простейшее кодирование массивов информации.

2) Статистическая теория оперирует понятием энтропии как меры неопределенности, учитывающей вероятность появления, а, следовательно, и информативность тех или иных сообщений.

3) Семантическая теория учитывает целесообразность, полезность или существенность информации.

4.1. Структурные меры информации

В структурной теории различают геометрическую меру информации, комбинаторную меру и аддитивную меру (мера Хартли).

4.1.1. Геометрическая мера информации

Определение количества информации геометрическим методом сводится к измерению длины линии, площади или объема геометрической модели данного информационного комплекса в виде дискретных единиц-квантов. Под информационными элементами понимаются неделимые части – кванты информации в дискретных моделях реальных информационных комплексов, а также элементы алфавита в числовых системах.

Геометрическим методом определяется потенциальное, т. е. максимально возможное количество информации в виде структурных габаритов. Это полностью будет пониматься информационной емкостью исследуемой части информационной

системы. Информационная емкость вычисляется по сумме дискретных значений по всем измерениям [7].

Пусть информация отражается полным комплексом XTN. Если дискретные отсчеты осуществляются по осям X, T и N, соответственно через интервалы Δx , Δt , Δn , то непрерывные координаты распадаются на элементы (кванты) количество которых составляет

$$m_x = \frac{x}{\Delta x}, \quad m_T = \frac{T}{\Delta T}, \quad m_N = \frac{N}{\Delta N}.$$

Тогда количество информации в полном комплексе XTN, определенное геометрическим методом, равно в квантах

$$M = m_x \cdot m_T \cdot m_N.$$

4.1.2. Комбинаторная мера информации

К комбинаторной мере целесообразно прибегать тогда, когда требуется оценить возможность передачи информации при помощи различных комбинаций. Это одна из форм кодирования информации.

Количество информации в комбинаторной мере вычисляется как количество комбинации элементов, т. е. оценке подвергается комбинация потенциальных структурных разнообразных информационных комплексов.

В комбинаторике рассматриваются разные виды сочетаний элементов.

Сочетания из h элементов по ℓ различаются составом элементов, их возможное число равно

$$Q = (h^\ell) = \frac{h!}{\ell!(h-\ell)!} = \frac{h(h-1)\dots(h-\ell+1)}{1 \cdot 2 \dots \ell}.$$

Сочетания из h элементов по ℓ с повторениями также различаются составом элементов, но элементы в них могут повторять до ℓ раз.

$$Q_{\text{повт}} = (h^\ell)_{\text{повт}} = \frac{(h+\ell-1)!}{\ell!(h-1)!}.$$

Перестановки h элементов различаются их порядком. Число возможных перестановок h элементов равно:

$$Q = h!$$

Перестановка с повторением элементов, причем один из элементов повторяется α раз, другой β раз, наконец, последний γ раз, характеризуется числом

$$Q = \frac{(\alpha + \beta + \dots + \gamma)!}{\alpha! \beta! \dots \gamma!}$$

Размещения из h элементов по ℓ различаются и составом элементов, и их порядком. Возможное число размещений из h элементов по ℓ :

$$Q = (h^\ell) = \frac{h!}{(h - \ell)!} = h(h - 1)(h - 2) \dots (h - (\ell + 1)) \dots$$

Возможное число размещений с повторениями по ℓ из h элементов:

$$Q = (h^\ell)_{\text{повт}} = h^\ell.$$

При применении комбинаторной меры возможное количество информации Q совпадает с соединениями, т. е. определение количества информации Q в комбинаторной мере заключается не в простом подсчете квантов (геометрическая мера), а в определении количества возможных или действительно существующих комбинаций, т. е. в оценке структуризации.

Пример: Вычислить количество информации для десяти элементов при их различных соединениях.

1) В случае сочетаний из 10 элементов по 0, 1, 2, 3, 4, 5, 6, 7, 8 и 9 элементам получается следующее число образований, являющихся мерой количества информации:

$$Q = \frac{10!}{0!(10 - 0)!} + \frac{10!}{1!(10 - 1)!} + \dots + \frac{10!}{9!(10 - 9)!} + \frac{10!}{10!(10 - 10)!} = 1024$$

Перестановки этих 10 элементов дают

$$Q = h! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 \cdot 8 \cdot 9 \cdot 10 = 3628800 \text{ образований}$$

2) Размещения 10 разных элементов по 10 разным позициям приведет к еще большему появлению количества информации.

$$Q = h^1 = 10^{10} \text{ образований}$$

4.1.3. Аддитивная мера (мера Хартли)

В теории информации важную роль играет комбинаторика чисел и кодов. Введем понятия глубины h и длины ℓ числа.

Глубиной h числа называется количество различных элементов, содержащихся в принятом алфавите. Глубина числа соответствует основанию системы счисления и кодирования.

Длиной ℓ числа называется количество числовых гнезд, т. е. количество повторений в алфавите, необходимых и достаточных для представления чисел нужной величины. Один набор из ℓ гнезд-алфавитов составляет одну числовую группу, способную предоставлять и хранить одно полное число длиной ℓ . Некоторое количество чисел N представляет числовым полем. При глубине h и длине ℓ количество чисел, которые можно представить с помощью числовых групп, выразится формулой

$$Q = h^{\ell}.$$

Вследствие показательного закона Q зависит от ℓ . Число Q не является удобной мерой для оценки информационной емкости, поэтому Хартли ввел аддитивную двоичную логарифмическую меру, позволяющую вычислять количество информации в двоичных единицах – битах (бит). Для этого берется не само число Q , а его двоичный логарифм:

$$J = \log_2 Q = \log_2 h^1 = \ell \log_2 h \text{ (бит)}.$$

Здесь J – количество информации по Хартли.

Если количество разрядов (длина ℓ числа) равна единице, имеется в двоичной системе счисления (глубина $h = 2$) и исполняется двоичный логарифм, то потенциальное количество информации равно 1 биту ($\log_2 2 = 1$).

Это и есть единица измерения в принятой системе. Она соответствует одному элементарному событию, которая может произойти или не произойти. Аддитивная мера удобна тем, что она обеспечивает возможность считать пропорциональность J относительно числа ℓ .

4.2. Семантические меры информации

Под семантикой понимается смысл, содержание информации. Семантика – это одна из частей более общей науки – семиотики, в семиотике различают синтаксический, прагматический аспекты информации (рис. 9)



Рис. 9. Структура семантической меры измерения информации

В семантике имеются следующие основные понятия: знак, слово, язык [7].

Знаком называется условное изображение элемента сообщения.

Словом называется совокупность знаков, имеющих смысловое (предметное) значение.

Языком называется словарь и правила пользования им.

Структурная и статистическая мера информации относятся к синтаксическому аспекту.

Синтаксический аспект отображается теорией сигналов и кодов, рассматривающей условные обозначения элементов информации.

Синтаксические оценки не имеют прямого отношения к мерам информации, и поэтому нужно опустить их рассмотрение.

Далее рассмотрим оценки, отвечающие семантическому и прагматическому аспектам теории информации. В инженерных применениях прагматические оценки сливаются с семантическими, поскольку, не имеющие смысла сведения бесполезны, а бесполезные сведения бессмысленны.

Однако эффективность логического вывода, степень приближения к истине требует некоторой формализации смысла.

Один из путей формализации смысла предлагается семантической теорией информации. Карнап и Бар–Хиллел предложили использовать для целей измерения смысла функции истинности и ложности логических высказываний (предложений). За основу дискретного описания объекта берется неделимое предложение. Полученная таким образом оценка получило название *содержательности информации*.

4.2.1. Содержательность информации

Мера содержательности обозначается cont (от английского слова *content* – *содержание*).

Содержательность события i выражается через функцию меры $m(i)$ – содержательности его отрицания и определяется

$$\text{как } \text{cont}(i) = m(\sim i) = 1 - m(i),$$

где i – рассматриваемое событие; m – функция меры; $\sim$ – знак отрицания.

Логическая оценка количества информации $\inf$ равна

$$\text{Inf} = \log_2 \left(\frac{1}{1 - \text{cont}(i)} \right) = \log_2 \frac{1}{m(i)} = -\log_2 m(\sim i).$$

4.2.2. Целесообразность информации

Если информация используется в системах управления, то и полезность разумно оценивать по тому эффекту, который она оказывает на результат управления. В связи с этим А. А. Харкевичем была предложена мера целесообразности информации, которая определяется как измерение вероятности достижения цели при получении дополнительной информации [9].

Мера целесообразности в общем виде может быть аналитически выражена в виде соотношения

$$I_{\text{цел}} = \log_2 p_1 - \log_2 p_0 = \log_2 \frac{p_1}{p_0},$$

где p_0 и p_1 – начальная (до получения информации) и конечная (после получения информации) вероятность достижения цели.

Могут быть следующие случаи:

1) полученная информация может быть пустой, т. е. не иметь вероятности достижения цели, и в этом случае ее мера равна нулю;

2) полученная информация может изменять положение дел в нужную сторону, т. е. уменьшать вероятность достижения цели, и тогда она будет дезинформацией (ложной), которая уменьшается отрицательным значением количества информации;

3) полученная информация увеличивает вероятность достижения цели и измеряется положительной величиной количества информации.

Пример: На рис. 10 показано исходное состояние некоторой системы, соответствующей вершине 1, из которого возможно два пути: 1–2 и 1–3. Пусть точка 3 представляет собой цель, а точка 2 некоторый промежуточный путь в стороне от цели.

Если неизвестны пути к цели, то вероятности достижения цели по путям 1–2 и 1–3 будут одинаковы, а так как схема опыта содержит два исхода, то можно записать условие

$$p = p(1 - 2) = p(1 - 3) = \frac{1}{2}.$$

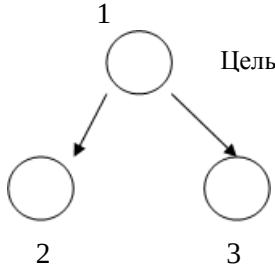


Рис. 10. Исходная система

1. Предположим, что достигнута точка 2 и в этой точке получается информация следующего характера (рис. 11).

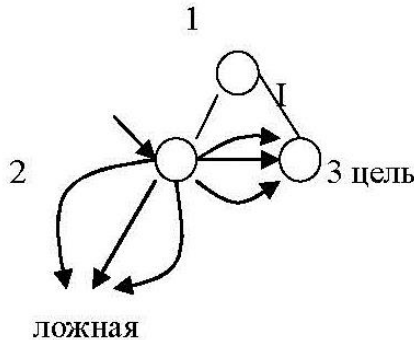


Рис. 11. Промежуточное состояние системы

Состояние системы описывается таким образом:

$$I = \log_2 \frac{p(2-3)}{p(1-2)} = \log_2 \frac{3/6}{3/6} = 0.$$

Эта информация оказалась пустой.

2. В точке 2 получена ложная информация, которая уменьшила вероятность достижения цели в виду преобладания ложных направлений (рис. 13). В этом случае целесообразность равна

$$I = \log_2 \frac{p(2-3)}{p(1-2)} = \log_2 \frac{1/6}{1/2} = -1,58.$$

3. Благоприятный случай. В вершине 2 поступает информация, которая два исхода из места оставляет ложными, но четыре исхода из шести являются благоприятными с точки зрения достижения цели 2 (рис. 12).

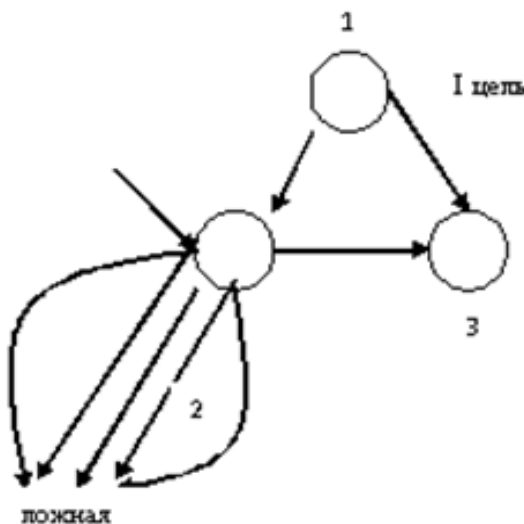


Рис. 12. Состояние системы при получении ложной информации

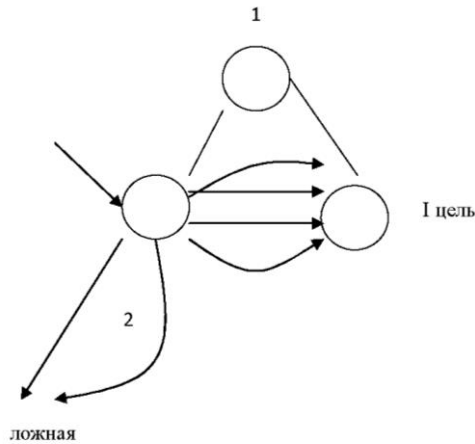


Рис. 13. Состояние системы при благоприятном случае

В благоприятном случае целесообразность равна

$$I = \log_2 \frac{p(2-3)}{p(1-2)} = \log_2 \frac{4/6}{1/2} = 0,42.$$

4.2.3. Динамическая энтропия

В том случае, когда целью извлечения информации являются обращенные в 0 неопределенные ситуации, удобно пользоваться понятием динамической энтропии (в ходе распознавания образов, диагноза болезней или расследования преступлений).

Энтропия (неопределенность) ситуации изменяется во времени: $H = H(t)$.

Большинство ситуаций можно представить как множество отношений между следствиями $\{a_i\}$ и причинами $\{b_j\}$. Следствия наблюдаются, а причины раскрываются по ним. Отношения между следствиями и причинами оцениваются вероятностями $p_{ij}(t)$. При этом $p_{ij}(t) = 0$, если нет связи между следствиями и причинами, $0 \leq p_{ij}(t) \leq 1$, если между ними имеется небольшая связь.

Общее количество следствий в момент t обозначить $N(t)$, а общее количество причин – $M(t)$. Тогда отдельные следствия a_i получают нумерацию $i=1,2,\dots,N(t)$, а причины – $j=1,2,\dots,M(t)$.

Энтропия для данного множества отношений:

$$H(t) = - \sum_{i=1}^{N(t)} \sum_{j=1}^{M(t)} p_{ij}(t) \log_2 p_{ij}(t).$$

По истечении единичного интервала времени поступает дополнительная информация, могущая изменить количество следствий $\{a_i\}$ от $N(t)$ до $N(t+1)$, количество причин $\{b_j\}$ от $N(t)$ до $N(t+1)$ и вероятности отношений между ними от $p_{ij}(t)$ до $p_{ij}(t+1)$.

В результате энтропия в момент $t+1$ равна

$$H(t+1) = - \sum_{i=1}^{N(t+1)} \sum_{j=1}^{M(t+1)} p_{ij}(t+1) \log p_{ij}(t+1).$$

В качестве меры информации целесообразно принять разность:

$$I = \Delta H = H(t) - H(t+1),$$

которая может быть как положительной, так и отрицательной величиной, в зависимости от того уменьшается или увеличивается неопределенность ситуации.

4.3. Статистические меры информации

4.3.1. Вероятность и информация

При вероятностном подходе информация рассматривается как сообщение об исходе случайных событий, реализации случайных величин и функций, а количество информации ставится в зависимость от априорных вероятностей этих событий, величин, функций.

Предположим, что источник сообщений может в каждый момент времени случайным образом принять одно из конечного множества возможных состояний. Каждому состоянию источника X ставится в соответствие условное обозначение в виде знака. Совокупность знаков $x_1, x_2, \dots, x_j, \dots, x_N$, соответствующих всем N возможным состояниям источника, называют его *алфавитом*, а количество состояний N – *объемом алфавита*. Формирование таким источником сообщений сводится к выбору им некоторого состояния x_j и выдачи соответствующего знака. Таким образом, под элементарным дискретным сообщением будем понимать символ x_j , выдаваемый дискретным источником. Отдельные состояния источника могут выбираться им чаще, другие реже. Поэтому в общем случае источник X определен совокупностью состояний с вероятностями их появления, составляющими в сумме 1.

$$X = \begin{pmatrix} x_1 & x_2 \dots & x_i \dots & x_n \\ p(x_1) & p(x_2) \dots & p(x_i) \dots & p(x_n) \end{pmatrix} \quad (4.1)$$

$$\sum_{i=1}^N P(x_i) = 1,$$

собственная информация символа x_i (количество информации, доставляемое самим символом x_i) определяется как $-\log(P(x_i))$, т. е. информация о каком-либо событии измеряется логарифмом величины, обратной вероятности его появления. Выбор основания логарифма $\log(P(x_i))$ определяет единицу количества информации. Если $a = 2$, то единица информации называется *двоичной* (бит), при $a = e$ – натуральной (нат), а при $a = 10$ – десятичной (дит). Двоичная единица количества информации, например, есть собственная информация символа, обладающего двумя равновероятными состояниями. Переход от одной системы логарифмов к другой равносильен простому изменению единицы измерения информации. Этот переход осуществляется по формуле

$$\log_b k = \log_a a \cdot \log_a k$$

Отсюда

$$1 \text{ нат} = \log_2 e \text{ бит} = 1,443 \text{ бит} \cdot$$

$$1 \text{ дит} = \log_2 10 \text{ бит} = 3,32 \text{ бит} \cdot$$

Таким образом, количество информации, содержащееся в дискретном сообщении, зависит от степени неожиданности этого сообщения, характеризуемой вероятностью его появления. Количество информации в сообщении тем больше, чем оно более неожиданно.

Источник удобнее характеризовать количеством собственной информации, содержащимся в среднем в одном символе ансамбля X , т. е. с помощью математического ожидания количества информации в отдельных событиях. Полагая $P_i = P(x_i)$, имеем

$$H(N) = - \sum_{i=1}^N p_i \log p_i \cdot \quad (4.2)$$

Это среднее количество собственной информации и названо *энтропией*. Мера (4.2) была предложена Клодом Шенноном в его фундаментальной работе «Математические основы теории связи», опубликованной в 1948 г., в которой были заложены основы современной теории информации.

Энтропию можно рассматривать и как меру количества информации, получаемой при полном устранении неопределенности относительно состояния источника.

В случае равновероятных состояний источника формула Шеннона имеет вид

$$H(N) = \log N \cdot \quad (4.3)$$

т.е., совпадает с формулой Хартли.

Понятие энтропия (от греческого *энтропе* – обращение) распространено в ряде областей знаний.

Так, Больцманом была определена энтропия физической системы следующим образом:

$$H = - \frac{1}{M_n} \sum_{i=1}^N m_i \ln \frac{m_i}{M_n} \quad (4.4)$$

M_n – число молекул в данном пространстве;

m_i – число молекул, обладающих скоростью $Y_i + \Delta Y$.

Заметим, что, так как $\frac{m_i}{M_n}$ есть вероятность того, что молекула имеет скорость $Y_i + \Delta Y$, то формулу (4.4) можно записать как

$$H = - \sum_{i=1}^N p_i \ln p_i .$$

Энтропия характеризует среднюю неопределенность выбора одного состояния из ансамбля. При ее определении используют только вероятности состояний, полностью игнорируя их содержательную сторону. Поэтому энтропия не может служить средством решения любых задач, связанных с неопределенностью. Например, при использовании этой меры для оценки неопределенности действия лекарства, приводящего к полному выздоровлению больных в 90 % случаев и улучшению самочувствия в остальных 10 % случаев, она получится такой же, как и у лекарства, вызывающего в 90 % случаев смерть, а в 10 % – ухудшение состояния больных.

Пример: Заданы ансамбли X и Y двух дискретных случайных величин

$$X = \begin{pmatrix} 0,5 & 0,7 & 0,9 & 0,3 \\ 0,25 & 0,25 & 0,25 & 0,25 \end{pmatrix}$$

$$Y = \begin{pmatrix} 5 & 10 & 15 & 8 \\ 0,25 & 0,25 & 0,25 & 0,25 \end{pmatrix}$$

Сравнить энтропии ансамблей.

Решение: $H(X) = H(Y) = \log_2 4 = 2$.

Энтропия как мера неопределенности согласуется с экспериментальными данными, полученными при изучении психологических реакций человека, в частности реакции выбора. Установлено, что время безошибочной реакции на последователь-

ность беспорядочно чередующихся равновероятных раздражителей (например, загорающихся лампочек) растет с увеличением их числа так же, как и энтропия. Это время характеризует неопределенность выбора одного раздражителя.

Замена равновероятных раздражителей неравновероятными приводит к снижению среднего времени реакции ровно настолько, насколько уменьшается энтропия.

4.3.2. Свойства энтропии ансамбля

1. $\overline{H(p)}$ непрерывна на интервале $0 \leq p \leq 1$, т. е., при малых измерениях p величина H изменяется мало.

2. $\overline{H(p)}$ симметрична относительно p , т. е., не изменяется при любой перемене аргументов p_i .

3. Если событие X_n состоит из двух событий X'_n и X''_n с вероятностями Q_1 и Q_2 , $Q_1 + Q_2 = P_N$, то

$$H(p_1, p_2, \dots, p_{N-1}, q_1, q_2) = H(p_1, p_2, \dots, p_N) + p_N \cdot H(q_1/p_N, q_2/p_N)$$

4. Энтропия всегда неотрицательна.

5. Энтропия равна нулю, когда вероятность одного из событий равна единице.

6. Энтропия имеет наибольшее значение, когда все вероятности равны между собой

$$p_1 = p_2 = \dots = p_N, \text{ при этом } H = -\log \frac{1}{N} = \log N.$$

Пример: Определить максимум энтропии системы, состоящей из 6 элементов, каждый из которых может быть в одном из четырех состояний.

Решение:

$$H_{\max} = \log_2 N = \log_2 m^n = \log_2 4^6 = 6 \cdot 2 = 12 \text{ бит.}$$

Количество информации

Количество информации есть уменьшение энтропии вследствие опыта.

Если неопределенность снимается полностью, то информация равна энтропии:

$$I = H,$$

в противном случае – разнице между начальной и конечной энтропией:

$$I = H_1 - H_2. \quad (4.5)$$

Наибольшее количество информации получается, когда полностью снимается максимальная неопределенность – когда вероятности всех событий были одинаковы

$$I' = -\log_2 p,$$

где p – вероятность реализации в условиях равной вероятности всех событий.

Если энтропия источника сообщений не равна максимальной энтропии для алфавита с данным количеством качественных признаков (имеются в виду качественные признаки алфавита, при помощи которых составляются сообщения), то это прежде всего означает, что сообщения данного источника могли бы нести большее количество информации.

Пример: Выстрел из орудия не поражает цель с вероятностью p . Через какое число выстрелов следует поинтересоваться, уничтожена ли цель, чтобы в результате ответа получить максимальное количество информации.

Решение: Ансамбль интересующих нас событий A . A_1 – цель поражена, A_2 – цель не поражена. Вероятность того, что цель не поражена после k -выстрелов, равна p^k . Вероятность противоположного события $1 - p^k$. Так как после ответа неопределенность устраняется полностью, количество информации равно энтропии, а она максимальна при равновероятности событий, т. е. при $p^k = 1 - p^k$, откуда $k = -1/\log_2 p$.

Пусть задан двоичный источник без памяти с алфавитом $X = \{0, 1\}$ и с вероятностями для символов «0» и «1» – p_0 и $p_1 = 1 - p_0$ соответственно. Выбор символов проводится незави-

симо. Его энтропия, называемая также функцией Шеннона, зависит только от вероятности p .

Энтропия двоичного источника (функция Шеннона):

$$\frac{H_b(p)}{\text{бит}} = -p \log_2 p - (1-p) \log_2 (1-p) \cdot$$

На рис. 14 показано поведение функции Шеннона. Энтропия двоичного источника всюду положительна и симметрична относительно $p = 1/2$ и имеет максимум при одинаковой вероятности символов «0» и «1». Максимальная энтропия, равная 1 бит, соответствует двоичному решению, т. е. на вопрос о значении символа это ответ: либо «да» либо «нет» (в приложении табл. 12 приведены значения $p \cdot \log_2 p$, $(1-p) \log_2 (1-p)$ и H двоичного источника).

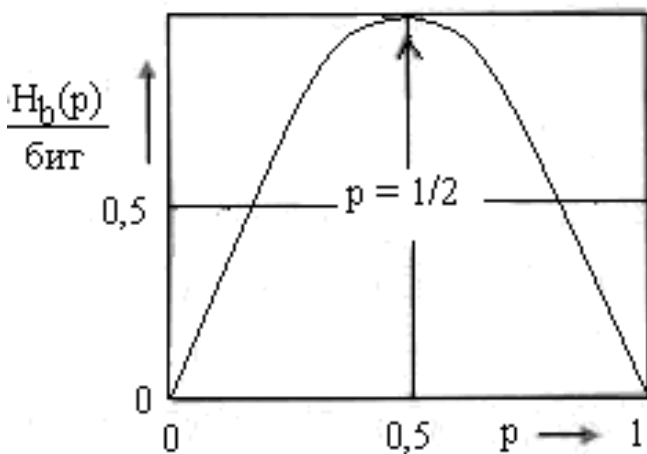


Рис. 14. Энтропия двоичного источника

Приведем пример использования понятия энтропии для вычисления необходимого для снятия неопределенности числа опытов. Пусть имеется 27 внешне неотличимых монет с одной фальшивой среди них. Определить минимальное число взвешиваний, которое необходимо провести на

равноплечих весах, чтобы среди 27 внешне неотличимых монет найти одну фальшивую, более легкую.

Решение: Общая неопределенность ансамбля V в соответствии с (4.3) составляет $H(V) = \log_2 27$ бит. Одно взвешивание способно прояснить неопределенность ансамбля V' , состоящего из 3-х элементов. Предположим, что монета № 1 находится на левой чаше весов, монета № 2 находится на правой чаше весов, а третья монета отложена. Тогда в результате опыта есть три возможных исхода (левая чаша весов легче; правая чаша весов легче; весы находятся в равновесии; соответственно: в первом случае фальшивая монета № 1; во втором – фальшивая монета № 2; в третьем – фальшивая монета № 3). Неопределенность, снимаемая одним опытом,

$$H(V') = \log_2 3 \text{ бит.}$$

Так как $H(V) = 3\log 3 = 3H(V')$, то для определения фальшивой монеты достаточно произвести три взвешивания.

Энтропия связанных источников. До сих пор в своих рассуждениях мы исходили из предположения независимости последовательных событий. Однако стоит лишь только открыть немецкий орфографический словарь, мы сразу же обнаружим зависимость между рядом стоящими буквами, например: << qu >>, << ch >>, << ck >>, << tz >> и << sch >>. Читая немецкий текст, мы видим, что после «q» за редким исключением следует «u». В этом случае «u», как почти неизбежное событие, практически не несет в себе никакой информации. Поэтому при определении информации подобного рода источников мы должны принимать во внимание взаимную связь между событиями.

4.3.3. Условная энтропия и энтропия объединения

Под объединением двух источников X и Y понимают обобщенный источник информации (X, Y) , характеризующийся вероятностями $p(x_i, y_j)$ всех возможных комбинаций состояний x_i источника X и y_j источника Y . Аналогично трактуется и объединение ансамблей, т. е.

$$H(X, Y) = - \sum_{i=1}^N \sum_{j=1}^k p(x_i, y_j) \log p(x_i, y_j), \quad (4.6)$$

здесь $p(x_i, y_j)$ – вероятности совместной реализации состояний

$$x_i (1 \leq i \leq N) \text{ и } y_j (1 \leq j \leq k).$$

В случае статистической независимости источников информации X и Y имеем

$$p(x_i, y_j) = p(x_i) p(y_j),$$

тогда

$$\begin{aligned} H(X, Y) &= - \sum_{i=1}^N \sum_{j=1}^k p(x_i) p(y_j) \log p(x_i) p(y_j) = \\ &= - \sum_{i=1}^N p(x_i) \log p(x_i) \sum_{j=1}^k p(y_j) - \sum_{j=1}^k p(y_j) \log p(y_j) \sum_{i=1}^N p(x_i) \end{aligned}$$

Учитывая, что

$$\sum_{i=1}^N p(x_i) = 1 \text{ и } \sum_{j=1}^k p(y_j) = 1,$$

получим

$$H(XY) = H(X) + H(Y) = H(X, Y). \quad (4.7)$$

В общем случае, если мы передаем m -сигналов A и ожидаем получить m -сигналов B , влияние помех в канале связи полностью описывается канальной матрицей. Со стороны источника сообщений канальная матрица, описывающая канал связи, имеет вид

$$P(Y/X) = \begin{vmatrix} p(y_1 / x_1)...p(y_j / x_1)...p(y_k / x_1) \\ \\ p(y_1 / x_2)...p(y_j / x_2)...p(y_k / x_2) \\ \\ p(y_1 / x_N)...p(y_j / x_N)...p(y_k / x_N) \end{vmatrix}$$

Вероятности, расположенные по диагонали, определяют правильный прием, остальные – ложный.

Значение цифр, заполняющих колонки канальной матрицы, обычно уменьшаются по мере удаления от главной диагонали, и при полном отсутствии помех, все, кроме цифр, расположенных по главной диагонали, равны нулю.

Если описывается канал связи со стороны источника сообщений, то прохождение данного вида сигнала в данном канале связи описывается распределением условных вероятностей вида $p(y_i/x_i)$, так для сигнала x_1 распределением вида

$$p(y_1/x_1), p(y_2/x_1), \dots, p(y_j/x_1), \dots, p(y_k/x_1). \quad (4.8)$$

Сумма вероятностей распределения (4.8) всегда должна равняться 1.

Учитывая, что $P(y_j, x_i) = P(x_i) \cdot P(y_j / x_i)$ для определения энтропии статистически связанных ансамблей имеем

$$\begin{aligned} H(X, Y) &= -\sum_{i=1}^N \sum_{j=1}^k p(x_i) p(y_j/x_i) \log p(x_i) p(y_j/x_i) = \\ &= -\sum_{i=1}^N p(x_i) \log p(x_i) \sum_{j=1}^k p(y_j/x_i) - \sum_{i=1}^N p(x_i) \sum_{j=1}^k p(y_j/x_i) \log p(y_j/x_i) \end{aligned} \quad (4.9)$$

Выражение

$$- \sum_{j=1}^k p(y_j / x_i) \log p(y_j / x_i) = H_{X_i}(Y)$$

есть случайная величина, характеризующая неопределенность, приходящуюся на одно состояние ансамбля Y , при условии, что реализовалось конкретное состояние x_i ансамбля X , которая называется *частной условной энтропией ансамбля Y* .

При усреднении по всем состояниям ансамбля X получаем среднюю неопределенность, приходящуюся на одно состояние ансамбля Y , при известных состояниях ансамбля X – общую условную энтропию ансамбля Y по отношению к ансамблю X .

$$H_X(Y) = H(Y / X) = \sum_{i=1}^N p(x_i) \sum_{j=1}^k p(y_j / x_i) \log p(y_j / x_i)$$

Таким образом, из (4.9) следует, что энтропия объединения двух статистически связанных ансамблей X и Y равна безусловной энтропии одного ансамбля плюс условная энтропия другого относительно первого.

$$H(XY) = H(X) + H(Y/X), \quad (4.10)$$

Распространяя правило (4.10) на объединение любого числа зависимых ансамблей, получим

$$H(XYZ...W) = H(X) + H_X(Y) + H_{XY}(Z) + \dots + H_{XYZ}(W). \quad (4.11)$$

Свойства:

$$H_X(Y) \leq H(Y), \quad (4.12)$$

$$H_Y(X) \leq H(X). \quad (4.13)$$

Действительно, наличие сведений о результатах реализации состояний одного ансамбля никак не может увеличить неопределенность выбора состояния из другого ансамбля. Эта неопределенность может только уменьшиться, если существует взаимосвязь в реализациях состояний из обоих ансамблей.

2. В случае отсутствия статистической связи в реализациях комбинаций состояний x_i источника X и y_j источника Y сведения о результатах выбора состояний из одного ансамбля не снижают неопределенности выбора состояний из другого ансамбля, что находит отражение в равенствах $H_X(Y) = H(Y)$, $H_Y(X) = H(X)$.

Если имеет место однозначная связь в реализациях состояний x_i ($1 \leq i \leq N$) из ансамбля X и y_j ($1 \leq j \leq k$) из ансамбля Y , то условная энтропия любого из ансамблей равна нулю:

Действительно, условные вероятности $p(x_i / y_j)$ и $p(y_j / x_i)$ в этом случае принимают значения, равные нулю или единице.

Пример: Связанные источники

Рассмотрим числовой пример, наглядно поясняющий приведенные выше определения и формулы. Для этой цели была подобрана задача, методика решения которой может непосредственно использоваться на практике.

Пусть мы имеем выборку 100 000 пар совместных событий (x_i, y_j) . Дискретных источников X и Y и алфавит каждого источника содержит четыре события. Пусть пара (x_i, y_j) встретилась 10 000 раз. Тогда оценка вероятности пары (x_i, y_j) равна $10\,000/100\,000 = 0,1$. Оценки остальных пар событий также получены подсчетами их относительной частоты и сведены в табл. 14.

Таблица 14

Оценка совместной вероятности пар символов $p(x_i, y_j)$ и вероятности отдельных символов $p(x_i)$ и $p(y_j)$

	y_1	y_2	y_3	y_4	$p(x_i)$
x_1	0,10	0,05	0,05	0	0,20
x_2	0,05	0,15	0,15	0	0,35
x_3	0	0,10	0,1	0,1	0,30
x_4	0	0,05	0,05	0,05	0,15
$p(y_j)$	0,15	0,35	0,35	0,15	1

Будем считать, что полученные оценки близки к вероятностям пар событий и в дальнейшем будем говорить уже о вероятностях. Вероятности событий x_i, y_j получены суммированием строк и столбцов. Контрольная сумма $\sum_{i=1}^4 x_i = \sum_{j=1}^4 y_j = 1$ приведена

в правом нижнем углу. Теперь, когда нам известны все вероятности, необходимые для подсчета энтропии, определим:

- 1) энтропии источников X и Y ;
- 2) совместную энтропию источников;
- 3) обе условные энтропии.

Для контроля мы также вычислим:

- 4) условные вероятности $P(y_j/x_i)$;
- 5) определим условную энтропию $H(Y/X)$.

Замечание: Для простоты проведем расчеты с точностью до 4 знаков после запятой.

Решение:

$$1) \quad H(Y) = - \sum_{j=1}^4 p(y_j) \log_2 p(y_j) =$$

$$= -2[0,15 \cdot \log_2(0,15) + 0,35 \cdot \log_2(0,35)] = 1,85 \text{ бит/симв}$$

$$H(X) = - \sum_{i=1}^4 p(x_i) \log_2 p(x_i) = -[0,2 \log_2 0,2 + 0,35 \log_2 0,35 +$$

$$+ 0,3 \log_2 0,3 + 0,15 \log_2 0,15] = 1,92 \text{ бит/симв}$$

$$2) \quad H(X, Y) = - \sum_{i=1}^4 \sum_{j=1}^4 p(x_i, y_j) \log_2 p(x_i, y_j) = -[0,1 \log_2 0,1 +$$

$$+ 2 \cdot 0,05 \cdot \log_2(0,05) + 0,05 \cdot \log_2(0,05) + 2 \cdot 0,15 \cdot \log_2(0,15) +$$

$$+ 3 \cdot 0,1 \cdot \log_2 0,1 + 3 \cdot 0,05 \cdot \log_2 0,05] = 4,69 \text{ бит/симв}$$

3) Получаем

$$H(X/Y) = H(X, Y) - H(Y) = 4,69 - 1,85 = 2,84 \text{ бит/симв}$$

$$H(Y/X) = H(X, Y) - H(X) = 4,69 - 1,92 = 2,77 \text{ бит/симв}$$

Таблица 15

Условная вероятность $P(y_j/x_i)$

	y_1	y_2	y_3	y_4	$\sum_{j=1}^4 p(y_j/x_i)$
x_1	1/2	1/4	1/4	0	1
x_2	1/7	3/7	3/7	0	1
x_3	0	1/3	1/3	1/3	1
x_4	0	1/3	1/3	1/3	1

В табл. 15 приведены условные вероятности, подсчитанные исходя из табл. 14. Заметим, что при этом мы получили так называемую стохастическую матрицу. Сумма условных вероятностей для каждой строки равна 1.

$$H(Y/X) = - \sum_{i=1}^4 \sum_{j=1}^4 p(x_i, y_j) \log_2 p(y_j/x_i) = 1,5496 \text{ бит}.$$

4.3.4. Дифференциальная энтропия

На практике множество возможных состояний источника информации составляет континуум, т. е. источники – непрерывные.

Определим энтропию непрерывного источника информации следующим образом : разобьем диапазон изменения непрерывной случайной величины U , характеризующейся плотностью распределения вероятностей $p(u)$, на конечное число n малых интервалов шириной Δu . При реализации любого значения u , принадлежащего интервалу $(u_i, u_i + \Delta u)$, будем считать, что реализовалось значение u_i дискретной случайной величины U . Поскольку Δu мало, вероятность $p(u_i < u < u_i + \Delta u)$ реализации значения u из интервала $u_i, u_i + \Delta u$:

$$p(u_i \leq u \leq u_i + \Delta u) = \int_{u_i}^{u_i + \Delta u} p(u) du \approx p(u_i) \Delta u.$$

Тогда энтропия дискретной случайной величины U может быть записана в виде

$$H(\tilde{U}) = - \sum_{i=1}^n p(u_i) \Delta u \log [p(u_i) \Delta u].$$

или

$$H(\tilde{U}) = - \sum_{i=1}^n p(u_i) \Delta u \log p(u_i) - \sum_{i=1}^n p(u_i) \Delta u \log \Delta u.$$

Так как

$$\sum_{i=1}^n p(u_i) \Delta u = 1$$

то

$$H(\tilde{U}) = - \sum_{i=1}^n p(u_i) \Delta u \log p(u_i) - \log \Delta u.$$

По мере уменьшения Δu $p(u_i < u \leq u_i + \Delta u)$ все больше приближается к вероятности $p(u_i)$, равной нулю, а свойства дискретной величины $\tilde{U}$ — к свойствам непрерывной случайной величины U . Переходя к пределу при $\Delta u \rightarrow 0$, получаем следующее выражение для энтропии $H(U)$ непрерывного источника:

$$\lim_{\Delta u \rightarrow 0} H(\tilde{U}) = \lim_{\Delta u \rightarrow 0} \left\{ - \sum_{i=1}^n p(u_i) \Delta u \log p(u_i) \right\} - \lim_{\Delta u \rightarrow 0} \log \Delta u$$

или

$$H(U) = - \int_{-\infty}^{\infty} p(u) \log p(u) du - \lim_{\Delta u \rightarrow 0} \log \Delta u. \quad (4.14)$$

Эта величина при $\Delta u \rightarrow 0$ стремится к бесконечности, что полностью соответствует интуитивному представлению о том, что неопределенность выбора из бесконечно большого числа возможных состояний (значений) бесконечно велика.

Первый член в правой части соотношения (4.14) имеет конечное значение, которое зависит только от закона распределения непрерывной случайной величины U и не зависит от шага квантования Δu . Он имеет точно такую же структуру, как и энтропия дискретного источника.

Второй член того же соотношения, наоборот, зависит лишь от шага квантования случайной величины U , из-за него $H(U)$ обращается в бесконечность.

Один из подходов для получения конечной характеристики информационных свойств непрерывного источника состоит в том, что в качестве неопределенности непрерывного источника принимают первый член соотношения (4.14):

$$h(U) = - \int_{-\infty}^{\infty} p(u) \log p(u) du . \quad (4.15)$$

Она получила название *относительной дифференциальной энтропии* или просто *дифференциальной энтропии* непрерывного источника информации (непрерывного распределения случайной величины U).

Ее можно трактовать как среднюю неопределенность выбора случайной величины U с произвольным законом распределения по сравнению со средней неопределенностью выбора случайной величины U' , изменяющейся в диапазоне, равном единице, и имеющей равномерное распределение.

Свойства дифференциальной энтропии

А. Если единственным ограничением для случайной величины U является область ее возможных значений $[\alpha, \beta]$, то максимальной дифференциальной энтропией обладает равномерное распределение вероятностей в этой области.

Б. Если ограничения на область значений непрерывной случайной величины U отсутствуют, но известно, что дисперсия ее ограничена, то максимальной дифференциальной энтропией обладает нормальное распределение величины U .

Таблица 16

Пример дифференциальной энтропии

Распределения	Функция плотности распределения вероятности	Дифференциальная энтропия
Равномерное	$f(x) = \begin{cases} \frac{1}{2\sqrt{3}} & \text{для } x \leq \sqrt{3} \\ 0 & \text{const} \end{cases}$	$\frac{\ln(2\sqrt{3}\sigma)}{\ln 2} = 1,79$
Лапласа	$f(x) = \frac{1}{\sqrt{2}\sigma^2} \exp\left(-\sqrt{2}\frac{ x }{\sigma}\right)$	$\frac{\ln(\sqrt{2}\sigma e)}{\ln 2} = 1,94$
Гауссовское	$f(x) = \frac{1}{\sqrt{2}\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right)$	$\frac{\ln(2\pi\sigma^2 e)}{2 \ln 2} = 2,04$

Пример: Телефония

Практическая польза приведенных выше результатов может быть наглядно показана при помощи оценки достижений скорости передачи информации (в битах) в цифровых телефонных линиях. Современные стандартные методы цифровой передачи речи (логарифмические PCM) требуют затраты 8 бит на кодирование одного отсчета, при частоте отсчетов 8 кГц. Таким образом, скорость передачи речи составляет 64 кбит/секунд.

Исходя из равномерного распределения вероятностей в интервале $[-1,1]$, опытным путем получим $\sigma^2 = 1/3$. Таким образом, дифференциальная энтропия на один отсчет составляет

$$\hat{H}(X) = \frac{\ln(2\sqrt{3}\sigma)}{\ln 2} \Big|_{\sigma^2=1/3} = 1 \text{ бит}$$

Так как отсчеты проводятся с частотой 8 кГц, получаем, что необходимая скорость передачи речи составляет 8 кбит/секунд. При оценке энтропии мы не принимали во внимание связи между соседними отсчетами (память источника), и поэтому реальная дифференциальная энтропия источника речи будет еще меньше. Известно, что современные алгоритмы кодирования речи позволяют осуществлять передачу речевого сиг-

нала со скоростью около 8 кбит/с при качестве, сравнимом со стандартным РСМ.

4.3.5. Эпсилон-энтропия

Пусть:

1. Отдельные состояния источника информации есть независимые реализации случайной величины U . Ансамбль реализаций случайной величины U описывается плотностью распределения вероятностей $p(u)$.

2. О значениях случайной величины U можно судить по значениям другой случайной величины Z , если мера их различия не превышает заданной точности воспроизведения.

3. Требование к точности задается с использованием критерия $\tilde{V}(ZU)$:

$$\tilde{V}(ZU) = \iint p(u)p_u(z)(z-u)^2 du dz \leq \varepsilon^2, \quad (4.16)$$

где $p_u(z)$ – условная плотность распределения – функция правдоподобия того, что конкретный сигнал u будет воспроизведен как сигнал z ; ε – заданное значение точности.

Так как плотность $p(u)$ определена, то для выполнения условия (4.16) варьировать можно только условной плотностью распределения $p_u(z)$.

Если случайная величина Z воспроизводит случайную величину U с некоторой точностью ε , то количество информации, содержащееся в воспроизводящей величине Z относительно U , конечно и определяется

$$I(ZU) = \iint p(u)p_u(z) \log \frac{p_u(z)}{p(z)}, \quad (4.17)$$

где $p(z) = \int p(u)p_u(z)du$ – плотность воспроизводящей величины Z .

Желательно обеспечить заданную верность воспроизведения при минимальном количестве получаемой информации. Поэтому среди множества функций $p_u(z)$, удовлетворяющих

условию (4.16), целесообразно выбрать такую, которая обеспечивает наименьшее $I\{ZU\}$

Минимальное количество информации в одной случайной величине Z относительно другой U , при котором удовлетворяется заданное требование к точности воспроизведения величины U , называется ε (эпсилон) – энтропией величины U и обозначается $H_\varepsilon(U)$:

$$H_\varepsilon(U) = \min_{(p_u(z))} I(ZU)$$

при

$$V(ZU) \leq \varepsilon^2. \quad (4.18)$$

Большая практическая важность этой величины обусловлена тем, что ε (эпсилон) – энтропия есть минимальное количество двоичных цифр, которое нужно передать, чтобы воспроизвести сообщение с заданной верностью (точностью).

4.4. Информационные потери при передаче по дискретным каналам связи с шумами

Потери информации в каналах связи с шумами обычно описывают при помощи условной энтропии и энтропии объединения. Если помех нет или их уровень настолько низок, что они не в состоянии уничтожить сигнал или имитировать полезный сигнал в отсутствие передачи, то при передаче x_i мы будем твердо уверены, что получим y_j -сигнал, соответствующий переданному x_i -му сигналу. События X и Y статистически жестко связаны, условная вероятность максимальна

$$p(y_j/x_i) = 1,$$

а частная условная энтропия

$$H_{x_i}(Y) = -\sum_{j=1}^m p(y_j/x_i) \log p(y_j/x_i) = 0,$$

так как $\log p(y_j/x_i) = \log 1 = 0$.

В этом случае количество информации, содержащееся в принятом ансамбле сообщений Y , равно энтропии передаваемых сообщений ансамбля X , т. е.

$$I(Y, X) = H(X).$$

При высоком уровне помех любой из принятых сигналов y_j может соответствовать любому переданному сигналу x_i , статистическая связь между переданными и принятыми сигналами отсутствует.

В этом случае вероятности $p(x_i)$ и $p(y_j)$ есть вероятности независимых событий и $p(y_j/x_i) = p(y_j)$, $p(x_i/y_j) = p(x_i)$.

$$H(X/Y) = - \sum_i \sum_j p(y_j) p(x_i/y_j) \log p(x_i/y_j) =$$

$$= - \sum_i \sum_j p(y_j) p(x_i) \log p(x_i) = \sum_j p(y_j) H(X) = H(X),$$

так как $\sum_j p(y_j) = 1$, т. е. условная энтропия равна безусловной, а количество информации, содержащееся в Y относительно X равно нулю

$$I(X, Y) = H(X) - H(X/Y) = 0.$$

Информационные характеристики реальных каналов связи лежат между этими двумя предельными случаями. При этом потери информации при передаче k -символов по данному каналу связи

$$\Delta I = kH(X/Y).$$

Так как имеет место

$$H(X, Y) = H(X) + H(Y/X) = H(Y) + H(X/Y),$$

то, потери в канале связи могут быть учтены при помощи энтропии объединения следующим образом:

$$I(Y, X) = H(X) + H(Y) - H(Y, X),$$

а с использованием условной энтропии

$$I(Y, X) = H(X) - H(X/Y) = H(Y) - H(Y/X).$$

Найденные зависимости можно пояснить при помощи диаграммы потоков информации.

$H(X/Y)$ определяет среднюю меру неопределенности посланного символа источника X в том случае, когда символы приемника Y известны, т. е. оставшуюся неопределенность. Величину $H(X/Y)$ часто называют также *утечкой информации*.

Так как энтропию $H(X)$ можно определить как собственную информацию источника X , $H(X) = I(X, X)$. В бесшумном канале $H(X/Y) = 0$, т. е. информация передается от источника к приемнику и обратно без потерь.

Если канал полностью зашумлен, то $H(X/Y) = H(X)$ и передачи информации не происходит.

$H(Y/X)$ определяет среднюю неопределенность принятого символа при известных посланных символах, поэтому ее называют «посторонней» шумовой информацией.

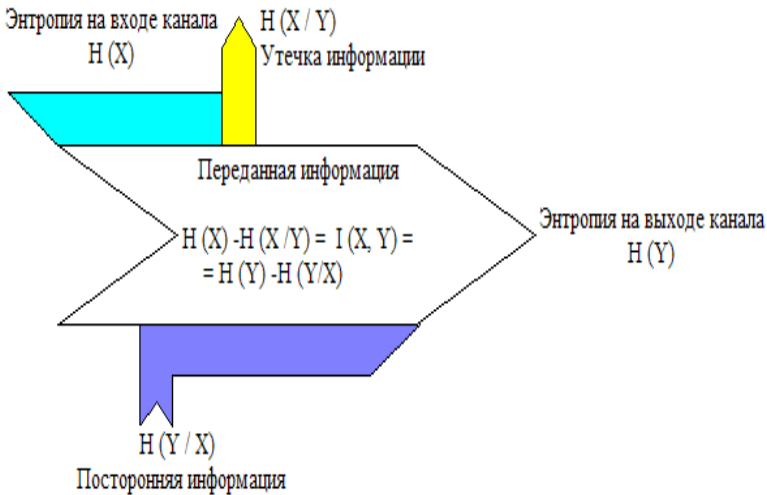


Рис. 15. Диаграмма потоков информации

Примеры:

Задача 7

В первом урне находится a черных и b белых шаров, во второй – b черных и a белых. Определить сколько информации сообщает нам цвет извлеченных из урны шаров о том, из какой урны был вытянут шар?

X_1 – первая урна;

X_2 – вторая урна;

Y_1 – белый шар;

Y_2 – черный шар.

$I(X, Y)$ – ?

Решение:

$$p(x_1) = p(x_2) = p(y_1) = p(y_2) = \frac{1}{2};$$

$$p(y_1 / x_1) = \frac{a}{a+b}; \quad p(y_2 / x_1) = \frac{b}{a+b};$$

$$p(y_2 / x_2) = \frac{a}{a+b};$$

$$p(y_1 / x_2) = \frac{b}{a+b};$$

$$I(X, Y) = \sum_i \sum_j p(x_i, y_j) \log_2 \frac{p(y_j / x_i)}{p(y_j)} = \frac{1}{2} * \frac{a}{a+b} \log_2 \frac{2a}{a+b} + \frac{1}{2} * \frac{b}{a+b} \log_2 \frac{2b}{a+b} +$$

$$\frac{1}{2} * \frac{b}{a+b} \log_2 \frac{2b}{a+b} + \frac{1}{2} * \frac{a}{a+b} \log_2 \frac{2a}{a+b} = \frac{a}{a+b} \log_2 \frac{2a}{a+b} + \frac{b}{a+b} \log_2 \frac{2b}{a+b};$$

Пусть $a = 1$, $b = 3$, тогда $I(X, Y) = 0,16347$.

Задача 8

Канал связи описан следующей канальной матрицей:

$$p(b/a) = \begin{vmatrix} 0,98 & 0,01 & 0,01 \\ 0,1 & 0,75 & 0,15 \\ 0,2 & 0,3 & 0,5 \end{vmatrix}.$$

Вероятности появления символов источника сообщений равны $p(a_1) = 0,7$; $p(a_2) = 0,2$; $p(a_3) = 0,1$. Чему равны информационные потери при передаче сообщения из 400 символов алфавита a_1, a_2, a_3 ? Чему равно количество принятой информации?

Решение:

Общая условная энтропия:

$$\begin{aligned} H(B/A) &= -\sum_i p(a_i) \sum_j p(b_j/a_i) \log_2 p(b_j/a_i) = \\ &= -[0,7(0,98 \log_2 0,98 + 0,01) + 0,2(0,75 \log_2 0,75 + \\ &+ 0,1 \log_2 0,1 + 0,15 \log_2 0,15) + 0,1(0,2 \log_2 0,2 + \\ &+ 0,3 \log_2 0,3 + 0,5 \log_2 0,5)] = 0,473 \text{ бит / символ} \end{aligned}$$

Потери в канале связи:

$$\Delta I = k * H(B/A) = 400 * 0,473 = 189,5 \text{ бит};$$

Энтропия приемника – m

$$H(B) = -\sum_{j=1}^m p(b_j) \log_2 p(b_j)$$

$$\begin{aligned} p(b_1) &= \sum_i p(a_i) p(b_1/a_i) = p(a_1) p(b_1/a_1) + p(a_2) p(b_1/a_2) + p(a_3) p(b_1/a_3) = \\ &= 0,7 * 0,98 + 0,2 * 0,1 + 0,1 * 0,2 = 0,726 \end{aligned}$$

$$\begin{aligned} p(b_2) &= p(a_1) p(b_2/a_1) + p(a_2) p(b_2/a_2) + p(a_3) p(b_2/a_3) = \\ &= 0,7 * 0,01 + 0,2 * 0,75 + 0,1 * 0,3 = 0,187; \end{aligned}$$

$$\begin{aligned} p(b_3) &= p(a_1) p(b_3/a_1) + p(a_2) p(b_3/a_2) + p(a_3) p(b_3/a_3) = \\ &= 0,7 * 0,01 + 0,2 * 0,15 + 0,1 * 0,5 = 0,087; \end{aligned}$$

$$p(b_1) + p(b_2) + p(b_3) = 0,726 + 0,187 + 0,087 = 1;$$

$$H(B) = -(0,726 \log_2 0,726 + 0,187 \log_2 0,187 + 0,087 \log_2 0,087) = 1,094 \text{ бит} / \text{символ}.$$

Среднее количество полученной информации:

$$I = k[I(B, A)] = k[H(B) - H(B/A)] = kH(B) - \Delta I = 248,1 \text{ бит}.$$

4.4.1. Скорость передачи информации по дискретному каналу связи

Характеризуя дискретный канал связи, используют два понятия скорости передачи: технической и информационной.

Под технической скоростью передачи V_τ , называемой также *скоростью манипуляции*, подразумевают число элементарных сигналов (символов) передаваемых по каналу в единицу времени. Она зависит от свойств линии связи и быстродействия аппаратуры канала. С учетом возможных различий в длительностях символов

$$V_\tau = \frac{1}{\tau_{\text{cp}}},$$

где τ_{cp} – среднее значение длительности символа. При одинаковой продолжительности τ всех передаваемых символов $\tau_{\text{cp}} = \tau$. Единицей измерения технической скорости служит бод – скорость, при которой за одну секунду передается один символ.

Информационная скорость или скорость передачи информации, определяется средним количеством информации, которое передается по каналу в единицу времени. Она зависит как от характеристики данного канала связи, таких как объем алфавита используемых символов, техническая скорость их передачи, статистические свойства помех в линии, так и от вероятностей, поступающих на вход символов и их статистической взаимосвязи.

При известной скорости манипуляции V_τ скорость передачи информации $V_u(X, Y)$ задается соотношением

$$V_u(X, Y) = V_t I(X, Y),$$

где $I(X, Y)$ – среднее количество информации, переносимое одним символом.

Пример: Сообщения передаются в двоичном коде ($m = 2$), время передачи $0 \tau_0 = 1$ с, длительность импульса, соответствующего $\tau_1 = 5$ секунд. Определить скорость передачи информации, если вероятность появления символа 0 - $p_0 = 0,37$, вероятность появления символа 1 - $p_1 = 0,63$.

Решение:

$$C = \frac{H_1}{\tau_{cp}} = \frac{-\sum_{i=1}^m p_i \cdot \log_2 p_i}{\sum_{i=1}^m \tau_i \cdot p_i} = \frac{-(0,37 \cdot \log_2 0,37 + 0,63 \cdot \log_2 0,63)}{0,63 \cdot 5 + 0,37 \cdot 1} = 0,27 \text{ бит/сек}$$

4.4.2. Пропускная способность дискретного канала без помех

Для теории и практики важно иметь представление, до какого предела и каким путем можно повысить скорость передачи информации по конкретному каналу связи. Предельные возможности канала по передаче информации, характеризуются его пропускной способностью.

Пропускная способность канала C_{Π} равна той максимальной скорости передачи информации по данному каналу, которой можно достичь при самых совершенных способах передачи и приема.

$$C_{\Pi} = \max V_u(X, Y) = \max V_t I(X, Y).$$

При заданном алфавите символов и фиксированных основных характеристиках канала (например, полосе частот, средней и пиковой мощности передатчика), остальные характеристики должны быть выбраны такими, чтобы обеспечить наибольшую скорость передачи по нему элементарных сигналов, т. е., обеспечить максимальное значение V_t . Максимум среднего количе-

ства информации, приходящейся на один символ принятого сигнала $I(X, Y)$, определяется на множестве распределений вероятностей между символами $X_1, \dots, X_1, \dots, X_m$.

Пропускная способность канала, как и скорость передачи информации по каналу, измеряется числом двоичных единиц информации в секунду (бит в секунду).

При отсутствии помех существует взаимно-однозначное соответствие между множеством символов $\{Y\}$ на выходе канала и $\{X\}$ на его входе и

$$I(X, Y) = I(Y, X) = H(X).$$

Максимум возможного количества информации на символ равен $\log m$, где m – объем алфавита символов, откуда пропускная способность дискретного канала без помех.

$$C_{\Pi} = V_{\tau} \log m.$$

Следовательно, для увеличения скорости передачи информации по дискретному каналу без помех и приближения ее к пропускной способности канала, последовательность букв сообщения должна подвергнуться такому преобразованию в коде, при котором различные символы в его выходной последовательности появлялись по возможности равновероятно, а статистические связи между ними отсутствовали бы. Доказано, что это выполнимо для любой эргодической последовательности букв, если кодирование осуществлять блоками такой длины, при которой справедлива теорема об их асимптотической равновероятности.

Расширение объема алфавита символов m приводит к повышению пропускной способности канала, однако возрастает и сложность технической реализации.

4.4.3. Пропускная способность дискретного канала с помехами

При наличии помех соответствие между множествами символов на входе и выходе канала связи перестает быть однозначным.

Среднее количество информации, передаваемое в этом случае, определяется соотношением

$$I(X, Y) = H(Y) - H(Y/X) = H(X) - H(X/Y).$$

Если объем алфавита входных символов X равен m_1 , а выходных Y – m_2 , то

$$I(X, Y) = \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} p(y_j, x_i) \log \frac{p(y_j, x_i)}{p(y_j) \cdot p(x_i)}. \quad (4.19)$$

Соответственно скорость передачи информации по каналу с помехами

$$V_u(X, Y) = V_\tau \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} p(y_j, x_i) \log \frac{p(y_j, x_i)}{p(y_j) \cdot p(x_i)}.$$

Считая скорость манипуляции V_τ предельно допустимой при заданных технических характеристиках канала, величину $V_u(X, Y)$ можно максимизировать, изменяя статистические свойства последовательностей символов на входе канала посредством преобразователя (кодера канала).

Получаемое при этом предельное значение C_Π скорости передачи информации по каналу называют *пропускной способностью дискретного канала связи с помехами*.

$$C_\Pi = \max_{p\{X\}} V_\tau I(X, Y),$$

где $p\{X\}$ – множество возможных распределений вероятностей входных сигналов. Важно подчеркнуть, что при наличии помех пропускная способность канала определяет наибольшее количество информации в единицу времени, которое может быть пере-

дано со сколь- угодно малой вероятностью ошибки, т. е. с математической точки зрения, поиск пропускной способности дискретного канала без памяти сводится к поиску распределения вероятностей входных символов источника, обеспечивающего максимум информации $I(x, y)$. При этом на вероятности входных символов $x \in X$ накладываются ограничения

$$0 < p(x) \leq 1 \quad \text{и} \quad \sum_x p(x) = 1. \quad (4.20)$$

В принципе определение максимума $I(x, y)$ при ограничениях (4.20) возможно при использовании мультипликативного метода Лагранжа. Однако такое решение требует чрезмерно больших затрат.

Для симметричных каналов имеет место следующая теорема:

В симметричных каналах без памяти пропускная способность достигается при равномерном распределении вероятностей входных символов источника X .

Свойства симметричного канала связи:

- 1) В симметричном канале связи $H(X) = H(Y)$.
- 2) Условная энтропия $H(X/Y) = H(Y/X)$.
- 3) Среднее количество информации в принятом ансамбле сообщений относительно переданного

$$\begin{aligned} I(X, Y) = I(Y, X) &= H(X) - H(Y/X) = H(Y) - H(Y/X) = \\ &= H(Y) - H(X/Y) = H(X) + H(Y) - H(X, Y). \end{aligned}$$

4) Канальная матрица для симметричного канала связи со стороны источника и со стороны приемника выглядит одинаково.

5) В симметричном канале связи сумма вероятностей в каждой строке и в каждом столбце равна единице. Пропускная способность симметричного канала от X к Y равна пропускной способности того же канала от Y к X .

В разделе 5 будет показано, что к пропускной способности канала связи с помехами можно приблизиться, кодируя эргодическую последовательность букв источника сообщений блоками

такой длины, при которой справедлива теорема об асимптотической равновероятности длинных последовательностей.

Произвольно малая вероятность ошибки оказывается достижимой только в пределе, когда длина блоков становится бесконечной.

При удлинении кодируемых блоков возрастает сложность технической реализации кодирующих и декодирующих устройств и задержка в передаче сообщений, обусловленная необходимостью накопления требуемого числа букв в блоке. В рамках допустимых усложнений на практике при кодировании могут преследоваться две цели: либо при заданной скорости передачи информации стремятся обеспечить минимальную ошибку, либо при заданной достоверности – скорость передачи, приближающуюся к пропускной способности канала.

Пределные возможности канала никогда не используются полностью. Степень его загрузки характеризуется *коэффициентом использования канала*

$$\lambda = I(Z)/C_{\Pi},$$

где $I(Z)$ – производительность источника сообщений;
 C_{Π} – пропускная способность канала связи.

Поскольку нормальное функционирование канала возможно, как показано далее, при изменении производительности источника в пределах $0 \leq I(Z) \leq C_{\Pi}$, λ теоретически может изменяться в пределах от 0 до 1.

Определим пропускную способность симметричного канала без памяти. Симметричный канал без памяти – это дискретный канал, в котором каждый передаваемый кодовый символ принимается ошибочно с вероятностью p и правильно с вероятностью $(1-p)$. Причем в случае ошибки вместо переданного i -го символа может быть с равной вероятностью принят любой другой из m -символов. Таким образом, условная вероятность того, что принят символ y_j , когда на самом деле был передан символ x_i :

$$p(y_j/x_i) = \begin{cases} \frac{p}{m-1}, & \text{при } i \neq j \\ 1-p, & \text{при } i = j \end{cases}, I'(X, Y) = \max_{P(X)} [H(Y) - H(Y/X)],$$

$$H(Y/X) = M \left[\log \frac{1}{p(y_j/x_i)} \right] = p \log \frac{m-1}{p} + (1-p) \log \frac{1}{1-p}.$$

Условная энтропия не зависит от распределения вероятностей в ансамбле X и определяется только переходной вероятностью канала

$$I'(X, Y) = \max_{P(X)} \left[H(Y) - p \log \frac{m-1}{p} - (1-p) \log \frac{1}{1-p} \right].$$

Поскольку в правой части равенства только $H(Y)$ зависит от распределения $P(X)$, то максимизировать надо $H(Y)$. Можно показать, что максимум $H(Y) = \log m$ и реализуется в том случае, когда все символы y_j равновероятны и независимы (это достигается тогда, когда входные символы равновероятны и независимы):

$$p(y_j) = \sum_{i=1}^m p(x_i) p(y_j/x_i) = \frac{1}{m} \sum_{i=1}^m p(y_j/x_i) = \frac{1}{m} \left[1-p + (m-1) \frac{p}{m-1} \right] = \frac{1}{m},$$

$$H(Y) = \log \left(\frac{1}{p(y_j)} \right) = \log m,$$

$$I'(X, Y) = \log m + p \log \frac{p}{m-1} + (1-p) \log(1-p),$$

$$C_{\Pi} = V \left[\log m - p \log \frac{m-1}{p} + (1-p) \log(1-p) \right].$$

Еще более упрощая задачу, переходят к двоичному каналу, т. е., $m = 2$ (рис. 16):

$$C_{\Pi} = V [I + p \log p + (1 - p) \log(1 - p)].$$

При $p = \frac{1}{2}$ пропускная способность равна 0: $C_{\Pi} = 0$.

Это ситуация так называемого обрыва канала, когда последовательности символов на входе и выходе канала независимы. В этом случае можно, например, значение символа на выходе определить бросанием монеты.

В случае $p = 0$ (канал без шумов – ошибок не происходит) пропускная способность максимальна. Когда $p = 1$, то, подвергая инверсии все выходные символы, можно получить правильный результат.

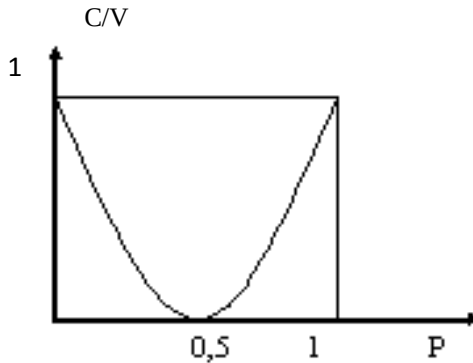


Рис. 16. Двоичный канал

Для представления о порядках величин C приведем примеры. Прямыми измерениями установлено, что пропускные способности зрительного, слухового и тактильного каналов связи человека имеют порядок 50 бит/секунд. Возможно ограничивающим фактором являются не сами рецепторы, а нервные волокна, передающие возбуждения. Если включить в канал и «исполнительные» органы человека (например, предложить ему нажимать педаль или кнопку в темпе получения сигналов), то пропускная способность близка к 10 бит/секунд. Интересно отметить, что многие бытовые технические устройства слабо согласованы с органами чувств человека. Например, канал телеви-

дения имеет пропускную способность в десятки миллионов бит/секунд.

4.4.4. Избыточность

Следствием ограничений на выбор источником знаков является также недоиспользование их как переносчиков информации. Известная априорная информация о вероятностях выбора отдельных знаков и их сочетаний приводит к уменьшению средней неопределенности выбора источником знака, а, следовательно, и переносимого им количества информации. При равновероятном и некоррелированном выборе ту же информационную нагрузку на знак можно обеспечить, используя алфавит меньшего объема. В связи с этим говорят об *избыточности алфавита* l источника сообщений или просто об избыточности источника.

Мерой избыточности служит величина D , показывающая, насколько хорошо используются знаки данного источника:

$$D = \frac{H_{\max}(Z) - H(Z)}{H_{\max}(Z)},$$

где $H_{\max}(Z)$ – максимально возможная энтропия, равная $\log_2 N$, $H(Z)$ – энтропия источника.

Если избыточность источника равна нулю, то формируемые им сообщения оптимальны в смысле наибольшего количества переносимой информации. Для передачи определенного количества информации I при отсутствии помех в этом случае необходимо $k_1 = I/H_{\max}(Z)$ знаков.

Поскольку энтропия сообщений, формируемых реальным источником, обладающим избыточностью, меньше максимальной, то для передачи того же количества информации I знаков требуется больше, а именно $k_2 = I/H(Z) > k_1$ знаков. Поэтому говорят также об избыточности знаков в сообщении или просто об избыточности сообщения, характеризуя ее тем же параметром D :

$$D = \frac{k_2 - k_1}{k_2} \Bigg/ = [H_{\max}(Z) - H(Z)]/[H_{\max}(Z)].$$

Избыточность нельзя рассматривать как признак несовершенства источника сообщений. Обычно она является следствием его физических свойств. Ограничения, существующие в любом естественном языке, связаны, например, с особенностями артикуляции, не позволяющими формировать слова, состоящие из произвольных сочетаний букв.

Последствия от наличия избыточности сообщений неоднозначны. С одной стороны, избыточные сообщения требуют дополнительных затрат на передачу, например, увеличения длительности передач или расширения практической ширины спектра канала связи, что нежелательно, с другой – при использовании сообщений, подчиняющихся априорно известным ограничениям, появляется возможность обнаружения и исправления ошибок, которые приводят к нарушению этих ограничений. Следовательно, наличие избыточности способствует повышению помехоустойчивости сообщений. Высокая избыточность большинства естественных языков обеспечивает, например, надежное общение людей даже при наличии у них акцентов и дефектов речи.

Однако при обмене информацией в автоматических системах естественная избыточность подлежит устранению. Это объясняется тем, что алгоритмы обнаружения и исправления ошибок, базирующихся на статистических закономерностях функционирования источника, оказываются слишком сложными для реализации их техническими средствами. В случае необходимости для повышения помехоустойчивости затем вводится «рациональная» избыточность, позволяющая обеспечить обнаружение и исправление наиболее вероятных и опасных по последствиям ошибок простыми техническими средствами. При низком уровне помех в канале связи устранение избыточности приводит к увеличению скорости передачи информации и может дать значительный экономический эффект.

Пример: Определить возможный эффект от устранения избыточности при передаче текста на русском языке.

Максимальная энтропия текста на русском языке (с учетом пренебрежения при передаче различиями в буквах е и ё, ь и ъ) равна 5 дв. ед. Энтропия с учетом неравномерного распределения вероятностей появления отдельных букв равна (4,42 дв. ед.). Имея сведения о переходных вероятностях и исходя из предположения, что текст представляет собой простую цепь Маркова, можно установить, что энтропия уменьшается до 3,52 дв. ед. Учет всех ограничений в языке, включая связи между словами, позволяет оценить минимальную величину энтропии значением 1,5 дв. ед. Таким образом, избыточность русского языка составляет

$$D = [H_{\max}(Z) - H(Z)]/[H_{\max}(Z)] = (5 - 1,5)/5 = 0,7.$$

Это означает, что каналы связи, построенные без учета ограничений, существующих в языке, и способные передавать равновероятные буквы, следующие друг за другом в любых сочетаниях, при передаче информации без помех текстом на русском языке используется всего на 30 %. Полное устранение избыточности позволило бы повысить эффективность их использования более чем в 3 раза!

Информационная избыточность показывает относительную недогруженность на символ алфавита и является безразмерной величиной

$$D = \frac{(H_{\max} - H)}{H_{\max}} = 1 - \frac{H}{H_{\max}},$$

где $\frac{H}{H_{\max}}$ – коэффициент сжатия.

Так как информация есть неопределенность, снимаемая при получении сообщения, то количество информации может быть представлено как произведение общего числа сообщений k на среднюю энтропию, приходящуюся на одно сообщение:

$$I = k \cdot H.$$

Для случая равновероятных и взаимонезависимых символов первичного алфавита количество информации в k -сообщениях алфавита m равно

$$I = k \cdot \log_2 m \text{ (бит)}.$$

Для неравновероятных алфавитов энтропия на символ алфавита:

$$H = \sum_{i=1}^m p_i \log_2 \left(\frac{1}{p_i} \right) = - \sum_{i=1}^m p_i \log_2 p_i \text{ (бит/символ)},$$

а количество информации в сообщении, составленном из k неравновероятных символов

$$I = k \cdot \sum_{i=1}^m p_i \log_2 p_i.$$

Кроме общего понятия избыточности существуют частные виды избыточности:

1) Избыточность, обусловленная неравновероятным распределением символов в сообщении

$$D_p = 1 - \left(\sum_i p_i \frac{\log_2 p_i}{\log_2 m} \right).$$

2) Избыточность, вызванная статистической связью между символами сообщения:

$$D_s = 1 - \frac{- \sum_i \sum_j p_{x_i} \cdot p(y_j/x_i) \cdot \log_2 p(y_j/x_i)}{\sum_i p_i \log_2 p_i}.$$

Полная информационная избыточность

$$D = D_s + D_p - D_s \cdot D_p.$$

Пример:

Сообщения составлены из алфавита a, b, c, d , $p(a)=0,2$; $p(b)=0,3$; $p(c)=0,4$; $p(d)=0,1$. Найти избыточность сообщений D ?

$$D = 1 - \frac{H}{H_{\text{макс}}};$$

Для алфавита из четырех букв максимальная энтропия:

$$H_{\text{макс}} = \log_2 m = \log_2 4 = 2 \frac{\text{бит}}{\text{символ}}.$$

Средняя энтропия на символ сообщения:

$$H = -\sum_i p_i \log_2 p_i = -(0,2 \cdot \log_2 0,2 + 0,3 \cdot \log_2 0,3 + 0,4 \cdot \log_2 0,4 + 0,1 \cdot \log_2 0,1) = 0,4644 + 0,5211 + 0,5288 + 0,3322 = 1,8465 \frac{\text{бит}}{\text{символ}}.$$

$$D = 1 - \frac{1,8465}{2} = 0,077$$

Контрольные вопросы и задания

1. Назовите основные меры информации.
2. Определите количество информации геометрическим методом.
3. Определите и сравните количество информации, полученной различными методами комбинаторики.
4. Вычислите количество информации мерой Хартли.
5. Покажите связь вероятности и информации.
6. Определите понятие энтропии.
7. Перечислите основные свойства энтропии ансамбля.
8. Как определить энтропию нескольких взаимозависимых ансамблей?
9. Как связаны между собой понятия количества информации и энтропии?
10. Как определяются потери информации в канале с шумом?
11. От чего зависит избыточность?
12. Поясните сущность понятия пропускной способности канала.
13. Сколько информации содержится в сообщении о том, что сумма очков на 2 подброшенных костях равна 9?
14. Укажите минимальное количество вопросов, позволяющих угадать день рождения любого человека при ответах «Да», «Нет».

15. Сообщение содержит информацию 33 бит и используется в нем 8 символов алфавита. Определите среднее количество символов в сообщении.

16. На экране индикатора сигнала, представляющего поле с 20-ю вертикальными и 10-ю горизонтальными полосами, появляется изображение объекта в виде яркостной отметки. Определить количество информации, содержащейся в сообщении. Объект находится в 50-м квадрате экрана.

17. Взаимодействие двух систем А и В описывается следующей матрицей:

$$P(A, B) = \begin{bmatrix} 0,4 & 0,1 & 0 \\ 0 & 0,2 & 0,1 \\ 0 & 0 & 0,2 \end{bmatrix}.$$

Определите безусловную энтропию системы А и системы В.

18. Задана матрица вероятностей системы, объединенной в одну систему из двух взаимосвязанных систем В и А:

$$p(A, B) = \begin{vmatrix} 0,3 & 0 & 0 \\ 0,2 & 0,3 & 0,1 \\ 0 & 0,1 & 0 \end{vmatrix}$$

Определите полную условную энтропию $H(A/B)$.

19. Определите общую избыточность некоторого 8-буквенного алфавита, если известно, что с учетом неравновероятности распределения символов, его энтропия $H = 2,7$ бит/символ, а с учетом взаимонезависимости букв энтропия этого алфавита уменьшается на 0,25 бит/символ.

20. Сообщения передаются в двоичном коде ($m = 2$). Время передачи $t_0 = 1$ с, длительность импульса, соответствующего 1, $t_1 = 5$ секунд. Определите скорость передачи информации для случаев: а) когда символы равновероятны и независимы; б) вероятность появления символа 0 – $p_0 = 0,37$, вероятность появления символа 1 – $p_1 = 0,63$.

5. ТЕХНОЛОГИИ РАБОТЫ С ИНФОРМАЦИЕЙ

5.1. Общие понятия теории кодирования информации

Одна из важных характеристик информационной системы – скорость и достоверность передаваемой информации. Здесь следует заметить, что теория информации рассматривает хранение информации как процесс передачи ее во времени.

Согласование статистических свойств источника сообщений и канала связи проводится с целью улучшения качества системы передачи. Качество оценивается по трем основным показателям:

- 1) достоверности;
- 2) средней скорости передачи;
- 3) сложности технической реализации системы.

Достигается согласование путем кодирования информации [3, 7, 9, 16, 17, 20]. В основной теореме Шеннона [32] о кодировании для дискретного канала без помех утверждается, что посредством преобразования сообщений в статистически независимые и равновероятные символы можно повысить скорость передачи вплоть до пропускной способности этого канала.

Под кодированием понимается представление сообщений в форме удобной для передачи по данному каналу [29].

Декодирование – операция восстановления сообщений по принятому сигналу.

На рис. 17 приведена общая схема передачи информации

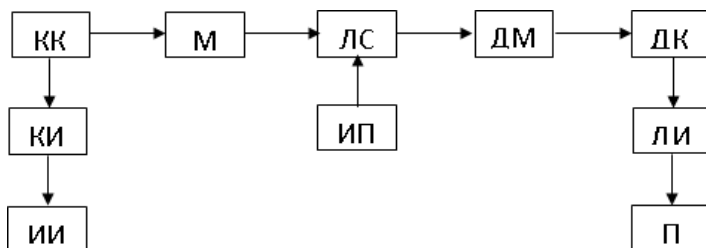


Рис. 17. Общая схема передачи информации

На рис. 17 введены следующие обозначения:

ИИ – источник информации, выдает сообщение, состоящее из последовательности букв первичного алфавита.

КИ – кодер источника преобразует эту последовательность в последовательность символов. Алфавит символов – вторичный алфавит меньше алфавита букв.

Таким образом, каждой букве соответствует некоторая последовательность символов, которая называется *кодовой комбинацией*. Число символов кодовой комбинации называется ее *значностью*.

Цель кодера источника – обеспечить такое кодирование, при котором путем устранения избыточности снижается среднее число символов, требующихся на букву сообщения (такое кодирование называется *эффективным* или *оптимальным*).

КК – кодер канала обеспечивает заданную достоверность при передаче или хранении информации, путем внесения избыточности (такое кодирование называется *помехоустойчивым*).

М – модулятор;

ЛС – линия связи, в которой на сигнал накладываются помехи ИП.

Искаженный сигнал поступает в демодулятор ДМ и преобразуется в символы. Декодер канала ДК и декодер источника ДИ выдают декодированное сообщение получателю П.

5. 2. Кодирование информации при передаче по дискретному каналу без помех

Передача или хранение сообщений сводится к передаче или хранению чисел [7]. Числа можно выразить в какой-либо системе счисления. Таким образом будет получен один из кодов, основанный на данной системе счисления.

Наиболее эффективной системой является система, обеспечивающая минимум произведения количества различных символов m на количество разрядов для выражения любого числа – *троичная* (Этот минимум виден на графике (рис. 18), где показана связь между величинами m и l при воспроизведении определенного десятичного большого числа $Q = 60\,000$).

Преимущества двоичной системы: логические элементы должны иметь всего два устойчивых состояния. Арифметические и логические действия также наиболее просто осуществляются в двоичной системе. Наиболее распространенная при кодировании и декодировании логическая операция – сложение по модулю два. В двоичной системе она также наиболее проста и определяется равенствами:

$$0 \oplus 0 = 0 \quad 1 \oplus 1 = 0$$

$$0 \oplus 1 = 1 \quad 1 \oplus 0 = 1$$

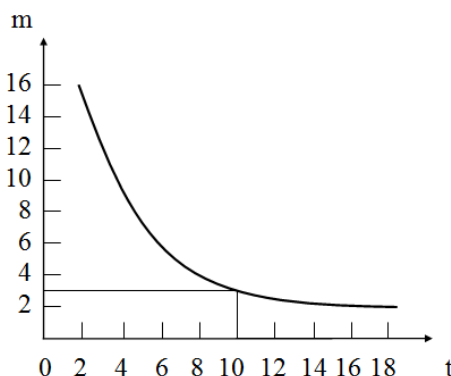


Рис. 18. График зависимости между величинами m и l при воспроизведении десятичного большого числа $Q \approx 60\,000$

5.2.1. Аналого-цифровые преобразователи (АЦП)

Техническая реализация процесса кодирования при непрерывном входном сигнале осуществляется аналого-кодовым преобразователем (чаще называемым сейчас АЦП) [7].

По способу получения цифрового эквивалента все разнообразие существующих преобразователей может быть разбито на три основные группы: преобразователи последовательного счета, преобразователи поразрядного уравнивания, преобразователи считывания. Для примера рассмотрим АЦП с развертывающим измерительным преобразованием (разновидность преобразователей последовательного счета).

На рис. 19 введены следующие обозначения:

ГКН – генератор линейно-изменяющегося компенсирующего напряжения U_K ;

И – схема совпадения;

ГИ – высокостабильный генератор импульсов;

БС – блок синхронизации;

ОС – орган сравнения;

СЧ – счетчик.

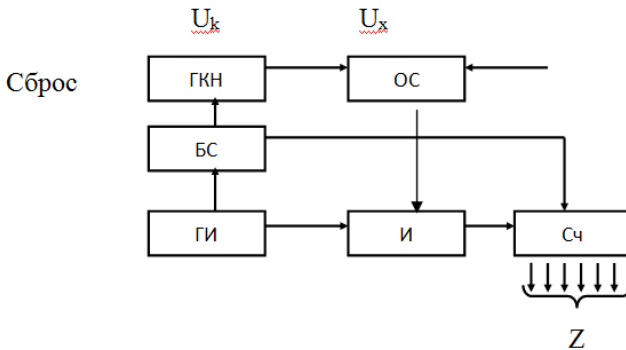


Рис. 19. АЦП с развертывающим измерительным преобразованием

В начале каждого цикла преобразования запускается генератор ГКН линейно-изменяющегося компенсирующего напряжения U_K . Одновременно сигнал с органа сравнения ОС открывает схему совпадения, и импульсы высокостабильного генератора начинают поступать в счетчик СЧ. В момент компенсации, когда преобразуемое напряжение U_x сравнивается с компенсирующим напряжением U_K , состояние органа сравнения изменяется, и доступ импульсов в счетчик прекращается. По окончании цикла развертки компенсирующего напряжения с блока синхронизации БС поступают импульсы на считывание цифрового эквивалента Z и сброс счетчика. Далее процесс повторяется. Временная диаграмма работы преобразователя имеет вид (рис. 20):

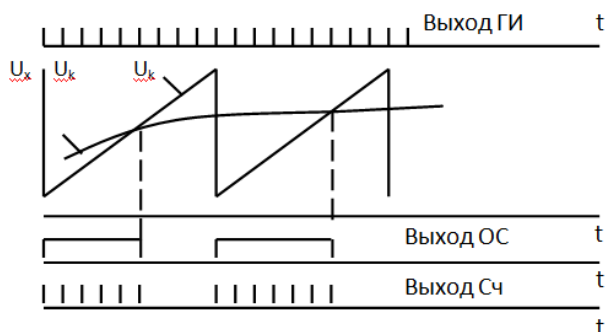


Рис. 20. Временная диаграмма работы преобразователя

5.2.2. Эффективное кодирование. Методика Шеннона – Фэно

В большинстве случаев знаки сообщений преобразуются в последовательности двоичных символов. При передаче десятичных цифр двоичным кодом максимально загруженными бывают только те символы вторичного алфавита, которые передают значения, являющиеся целочисленными степенями двойки. В остальных случаях тем же количеством символов может быть передано большее количество цифр. Например, тремя двоичными разрядами мы можем передать и цифру 5 и цифру 7, Фактически для передачи сообщения достаточно иметь длину кодовой комбинации

$$L \geq \frac{\log_2 m_1}{\log_2 m_2},$$

где m_1 и m_2 – соответственно объем первичного и вторичного алфавита. Поэтому для цифры 5 в двоичном коде можно записать $L \geq \frac{\log_2 5}{\log_2 2} = 2,32$ двоичных символов. Эту цифру необходи-

мо округлить до ближайшего целого числа, так как длина кода не может быть выражена дробным числом.

Избыточность от округления:

$$D_0 = \frac{k - \varphi}{k},$$

$$\varphi = \frac{\log_2 m_1}{\log_2 m_2},$$

где k – округление до ближайшего большего целого числа.

Если энтропия источника сообщений не равна максимальной энтропии для алфавита с данным количеством качественных признаков (имеются в виду качественные признаки алфавита, при помощи которых составляются сообщения), то это, прежде всего, означает, что сообщение данного источника могли бы нести большее количество информации, т. е. имеет место информационная избыточность.

В процессе кодирования, учитывая статистические свойства источника сообщений, можно минимизировать среднее число двоичных символов, требующихся для выражения одной буквы сообщения, что при отсутствии шума позволяет уменьшить избыточность, а значит, и время передачи информации.

Такое эффективное кодирование базируется на основной теореме Шеннона (теорема 1) для каналов без шума [32]:

Сообщения Z , составленные из букв некоторого алфавита, можно закодировать так, что среднее число двоичных символов на букву l_{cp} будет сколько угодно близко к энтропии источника этих сообщений $H(Z)$, но не меньше этой величины, т. е. $l_{cp} \rightarrow H(Z)$, но $l_{cp} \geq H(Z)$.

Для случая отсутствия статистической взаимосвязи между буквами конструктивные методы построения эффективных кодов даны впервые Шенноном и Фэнно. Код Шеннона и Фэнно (табл. 17 и 18) строится следующим образом:

$Z : z_1, z_2, \dots, z_n$ – буквы;

$p : p(z_1), p(z_2), \dots, p(z_n)$ – вероятности.

Вероятности $p(z_i)$ упорядочиваются по убыванию, множество Z делится на две группы так, чтобы суммы вероятностей в каждой подгруппе были по возможности одинаковы. Всем буквам верхней половины в качестве первого символа приписывается 1, а всем нижним – 0. Каждая из полученных групп в свою очередь разбивается на две подгруппы с одинаковыми суммарными вероятностями и т. д. Процесс повторяется до тех пор, пока в каждой подгруппе не останется по одной букве.

Пример: Сообщение состоит из восьми знаков с вероятностями, приведенными в табл. 17. Построить код Шеннона и Фэнно.

Решение:

В табл. 17 приведен построенный согласно вышеприведенной методике код Шеннона и Фэнно.

Таблица 17

Код Шеннона и Фэнно

Z	z_1	z_2	z_3	z_4	z_5	z_6	z_7	z_8
$p(Z)$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{64}$	$\frac{1}{128}$	$\frac{1}{128}$
Код	1	01	001	0001	00001	000001	0000001	0000000

Достигается наибольший эффект сжатия

$$H(Z) = -\sum_{i=1}^{\infty} p(z_i) \log_2 p(z_i) = \frac{127}{64},$$

$$l_{cp} = \sum_{i=1}^{\infty} p(z_i) n(z_i) = \frac{127}{64},$$

где $n(z_i)$ – число символов кодовой комбинации, соответствующей знаку z_i .

Пример: Сообщение состоит из восьми знаков с вероятностями, приведенными в табл. 18. Построить код Шеннона и Фэнно.

Решение:

В табл. 18 приведен построенный согласно вышеприведенной методике код Шеннона и Фэнно.

Таблица 18

Код Шеннона и Фэнно

z_1	0,22	11
z_2	0,2	101
z_3	0,16	100
z_4	0,16	01
z_5	0,1	001
z_6	0,1	0001
z_7	0,04	00001
z_8	0,02	00000

Энтропия $H(Z) = 2.76$ и среднее число символов $l_{cp} = 2.84$.

Осталась некоторая избыточность, которая может быть устранена, если перейти к кодированию достаточно большими блоками.

Пример: Сообщение состоит из двух знаков z_1 и z_2 с вероятностью появления $p(z_1) = 0.9$; $p(z_2) = 0.1$

$$l_{cp} = 1; H = 0,469$$

При кодировании блоков, содержащих по две буквы, получим следующие коды (табл. 19)

Таблица 19

Кодирование блоков

Блоки	Вероятности	Кодовые комбинации
$z_1 z_1$	0,81	0
$z_1 z_2$	0,09	01
$z_2 z_1$	0,09	001
$z_2 z_2$	0,01	000

Так как знаки статистически не связаны, вероятности блоков определяются как произведение вероятностей составляющих знаков. Среднее число символов $l_{cp} = 1,29$ и на букву l_{cp} равно 0,645.

Теоретический минимум может быть достигнут при кодировании блоков, включающих бесконечное число знаков.

Недостатки методики Шеннона–Фэнно – неоднозначность. От этих недостатков избавлена методика Хаффмана.

5.3. Методика Хаффмана

Методика Хаффмана гарантирует однозначное построение кода с наименьшим для данного распределения вероятностей средним числом символов на букву.

Для двоичного кода методика сводится к следующему: буквы алфавита сообщений выписываются в основной столбец в порядке убывания вероятностей; две последние буквы объединяются в одну вспомогательную букву, которой приписывают суммарную вероятность. Вероятности букв, не участвовавших в объединении, и полученная суммарная вероятность снова располагаются в порядке убывания вероятностей в дополнительном столбце, а две последние объединяются. Процесс продолжается до тех пор, пока не получим единственную вспомогательную букву с вероятностью, равной единице.

Для составления кодовой комбинации соответствующей данному знаку необходимо проследить путь перехода знака по строкам и столбцам таблицы (табл. 18). Для наглядности строится кодовое дерево. Из точки, соответствующей вероятности 1, направляются две ветви, причем, ветви с большей вероятностью присваивается символ 1, а с меньшей – 0. Такое последовательное ветвление продолжается до тех пор, пока не будет достигнута вероятность каждой буквы.

Пример: Пусть ансамбль состоит из восьми букв $z_1, z_2, z_3, z_4, z_5, z_6, z_7, z_8$. Задан вектор их вероятностей $\bar{p} = [0,22; 0,2; 0,16; 0,16; 0,1; 0,1; 0,04; 0,02]$. Результаты кодирования сведены в табл. 20.

Таблица 20

Путь перехода знака								
Буквы	Вероятности	1	2	3	4	5	6	7
z_1	0,22	0,22	0,22	0,26	0,32	0,42	0,58	1
z_2	0,20	0,20	0,20	0,22	0,26	0,32	0,42	
z_3	0,16	0,16	0,16	0,20	0,22	0,26		
z_4	0,16	0,16	0,16	0,16	0,20			
z_5	0,10	0,10	0,16	0,16				
z_6	0,10	0,10	0,10					
z_7	0,04	0,06						
z_8	0,02							

Затем, двигаясь по кодовому дереву (рис. 21) сверху вниз можно записать для каждой буквы соответствующую ей кодовую комбинацию:

Построенные по указанным выше методикам коды с неравномерным распределением символов, имеющие минимальную среднюю длину кодового слова, называют *оптимальными неравномерными кодами (ОНК)*.

Максимально эффективными будут те ОНК, у которых:

$$\log_2 m = \sum_{i=1}^N p_i l_i = l_{cp} = H \cdot$$

z_1	z_2	z_3	z_4	z_5	z_6	z_7	z_8
01	00	111	110	100	1011	10101	10100

Эффективность ОНК оценивают при помощи коэффициента статистического сжатия:

$$K_{c.c} = \frac{H_{\max}}{l_{c.p}},$$

и коэффициента относительной эффективности:

$$K_{o.э} = \frac{H}{l_{c.p}}$$

Коды Шеннона–Фэнно и Хаффмана удовлетворяют требованиям префиксности: ни одна комбинация кода не совпадает с началом более длинной комбинации.

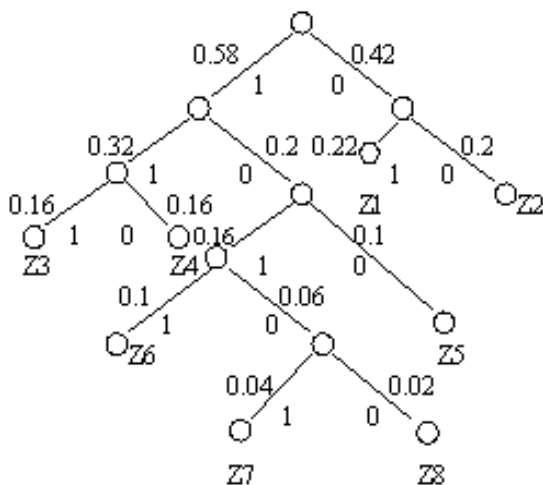


Рис. 21. Кодовое дерево

5.3.1. Алгоритм LZW

Впервые обнаружен в 1977 г. Абрахам Лемпел и Якоб Зив (Израиль). Усовершенствованный вариант предложен Terry Welch в 1984 году.

Сжатие происходит, когда код заменяет строку символов. Коды, генерируемые LZW-алгоритмом, могут быть любой длины, но они должны содержать больше бит, чем одиночный символ. Первые 256 кодов (когда используются 8-битные символы) по умолчанию соответствуют стандартному набору символов. Остальные коды соответствуют обрабатываемым алгоритмом строкам.

Рассмотрим работу алгоритма на примере 12-битных кодов. Значения 0–255 соответствуют отдельным байтам, а коды 256 – 4095 – подстрокам.

Каждый раз, когда генерируется новый код, новая строка добавляется в таблицу строк. LZW постоянно проверяет, является ли строка уже известной, и если так, то выводит существующий код без генерации нового.

Для демонстрации работы используем входную строку
/WED/WE/WEE/WEB/WET

На первом шаге цикла алгоритм выполняет проверку на наличие строки “/W”. Так как 256 символов уже определены для кодов 0 – 255, то первой определенной строке может быть поставлен в соответствие код 256. После этого система читает следующую букву “E”, добавляет вторую строку (“WE”) в таблицу и выводит код для буквы “W”. Этот процесс повторяется до тех пор, пока вторая подстрока, состоящая из прочитанных символов “/” и “W”, не сопоставится со строковым номером 256. В этом случае система выводит код 256 и добавляет трехсимвольную подстроку в таблицу.

Этот процесс продолжается до тех пор, пока не исчерпается входной поток и все коды не будут выведены (табл. 20).

(Если использовать 9-битные коды для вывода, то 19-символьная входная строка будет преобразована в 3,5-символьную выходную строку).

В отличие от примера сжатие обычно не начинается до тех пор, пока не будет построена достаточно большая таблица, обычно, после прочтения порядка 100 входных байт.

Методы теории информации позволяют доказать, что кодирование методом Лемпеля – Зива асимптотически оптимально. Это означает, что для очень длинного текста избыточность исчезает, т. е. среднее число бит, необходимое для кодирования одного символа, стремится к энтропии текста. Практически достижимая степень сжатия для длинных текстов составляет 50–60 %.

/WED/WE/WEE/WEB/WET

Таблица 21

Сжатие

Вход (символы)	Выход (коды)	Новые коды и соответствующие строки
/W	/	256 = /W
E	W	257 = WE
D	E	258 = ED
/	D	259 = D/
WE	256	260 = /WE
/	E	261 = E/
WEE	260	262 = /WEE
/W	261	263 = E/W
EB	257	264 = WEB
/	B	265 = B/
WET	260	266 = /WET
<eof>	T	

5.3.2. Арифметическое кодирование

При арифметическом кодировании текст представляется вещественными числами в интервале от 0 до 1. По мере кодирования текста отображающий его интервал уменьшается, а количество битов для его представления возрастает. Очередные символы текста сохраняют величину интервала исходя из значений их вероятностей, определяемых моделью.

Более вероятные символы делают это в меньшей степени, чем менее вероятные и, следовательно, добавляют меньше битов к результату.

Перед началом работы соответствующий тексту интервал есть $[0;1)$. При обработке очередного символа его ширина сужается за счет выделения этому символу части интервала.

Например, применим к тексту “eaii!” алфавита $\{a, e, i, o, u, !\}$ модель с постоянными вероятностями (табл. 22).

И кодировщику и декодировщику известно, что в самом начале интервал есть $[0;1)$.

После просмотра первого символа “e” кодировщик сужает интервал до $[0.2; 0.5)$, который модель выделяет этому символу. Второй символ “a” сузит этот интервал до первой пятой его части, так как для “a” выделен фиксированный интервал $[0.0; 0.2)$. В результате получим рабочий интервал $[0.0; 0.26)$ так как предыдущий интервал имел ширину в 0,3 единицы и одна пятая от него есть 0,06. Следующему символу “i” соответствует фиксированный интервал $[0.5; 0.6)$, что применительно к рабочему интервалу $[0.2; 0.26)$ суживает его до интервала $[0.23; 0.236)$.

Таблица 22

Арифметическое кодирование

Символ	Вероятность	Интервал
A	.2	$[0.0; 0.2]$
E	.3	$[0.2; 0.5]$
I	.1	$[0.5; 0.6]$
O	.2	$[0.6; 0.8]$
U	.1	$[0.8; 0.9]$
!	.1	$[0.9; 1.0]$

Продолжая в том же духе, имеем

В начале	$[0.0; 1.0)$
После просмотра “e”	$[0.2; 0.5)$
После просмотра “a”	$[0.2; 0.26)$
После просмотра “i”	$[0.23; 0.236)$
После просмотра “i”	$[0.233; 0.2336)$
После просмотра “!”	$[0.23354; 0.2336)$

Предположим, что все, что декодировщик знает о тексте, это конечный интервал $[0.23354; 0.2336)$. Он сразу же понимает, что первой закодированный символ есть “е”, так как итоговый интервал целиком лежит в интервале, выделенном моделью этому символу.

5.4. Помехоустойчивое кодирование

Помехоустойчивое кодирование базируется на теореме Шеннона № 2 (для канала с помехами):

При любой производительности источника сообщений, меньшей чем пропускная способность канала, существует такой способ кодирования, который позволяет обеспечить передачу всей информации, создаваемой источником сообщений, со сколь угодно малой вероятностью ошибки $[3, 9, 22, 28, 30, 31, 34]$.

Это достигается ценой введения при кодировании *избыточности*, которая позволяет так выбрать передаваемые последовательности символов, чтобы они удовлетворяли дополнительным условиям, проверка которых на приемной стороне дает возможность обнаруживать и исправлять ошибки.

У большинства существующих в настоящее время помехоустойчивых кодов указанные условия являются следствием их алгебраической структуры.

Алгебраические коды делятся на два больших класса:

- 1) блочные;
- 2) непрерывные.

Блочные коды: каждой букве сообщения сопоставляется блок из p - символов.

Блочный код называют *равномерным*, если p остается постоянным для всех букв сообщения.

Блочные коды бывают разделимыми и неразделимыми

Отличие неразделимых кодов в том, что *невозможно* разделить символы выходной последовательности на *информационные* и *проверочные*.

Непрерывными (древовидными) называют такие коды, в которых введение избыточных символов в кодируемую последовательность информационных символов осуществляется непрерывно, без деления ее на независимые.

Непрерывные коды также могут быть разделимыми и неразделимыми.

5.4.1. Теоретические основы блочных кодов

Способность кода обнаруживать и исправлять ошибки обусловлена наличием избыточных символов. На вход кодирующего устройства поступает последовательность из k информационных двоичных символов. На выходе ей соответствует последовательность из n двоичных символов, причем $n > k$. Всего может быть 2^k различных входных и 2^n различных выходных последовательностей. Из общего числа 2^n выходных последовательностей только 2^k последовательностей соответствуют входным, это *разрешенные кодовые комбинации*. Остальные $2^n - 2^k$ возможных выходных последовательностей для передачи информации не используются, это *запрещенные кодовые комбинации*.

Так как каждая из 2^k разрешенных комбинаций в результате действия помех может трансформироваться в любую другую, но всего имеется $2^k \cdot 2^n$ возможных случаев передачи. В это число входят: 2^k случаев безошибочной передачи; $2^k(2^k - 1)$ случаев перехода в другие разрешенные комбинации, что соответствует не обнаруживаемым ошибкам, $2^k(2^n - 2^k)$ случаев перехода в неразрешенные комбинации, которые могут быть обнаружены.

Часть обнаруживаемых ошибочных кодовых комбинаций от общего числа возможных случаев передачи составляет

$$\frac{2^n - 2^k}{2^n} = 1 - \frac{2^k}{2^n}.$$

Определим обнаруживающую способность кода, каждая комбинация которого содержит всего один избыточный символ ($n = k + 1$). Общее число выходных последовательностей составляет 2^{k+1} , т. е. вдвое больше общего числа кодируемых входных последовательностей. За подмножество разрешенных кодовых комбинаций можно принять, например, подмножество 2^k -комбинаций, содержащих четное число единиц (или нулей).

При кодировании к каждой последовательности из k информационных символов добавляется один символ (0 или 1) такой, чтобы число единиц в кодовой комбинации было четным. Искажение любого нечетного числа символов переводит разрешенную кодовую комбинацию в подмножество запрещенных комбинаций, что обнаруживается на приемной стороне по нечетности числа единиц. Часть опознанных ошибок составляет

$$1 - \frac{2^k}{2^{k+1}} = \frac{1}{2}.$$

Рассмотрим случай исправления ошибок. Любой метод декодирования можно рассматривать как правило разбиения всего множества запрещенных кодовых комбинаций на 2^k непересекающихся множеств M_i , каждое из которых ставится в соответствие одной из разрешенных комбинаций. При получении запрещенной комбинации, принадлежащей подмножеству M_i , принимают решение, что передавалась разрешенная комбинация A_i . Ошибка будет исправлена в тех случаях, когда полученная комбинация действительно образовалась из A_i , т. е. $2^n - 2^k$ - случаях.

Кратностью ошибки r называют количество искаженных символов в кодовой комбинации.

Связь корректирующей способности кода с кодовым расстоянием

Степень отличия любых двух кодовых комбинаций характеризуется расстоянием между ними в смысле Хэмминга, или просто кодовым расстоянием d .

(Чтобы получить кодовое расстояние между двумя комбинациями двоичного кода, достаточно подсчитать число единиц в сумме этих комбинаций по модулю 2).

Например:

$$\begin{array}{r} 1001111101 \\ 1100001010 \\ \hline 0101110111 \end{array}$$

Минимальное расстояние, взятое по всем параметрам кодовых разрешенных комбинаций кода, называют *минимальным кодовым расстоянием* d_0 .

Декодирование осуществляется по методу *максимального правдоподобия*, если принятая кодовая комбинация отождествляется с той разрешенной, которая находится от нее на наименьшем кодовом расстоянии. При использовании этого метода:

1. Для обнаружения r ошибок, требуется, чтобы $d_0 \geq r + 1$;
2. Для исправления всех ошибок кратности до S :
 $d_0 \geq 2S + 1$;
3. Для исправления всех ошибок кратности S и одновременного обнаружения всех ошибок кратности $r (r \geq S)$ $d_0 \geq r + S + 1$.

Показатели качества корректирующего кода – избыточность

$$R_n = \frac{n - k}{n}$$

или

$$R_k = \frac{n - k}{k}, \quad (5.1)$$

где n – общая длина кода; k – число информационных разрядов (n_i).

Оптимальными называются коды, обеспечивающие заданную корректирующую способность при минимально возможной избыточности. (Для практических расчетов при определении числа контрольных разрядов кодов с минимальным кодовым расстоянием $d_0 = 3$ удобно пользоваться выражениями $n_{kl(2)} = [\log(n + 1)]$ – если известна длина полной кодовой комбинации n и $n_{kl(2)} = [\log \{ (n_i + 1) + [\log_2 (n_i + 1)] \}]$, если при расчетах удобнее исходить из заданного числа информационных символов n_i (в общих выражениях квадратные скобки означа-

ют, что берется округленные значения до ближайшего числа в большую сторону)).

5.4.2 Линейные групповые коды

Линейными называются коды, в которых проверочные символы представляют собой линейные комбинации информационных символов. Для двоичных кодов в качестве линейной операции используют сложение по модулю 2.

Последовательность нулей и единиц, принадлежащих данному коду, называется кодовым вектором.

Вес кодового вектора равен числу его ненулевых компонентов. Свойство группового кода: минимальное кодовое расстояние между кодовыми векторами группового кода равно минимальному весу ненулевых кодовых векторов.

При построении групповых кодов удобно задавать их порождающей матрицей C . Порождающая матрица может быть представлена двумя матрицами I и P (информационной и проверочной).

Число столбцов матрицы P равно n_k , число столбцов матрицы I равно n_u :

$$C_{n:nu} = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1nu} \\ a_{21} & a_{22} & \dots & a_{2nu} \\ \dots & \dots & \dots & \dots \\ a_{nu1} & a_{nu2} & \dots & a_{nunu} \end{vmatrix} \begin{vmatrix} P_{11} & P_{12} & \dots & P_{1nk} \\ P_{21} & P_{22} & \dots & P_{2nk} \\ \dots & \dots & \dots & \dots \\ P_{nu} & P_{nu2} & \dots & P_{nunk} \end{vmatrix} = |I| |P|$$

В качестве матрицы I удобно брать единичную матрицу в канонической форме:

$$I_{nu} = \begin{vmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \end{vmatrix}$$

При выборе матрицы P исходят из следующих соображений: чем больше единиц в разрядах проверочной матрицы P ,

тем ближе соответствующий порождающий код к оптимальному, с другой стороны, число единиц в матрице Π определяет число сумматоров по модулю 2 в шифраторе и дешифраторе, т.е. чем больше единиц в матрице Π , тем сложнее аппаратура. Вес каждой строки матрицы Π должен быть не менее $W_{\Pi} = d_0 - W_u$, где W_u – вес соответствующей строки матрицы I .

Если матрица I – единичная, то $W_u = 1$.

При соблюдении перечисленных условий любую порождающую матрицу группового кода можно привести к следующему виду:

$$C_{n,nu} = \begin{vmatrix} a_1 & a_2 & a_3 & \dots & a_{nu} & p_1 & p_2 & \dots & p_{(n-nu)} \\ 1 & 0 & 0 & \dots & 0 & p_{11} & p_{12} & \dots & p_{1(n-nu)} \\ 0 & 1 & 0 & \dots & 0 & p_{21} & p_{22} & \dots & p_{2(n-nu)} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & p_{nu1} & p_{nu2} & \dots & p_{nu(n-nu)} \end{vmatrix},$$

называемому *канонической формой порождающей матрицы*.

Пример. Построение порождающей матрицы.

1. Построить матрицу для группового кода, способного исправлять одиночную ошибку при передаче 16 символов первичного алфавита.

Решение:

Так как число информационных разрядов кода $n_u = 4$ ($16 = 2^4$), то число строк образующей матрицы C должно быть равно 4.

Число столбцов матрицы C равно длине кода n : $n = n_u + n_k$, где n_k – число корректирующих разрядов. Для кодов $cd_0 = 3$ ($d_0 = 2r + 1 = 3$):

$$n_k = [\log_2 2(n_u + 1) + \{\log_2 2(n_u + 1)\}] = \log_2 (5 + 3).$$

Тогда число столбцов, содержащих контрольные разряды, равно 3.

$$n = n_u + n_k = 4 + 3 = 7.$$

Так как для исправления одиночной ошибки $d_0 = 3$, вес каждой строки проверочной матрицы Π должен быть

$$W_{\Pi} \geq d_0 - W_u$$

В качестве строк проверочной матрицы можно взять трехзначные двоичные комбинации с числом единиц, большим или равным 2 ($d=3$), а $W_u = 1$, потому что матрицу информационных разрядов лучше брать единичную: 111; 110; 101; 011. Окончательный вид производящей матрицы таков:

$$C_1 = \begin{vmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{vmatrix}$$

или

$$C_2 = \begin{vmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{vmatrix},$$

или

$$C_3 = \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{vmatrix}$$

и т. д.

Строки образующей матрицы C представляют собой n_u -комбинаций искомого кода. Остальные комбинации строятся на основании свойства линейных кодов: сумма кодовых векторов линейного кода дает вектор, принадлежащий данному коду.

Если известна информационная часть кода, то значения проверочных символов определяются по формуле

$$p_j = \sum_{i=1}^{n_u} p_{ij} a_i = p_{1j} a_1 + p_{2j} a_2 + \dots + p_{n_u j} a_{n_u}. \quad (5.2)$$

Примеры нахождения группового кода по заданной производящей (образующей) матрице.

2. Построить групповой код по заданной производящей матрице

$$C_{7:4} = \begin{vmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{vmatrix}.$$

Решение: Число строк матрицы $n_u = 4$, следовательно, число возможных информационных комбинаций $N = 2n_u = 2^4 = 16$.

Одна из 16 комбинаций есть 000 0000. Четыре комбинации кода есть четыре строки образующей матрицы. Все остальные комбинации есть результат всех вариантов суммирования этих строк. Таким образом, комбинации корректирующего кода имеет вид

- | | | | |
|------------|------------|-------------|--------------|
| 1) 0000000 | 5) 0010101 | 9) 0001011 | 13) 0011110 |
| 2) 1000111 | 6) 1010010 | 10) 1001100 | 14) 1011001. |
| 3) 0100110 | 7) 0110011 | 11) 0101101 | 15) 0111000 |
| 4) 1100001 | 8) 1110100 | 12) 1101010 | 16) 1111111 |

3. Пусть информационная часть кода, построенного по образующей матрице, приведенной в примере 13, имеет вид 1101. Найти проверочные разряды этого кода.

Решение:

Воспользуемся формулой (5.2).

Имеем

$$P_1 = \sum_{i=1} P_{i1} a_i = 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 0 + 0 \cdot 1 = 0$$

$$P_2 = \sum_{i=1} P_{i2} a_i = 1 \cdot 1 + 1 \cdot 1 + 0 \cdot 0 + 1 \cdot 1 = 1;$$

$$P_3 = 1 \cdot 1 + 0 \cdot 1 + 1 \cdot 0 + 1 \cdot 1 = 0.$$

Окончательный вид: 1 101 010; (этот код можно получить, суммируя первую, вторую и четвертую строки образующей матрицы).

Поиск ошибок в полученных кодах осуществляется в процессе декодирования с помощью вычисления синдрома, состоящего из проверочного вектора $S_1, S_2, \dots, S_{n_k}$, элементы которого находятся из соотношения:

$$S_j = P_j + \sum_{i=1} P_{ij} \cdot a_i, \quad (5.3)$$

где p_j ($j=1, 2, \dots, n_k$) и a_i ($i=1, 2, \dots, n_u$) – значения j -го проверочного и i -го информационного разрядов кода, поступающего в декодер на приемной строке.

Для того чтобы определить по виду синдрома, в каком разряде произошла ошибка, строится проверочная матрица H , которая представляет собой транспонированную матрицу P , дополненную единичной матрицей In_k , число столбцов которой равно числу проверочных разрядов. $H = |P^T In_k|$.

Столбцы такой матрицы представляют собой значения синдрома для разряда, соответствующего номеру столбца матрицы H .

Пример: Определение ошибки в групповом коде.

1. Принятый линейный групповой код с образующей матрицей:

$$C_{7,4} = \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{vmatrix}$$

И
П

имеет вид 0100110.

Решение:

Находим синдром:

$$S_1 = p_1 + a_2 + a_3 + a_4 = 1 + 1 + 0 + 0 = 0;$$

$$S_2 = p_2 + a_1 + a_3 + a_4 = 1 + 0 + 0 + 0 = 1;$$

$$S_3 = p_3 + a_1 + a_2 + a_4 = 0 + 0 + 0 + 1 = 1.$$

Находим проверочную матрицу H :

$$H = \begin{array}{cccccc|c} a_1 & a_2 & a_3 & a_4 & p_1 & p_2 & p_3 & \\ \hline 0 & 1 & 1 & 1 & 1 & 0 & 0 & S_1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & S_2 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & S_3 \end{array}.$$

Синдром 011 показывает, что в первом разряде ошибка, т. е. символ следует заменить на обратный.

Тривиальные систематические коды.

Код Хэмминга

Систематические коды представляют собой такие коды, в которых информационные и корректирующие разряды расположены по строго определенной системе и всегда занимают строго определенные места в кодовых комбинациях. Систематические коды являются равномерными, т. е. все комбинации кода с заданными корректирующими способностями имеют одинаковую длину. Групповые коды также являются систематическими, но не все систематические коды могут быть отнесены к групповым.

Тривиальные систематические коды могут строиться, как и групповые, на основе производящей матрицы. *Код Хэмминга* является типичным примером систематического кода. Однако при его построении к матрицам обычно не прибегают. Для вычисления основных параметров кода задается либо количество информационных символов n_i , либо количество информационных комбинаций $N = 2^{n_i}$. Число корректирующих разрядов зависит от числа информационных и количества ошибок, которые требуется обнаруживать или исправлять.

Соотношения между n , n_i и n_k для кода Хэмминга представлены в табл. 23.

Таблица 23

n	$n_{и}$	$n_{к}$	n	$n_{и}$	$n_{к}$
1	0	1	9	5	4
2	0	2	10	6	4
3	1	2	11	7	4
4	1	3	12	8	4
5	2	3	13	9	4
6	3	3	14	10	4
7	4	3	15	11	4
8	4	4	16	11	5

Зная основные параметры корректирующего кода, определяют, какие позиции сигналов будут рабочими, а какие контрольными. Как показала практика, номера контрольных символов удобно выбирать по закону 2^i , где $i = 0, 1, 2$ и т. д. – натуральный ряд чисел. Номера контрольных символов в этом случае будут соответственно: 1, 2, 4, 8, 16, 32 и т. д.

Затем определяют значения контрольных коэффициентов (0 или 1), руководствуясь следующим правилом: *Сумма единиц на контрольных позициях должна быть четной*. Если эта сумма четна, то значение контрольного коэффициента – 0, в противном случае – 1.

Проверочные позиции выбираются следующим образом: Составляется таблица для ряда натуральных чисел в двоичном коде. Число строк таблицы

$$n = n_{и} + n_{к}$$

Первой строке соответствует проверочный коэффициент a_1 второй a_2 и т. д. Затем выявляют проверочные позиции, выписывая коэффициенты по следующему принципу: в первую проверку входят коэффициенты, номера которых содержат в младшем разряде 1, т. е. $a_1, a_3, a_5, a_7, a_9, a_{11}$ и т. д.; во вторую – коэффициенты, содержащие 1 во втором разряде, т. е. $a_2, a_3, a_6, a_7, a_{10}, a_{11}$ и т. д.; в третью проверку – коэффициенты, которые содержат 1 в третьем разряде, и т. д. Номера проверочных коэффициентов соответствуют номерам проверочных позиций, что позволяет составить общую таблицу проверок (табл. 24).

Таблица 24

№ про- верки	Проверочные позиции	№ конт- рольно- го сим- вола
1	1 3 5 7 9 11 . . .	1
2	2, 3, 7, 10, 11, 14, 15, 18, 19, 22, 23, 26, 27, 30, 31 . . .	2
3	4, 6, 7, 12, 13, 14, 15, 20 21, 22, 23, 28, 29, 30, 31 . . .	3
4	8, 9, 10, 11, 12, 13, 14, 15, 24, 25, 26, 27, 28, 29, 30, 31 40, 41, 42, 43, 44, 45, 46 . . .	4

Старшинство разрядов считается слева направо, а при проверке – сверху вниз. Порядок проверок показывает также порядок следования разрядов в полученном двоичном коде.

Если в принятом коде есть ошибка, то результаты проверок по контрольным позициям образуют двоичное число, указывающее номер ошибочной позиции. Исправляют ошибку, изменяя символ ошибочной позиции на обратный.

Задача: Построить код Хэмминга для информационной комбинации 0101. Показать процесс обнаружения ошибки.

Решение: Количество информационных разрядов $n_i = 4$. По табл. 23 находим значения общей длины кодовой комбинации и число контрольных символов: $n_k = 3$ и $n = 7$. Так как общее количество символов кода $n = 7$, то контрольные коэффициенты займут три позиции: 1, 2 и 4.

Корректирующий код без значений контрольных коэффициентов будет иметь вид

$$K_1 K_2 0 K_3 101.$$

Пользуясь таблицей проверочных позиций (табл. 24), определяем значения контрольных коэффициентов.

Первая проверка: сумма $\Pi_1 \oplus \Pi_3 \oplus \Pi_5 \oplus \Pi_7$ должна быть четной, а сумма $K_1 \oplus 0 \oplus 1 \oplus 1$ будет четной при $K_1 = 0$.

Вторая проверка: сумма $P_2 \oplus P_3 \oplus P_6 \oplus P_7$ должна быть четной, а сумма $K_4 \oplus 0 \oplus 0 \oplus 1$ будет четной при $K_2 = 1$.

Третья проверка: сумма $P_4 \oplus P_5 \oplus P_6 \oplus P_7$ должна быть четной, а сумма $K_3 \oplus 1 \oplus 0 \oplus 1$ будет четной при $K_3 = 0$.

Окончательно корректирующий код принимает вид

0 1 0 0 1 0 1

Предположим, в результате искажений в канале связи вместо 0100101 было принято 0100111. Для обнаружения ошибки проводим проверки на четность, аналогичные проверкам при выборе контрольных коэффициентов.

Первая проверка: сумма $P_1 \oplus P_3 \oplus P_5 \oplus P_7 = 0 \oplus 0 \oplus 1 \oplus 1$ четна. В младший разряд номера ошибочной позиции записываем 0.

Вторая проверка: сумма $P_2 \oplus P_3 \oplus P_6 \oplus P_7 = 1 \oplus 0 \oplus 1 \oplus 1$ нечетна. Во второй разряд номера ошибочной позиции записываем 1.

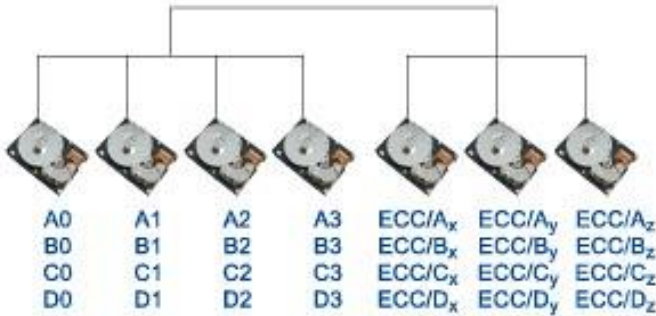
Третья проверка: сумма $P_4 \oplus P_5 \oplus P_6 \oplus P_7 = 0 \oplus 1 \oplus 1 \oplus 1$ нечетна. В третий разряд номера ошибочной позиции записываем 1.

Номер ошибочной позиции $110 = 6$. Следовательно, символ шестой позиции следует изменить на обратный.

Код Хэмминга используется в некоторых прикладных программах в области хранения данных, особенно в RAID 2; кроме того, метод Хэмминга давно применяется в памяти типа ECC и позволяет «на лету» исправлять однократные и обнаруживать двукратные ошибки RAID 2.

Отказоустойчивый дисковый массив с использованием кода Хэмминга (HammingCode ECC).

Рассмотрим схему резервирования данных с использованием кода Хэмминга (Hammingcode) для коррекции ошибок. Поток данных разбивается на слова, причем размер слова соответствует количеству дисков для записи данных. Для каждого слова вычисляется код коррекции ошибок, который записывается на диски, выделенные для хранения контрольной информации. Их число равно количеству бит в слове контрольной суммы.



RAID 2 не получил коммерческого применения.

Если слово состоит из четырех бит, то под контрольную информацию отводятся три диска. RAID 2 – один из немногих уровней, позволяющих обнаруживать двойные ошибки и исправлять «на лету» одиночные. При этом он является самым избыточным среди всех уровней с контролем четности. Эта схема хранения данных не получила коммерческого применения, поскольку плохо справляется с большим количеством запросов.

Преимущества:

- достаточно простая реализация;
- коррекция ошибок «на лету»;
- очень высокая скорость передачи данных;
- при увеличении количества дисков накладные расходы уменьшаются.

Недостатки:

- низкая скорость обработки запросов;
- высокая стоимость;
- большая избыточность

5.4.3. Циклическое кодирование

5.4.3.1. Построение циклических кодов. Обнаружение и исправление ошибок

Циклическим называется групповой код, все строки обрабатываемой матрицы которого могут быть получены циклическим

сдвигом одной комбинации, называемой образующей для данного кода.

При описании циклических кодов удобно любое n -разрядное двоичное число записывать в виде многочлена $(n-1)$ -й степени.

Например, для комбинации 10101

$$G(x) = 1 \cdot x^4 + 0 \cdot x^3 + 1 \cdot x^2 + 0 \cdot x^1 + 1 \cdot x^0 = x^4 + x^2 + 1.$$

Это позволяет свести действия над комбинациями к действиям над многочленами. При этом сложение двоичных многочленов сводится к сложению по $\text{mod } 2$ коэффициентов при равных степенях переменной x , умножение – к умножению по обычному правилу перемножения степенных функций, (при этом коэффициенты при одинаковых степенях складываются по $\text{mod } 2$), а деление – к делению по правилам деления степенных функций (при этом вычитание заменяется суммированием по $\text{mod } 2$). Циклический сдвиг соответствует простому умножению многочлена на x .

Идея построения циклических кодов базируется на использовании неприводимых в коде двоичных чисел многочленов $P(x)$.

Неприводимый многочлен не может быть представлен в виде произведения многочленов низших степеней, он делится только на единицу и сам на себя без остатка.

Разрешенные кодовые комбинации циклического кода $F(x)$ могут быть получены двумя способами.

1. Умножением кодовой комбинации $Q(x)$ простого кода на одноклен x^r и добавлением к этому произведению $R(x)$, полученного в результате деления произведения $Q(x) \cdot x^r$ на образующий многочлен $P(x)$ r -й степени (r – число корректирующих разрядов):

$$F(x) = Q(x) x^r + R(x).$$

2. Умножением кодовой операции $Q(x)$ простого кода на образующий многочлен:

$$F(x) = Q(x) P(x).$$

Примеры неприводимых многочленов (см. приложение 3):

1)	11	$m = 1,$	$x + 1$	$P(x^1);$
2)	111	$m = 2,$	$x^2 + x + 1$	$P(x^2);$
3)	1011	$m = 3,$	$x^3 + x + 1$	$P_1(x^3);$
	1101	$m = 3,$	$x^3 + x^2 + 1$	$P_2(x^3);$
4)	10011	$m = 4,$	$x^4 + x + 1$	$P_1(x^4);$
	11001	$m = 4,$	$x^4 + x^3 + 1$	$P_2(x^4);$
	11111	$m = 4,$	$x^4 + x^3 + x^2 + 1$	$P(x).$

При использовании первого метода получается разделимый код, при использовании второго – неразделимый.

Для формирования строк производящей матрицы по первому способу образования циклического кода берут не произвольные комбинации кода $Q(x)$, а лишь те из них, которые содержат единицу в одном разряде $Q_i(x)$, ($i = 1 - k$).

При втором способе образования циклического кода производящая матрица $G_{n,k}$ формируется умножением образующего полинома $P(x)$ степени g на одночлен x^{k-1} и последующих $(k-1)$ сдвигов полученной комбинации.

Пример: Дано: $n = 7$, $k = 4$, $P(x) = 1101$. Построить производящую матрицу с использованием первого и второго способов.

Решение:

1. Так как, $k = 4$ используем следующие единичные векторы $Q_i(x)$:

$$Q_1 = 0001, \quad Q_2 = 0010, \quad Q_3 = 0100, \quad Q_4 = 1000.$$

Производим следующую операцию над вектором $Q_1(x)$:

$$\frac{Q(x) x^r}{P(x)} = \frac{0001 \cdot 1000}{1101}.$$

В результате деления находим остаток $R_1(x) = 101$. После аналогичных операций с Q_2, Q_3, Q_4 имеем $R_2(x) = 111$, $R_3(x) = 011$, $R_4(x) = 110$.

Таким образом, производящая матрица имеет вид

$$G_{7,4} = \begin{vmatrix} 0001101 \\ 0010111 \\ 0100011 \\ 1000110 \end{vmatrix}.$$

Строки полученной матрицы – первые четыре комбинации кода. Пятая комбинация является нулевой, а остальные одиннадцать образуются суммированием по модулю два всех возможных сочетаний строк производящей матрицы.

2. При втором способе первую строку матрицы получаем из выражения

$P(x) \cdot x^{k-1} = 1101 \cdot 1000$, а последующие $(k-1)$ строки – с помощью циклического сдвига полученной комбинации.

Таким образом, производящая матрица имеет вид

$$G_{7,4} = \begin{vmatrix} 1101000 \\ 0110100 \\ 0011010 \\ 0001101 \end{vmatrix}.$$

Обнаружение ошибок сводится к делению принятой кодовой комбинации на образующий полином. Получение остатка свидетельствует о наличии ошибок. Остаток от деления в циклических кодах играет роль синдрома.

Пример: Дана разрешенная комбинация кода $(7,4)$; $F_7(0,1) = 1000110$, построенная с использованием образующего полинома $P(0,1) = 1101$. Предположим, что ошибка произошла в четвертом разряде. Тогда принятая комбинация имеет

вид $F_7(0,1) = 1001110$. Делим принятую комбинацию на образующий полином:

$$\begin{array}{r}
 \oplus \begin{array}{r} 1001110 \\ \underline{1101} \end{array} \bigg| \begin{array}{r} 1101 \\ \underline{1111} \end{array} \\
 \oplus \begin{array}{r} 1001 \\ \underline{1101} \end{array} \\
 \oplus \begin{array}{r} 1001 \\ \underline{1101} \end{array} \\
 \oplus \begin{array}{r} 1000 \\ \underline{1101} \end{array} \\
 \oplus \begin{array}{r} 1000 \\ \underline{1101} \end{array} \\
 101
 \end{array}$$

Наличие остатка свидетельствует об ошибке.

Для определения места ошибки применим один из способов, основанных на анализе синдрома. Для $d_0 = 3$:

1) сравниваем вес полученного остатка W с числом исправляемых ошибок $t_u = 1$, т. е. $W > t_u$;

2) проводим циклический сдвиг принятой комбинации на один разряд влево и деление на $P(x)$

$$\oplus \begin{array}{r} 0011101 \\ \underline{1101} \end{array} \bigg| \begin{array}{r} 1101 \\ \underline{} \end{array} \\
 111$$

Повторяем сдвиг и деление до тех пор, пока не получим, что

$$W \leq t_u;$$

$$\begin{array}{r}
 \oplus \begin{array}{r} 0111010 \\ \underline{1101} \end{array} \bigg| \begin{array}{r} 1101 \\ \underline{} \end{array} \\
 \oplus \begin{array}{r} 1110 \\ \underline{1101} \end{array} \\
 011
 \end{array}
 \quad
 \begin{array}{r}
 \oplus \begin{array}{r} 1110100 \\ \underline{1101} \end{array} \bigg| \begin{array}{r} 1101 \\ \underline{} \end{array} \\
 \oplus \begin{array}{r} 1110 \\ \underline{1101} \end{array} \\
 110
 \end{array}
 \quad
 \begin{array}{r}
 \oplus \begin{array}{r} 1101001 \\ \underline{1101} \end{array} \bigg| \begin{array}{r} 1101 \\ \underline{} \end{array} \\
 001
 \end{array}$$

Складываем по mod 2 последнее делимое с последним остатком:

$$\oplus \begin{array}{r} 1101001 \\ \underline{} \\ 1 \end{array} \\
 1101000$$

Проводим циклический сдвиг комбинации, полученной в результате суммирования последнего делимого с последним остатком, на четыре разряда вправо (так как перед этим мы четырежды сдвигали принятую комбинацию влево)

$$0110100 \rightarrow 0011010 \rightarrow 0001101 \rightarrow 1000110.$$

Как видно, последняя комбинация соответствует переданной, т. е. не содержит ошибки.

Пример: Используя метод образующих матриц, построить циклический код, исправляющий одиночные ошибки при передаче комбинаций четырехзначного двоичного кода на все сочетания (кроме нулевой комбинации).

Решение:

1. Код, обнаруживающий двоичные или исправляющий одиночные ошибки, должен обеспечить между комбинациями кодовое расстояние $d_0 = 3$, следовательно, число разрядов дополнительной матрицы $n_u = 3$, а каждый остаток содержит три разряда.

2. Выбираем многочлен, степень которого больше или равна трем, число ненулевых членов также должно быть больше или равно трем. Выбранный многочлен – $x^3 + x^2 + 1$.

3. Число строк (столбцов) транспонированной матрицы $n_u = 4$, так как исходный код четырехразрядный.

4. Число единиц в каждом остатке (вес остатка), полученный от деления единицы с нулями на образующий многочлен должно быть $r \geq d_0 - 1$, т. е. $r \geq 2$.

5. Соблюдая условия 1 и 4, находим остатки от деления единицы с

нулями на образующий многочлен:

$\oplus$	1000000	1101		Остатки
	1101			101
	$\oplus$ 1010			111
	1101			011
	$\oplus$ 1110			110
	1101			
	0110			

6. Строим образующую матрицу:

$$G_{7,4} = \begin{array}{c|c} \begin{matrix} 0001101 \\ 0010111 \\ 0100011 \\ 1000110 \end{matrix} & \begin{matrix} x_1 = 0001101 \\ x_2 = 0010111. \\ x_3 = 0100011 \\ x_4 = 1000110 \end{matrix} \end{array}$$

7. Суммированием по mod2 всех возможных сочетаний строк образующей матрицы находим остальные комбинации искомого кода.

Контрольные вопросы и задания

1. С помощью чего достигается согласование статистических свойств источника сообщений и канала связи?
2. К чему сводится передача или хранение информации?
3. На чем основана методика эффективного кодирования Шеннона – Фэнно?
4. С помощью какого приема возможно теоретическое достижение $I_{\text{ср}} = H(z)$ для любого распределения букв первичного алфавита при кодировании методом Шеннона – Фэнно?
5. Для чего служит эффективное кодирование?
6. Что характерно для префиксного кода?
7. Сравните методы Шеннона – Фэнно и Хаффмана.
8. Как оценивается эффективность ОНК?
9. В чем особенность алгоритма LZW?
10. Как представляется при арифметическом кодировании текст?
11. Сформулируйте вторую теорему Шеннона для канала с шумом.
12. Какие коды называются блоковыми?
13. Что называется минимальным кодовым расстоянием?
14. Какие корректирующие коды называются оптимальными.
15. От каких факторов зависит число корректирующих разрядов двоичного помехозащищенного кода?
16. Какой многочлен называется неприводимым?
17. Дайте определение циклическим кодам

Задания

1. Построить оптимальные неравномерные коды по методу Шеннона–Фано и по методу Хаффмена и рассчитать среднюю длину кодового слова. Первоначальный алфавит имеет следующие вероятности появления букв в тексте:

Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8
0,2	0,3	0,07	0,02	0,01	0,15	0,08	0,17

2. Определить минимальное кодовое расстояние d_0 , необходимое для построения корректирующего кода, исправляющего все трехкратные ошибки.

3. Построить порождающую матрицу группового кода для $n_i = 11$. Учесть условия максимальной простоты декодера.

4. Чему равно кодовое расстояние d между комбинацией 10010111 и комбинациями 11111111, 00000000, 00010111 ?

5. Задана следующая матрица группового кода:

$$C = \begin{vmatrix} 1000111 \\ 0100011 \\ 0010110 \\ 0001101 \end{vmatrix}$$

Определить, какие из приведенных ниже кодов, построенных по данной матрице, имеют ошибки: 1101001; 0111111; 0011100; 0010110. Исправить обнаруженные ошибки.

6. Построить оптимальные неравномерные коды по методу Шеннона–Фано и по методу Хаффмена и рассчитать среднюю длину кодового слова. Первоначальный алфавит имеет следующие вероятности появления букв в тексте:

Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8
0,25	0,1	0,07	0,02	0,01	0,3	0,11	0,14

7. Задача. Построить код Хэмминга для информационной комбинации 0101. Показать процесс обнаружения ошибки.

ЗАКЛЮЧЕНИЕ

В связи с бурным развитием техники и систем, основанных на ее эксплуатации все чаще встают вопросы, связанные с активным обменом информацией оптимизацией функционирования на основании получения и обработки информации. Сегодня нас окружает мир, где информационные процессы сопровождают практически каждый проект и его реализацию. Это проявляется прежде всего в том что любая организация, деятельность которой так или иначе связана с хранением и обработкой информации, имеет в своем распоряжении информационную систему. Целью современных информационных систем (связи, телевидения информационно-измерительных систем и т. д.) являются разработка и внедрение современных технологий и устройств обработки, преобразования и передачи информации.

Специалисты в области информационных систем должны владеть новыми технологиями обработки информации, знать основы теории информации и основы теории систем для того чтобы осуществлять грамотное проектирование и эксплуатацию современных информационных систем, имеющих в своем составе, как правило, вычислительную технику.

Развитие теории информационных систем связано прежде всего с развитием технологий обработки данных и достижениями вычислительной техники. Причем технологии обработки данных зависят от скорости и объема поступления информации. Так, на основе вычислительной техники эти технологии, начав развитие от файловых систем и перейдя к технологиям базы данных, сегодня имеют новое направление – Большие данные.

При разработке современных информационных систем требуется знать основы теории систем, теории информации и учитывать прогрессивные информационные процессы и технологии, связанные со сбором, хранением, передачей, поиском и обработкой информации.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Юркевич Е. В. Введение в теорию информационных систем. – М.: ИДТ, 2007. – 272.
2. Блинков Ю. В. Основы теории информационных процессов и систем: Учеб. пособие для обучающихся по специальности «Информационные системы и технологии».
3. Вернер М. Основы кодирования. – М.: Техносфера, 2004. – 288 с.
4. Крайников А. В. Вероятностные методы в вычислительной технике: Учеб. пособие для вузов / А. В. Крайников и др. – М.: Высшая школа, 1986. – 312 с.
5. Горелов Г. В. Теория передачи сигналов на железнодорожном транспорте: Учеб. пособие / Горелов Г. В., Фомин А. Ф. и др. – М.: Транспорт, 1999. – 415 с.
6. Теория вероятностей с элементами математической статистики: Учеб. пособие для вузов / Е. И. Гурский – М.: Высшая школа, 1971. – 328 с.
7. Прикладная теория информации: Учеб. пособие / В. И. Дмитриев. – М.: Высшая школа, 1989. – 328 с.
8. Душин В. К. Теоретические основы информационных процессов и систем: Учеб. пособие. – М: Дашков и К, 2001. – 348 с.
9. Зюко А. Г. Помехоустойчивость и эффективность систем связи. – М.: Связь, 1972. – 359 с.
10. Айтчанов Б. Х. Теория информации: Учеб. пособие / Б. Х. Айтчанов, Т. В. Яскевич – Алматы: КазНТУ, 2009. – 139 с.
11. Петров В. Н. Информационные системы: Учеб. пособие / В. Н. Петров. – СПб: Питер, 2002, 688 с.
12. Коган И. М. Прикладная теория информации: Учеб. пособие / И. М. Коган. – М.: Радио и связь, 1981. – 216 с.
13. Кузьмин И. В. Основы теории информации и кодирования: Учеб. пособие / И. В. Кузьмин, В. А. Кедрус. – Киев: Вища школа, 1977. – 280 с.
14. Лебедев Д. С. Учеб. пособие по курсу «Основы теории информации». – М.: Изд.: МГУ, 1966. – 160 с.

15. *Иванов И. В.* Теория информационных процессов и систем: Учеб. пособие / *И. В. Иванов.* – Белгород: Изд-во БГТУ, 2007. – 156 с.
16. *Логинов В. М.* Экономичное кодирование: Учеб. пособие / *В. М. Логинов и др.* – Киев: Техника, 1976. – 174 с.
17. *Новик А. А.* Эффективное кодирование: Учеб. пособие / *Новик А. А.* – М.: Энергия, 1965. – 236 с.
18. *Чернавский Д. С.* Синергетика и информация. Динамическая теория информации: Учеб. пособие / *Д. С. Чернавский.* – М.: URSS, 2006.
19. *Хлебников А. А.* Информационные технологии. – М.: КНОРУС, 2015.
20. *Питерсон У.* Коды, исправляющие ошибки: Учеб. пособие / *У. Питерсон, Э. М. Уэлдон.* – М.: Мир, 1976. – 594 с.
21. *Гордеев А. В.* Системное программное обеспечение: Учеб. пособие / *А. В. Гордеев, А. Ю. Молчанов.* – СПб: Питер, 2001. – 736 с.
22. *Самойленко Ф. И.* Помехозащищенное кодирование: Учеб. пособие / *Ф. И. Самойленко.* – М.: Наука, 1966. – 238 с.
23. *Сергиенко А. Б.* Цифровая обработка сигналов: Учеб. пособие / *А. Б. Сергиенко.* – СПб: Питер, 2003. – 608 с.
24. *Советов Б. Я.* Теория информации: Учеб. пособие / *Б. Я. Советов.* – Л.: Изд. ЛГУ, 1985. – 172 с.
25. *Советов Б. Я.* Информационные технологии: Учебник для вузов по специальности «Автоматизированные системы обработки информации и управления» / *Б. Я. Советов.* – М.: Высшая школа, 1994. – 368 с.
26. *Тарасенко Ф. П.* Введение в курс теории информации: Учеб. пособие / *Ф. П. Тарасенко.* – Томск: Изд-во ТГУ, 1963. – 240 с.
27. *Темников Ф. Е.* Теоретические основы информационной техники: Учеб. пособие / *Ф. Е. Темников, В. А. Афонин, В. И. Дмитриев.* – М.: Энергия, 1979. – 512 с.
28. *Харкевич А. А.* Борьба с помехами: Учеб. пособие / *А. А. Харкевич.* – М.: Физматгиз, 1965. – 276 с.
29. *Хартли Р.* Передача информации. Теория информации и ее приложения / Под ред. *А. А. Харкевича.* – М.: Физматгиз, 1959. – 350 с.

30. Хетагуров Я. А. Повышение надежности цифровых устройств методом избыточного кодирования: Учеб. пособие / Я. А. Хетагуров, Ю. П. Руднев. – М.: Энергия, 1974. – 271 с.

31. Хэмминг Р. В. Теория кодирования и теория информации: Учеб. пособие / Р. В. Хэмминг. – М.: Радио и связь, 1983. – 174 с.

32. Шеннон К. Работы по теории информации и кибернетике: Учеб. пособие. – М.: Издательство иностранной литературы, 1963. – 820 с.

Дополнительная литература

33. Бурков В. Н. Основы математической теории активных систем. – М.: Наука, 1977. – 255 с.

34. Блейхут Р. Теория и практика кодов, исправляющих ошибки. – М.: Мир, 1986.

35. Каменский С. В. Автоматические системы управления. Исследование некоторых алгоритмов кодирования и декодирования кодов: Учеб. пособие / С. В. Каменский, К. И. Меньшиков, 2006.

36. Хомоненко А. Д., Цыганков В. М., Мальцев М. Г. Базы данных. – 6-е изд. Изд.: КОРОНА–Век, 2009. – 734 с.]

37. Мак-Вильямс Ф. Дж. А., Слоэн Н. Дж. А. Теория кодов, исправляющих ошибки. – М.: Радио и связь, 1979.

38. Морелос-Сарагаса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение: Учеб. пособие / Морелос-Сарагаса Р. – М.: Техносфера, 2005.

39. Охорзин В. М. Построение каскадных кодов на основе кодов Боуза-Чоудхури-Хоквингема и Рида-Соломона: Учеб. пособие / В. М. Охорзин, Д. С. Кукунин, М. С. Новодворский. – М., 2006.

40. Соловьева Ф. И. Введение в теорию кодирования: Учеб. пособие. – Новосибирск: Гос.ун-т, 2006.

41. <http://www.HardwarePortal.ru>

42. <http://www.tadviser.ru/index.php/Статья>: Большие_данные_(Big_Data).

ГЛОССАРИЙ

Агрегирование – понятие, противоположное декомпозиции.

Архивация – устранение из сжимаемого текста избыточности.

АЦП – аналого-цифровой преобразователь – преобразует аналоговый сигнал в цифровой.

Бит – единица измерения информации (статистическая мера) при основании логарифма

Большие данные – совокупность подходов, инструментов и методов обработки структурированных и неструктурированных данных огромных объёмов для получения воспринимаемых человеком результатов

Вес кодовой комбинации – число ненулевых символов в ней.

Внешняя среда – множество элементов, которые не входят в систему, но изменение их состояния вызывает изменение поведения системы.

Входы и выходы – материальные или информационные потоки входящие и выходящие из системы

Дит – единица измерения информации (статистическая мера) при основании 10.

Диагностирование – определение состояния X по известным входу U и выходу Y .

Значность – число символов в кодовой комбинации.

Информация – совокупность сведений каких-либо событиях, явлениях, предметах.

Информационные системы – системы передачи, хранения и обработки информации.

Информационная среда – это среда (т. е. система и ее окружение) из взаимодействующих информационных систем, включая и информацию, актуализируемую в этих системах.

Информационная система управления – система, предназначенная для управления как другой системой, так и внутри системы (т. е. в качестве управляющей подсистемы).

Информационные разряды – n_i разряды, с помощью которых передается информация ($n_i = \log N$).

Интегративность – наличие единых целей,

Идентификация – описание связей между входами, выходами и состояниями (нахождение функции наблюдения η , переходного отображения σ).

Коэффициент статистического сжатия – отношение максимальной энтропии к средней длине кода (характеризует уменьшение количества двоичных знаков на символ сообщения при применении ОНК по сравнению с применением методов нестатистического кодирования).

Коэффициент относительной эффективности – отношение энергии к средней длине кода (показывает) насколько используется избыточность передаваемого сообщения.

Кодирование в широком смысле – представление сообщений в форме, удобной для передачи по данному каналу.

Кодирование в узком смысле – преобразование исходного алфавита в другой, с меньшим числом знаков.

Линия связи – любая физическая среда (воздух, металл и т.д.), обеспечивающая поступление сигналов от передающего устройства к приемнику.

Минимальное кодовое расстояние – минимальное расстояние, взятое по всем параметрам кодовых комбинаций кода.

Модель – описание системы, отображающее определенную группу ее свойств.

Модель функционирования (поведения) системы – модель, предсказывающая изменение состояния системы во времени.

Нат – единица информации (статистическая мера) при основании логарифма = e .

Прогнозирование – предсказание выхода Y по входу U .

Подсистема – часть системы с некоторыми связями и отношениями

Поведение – способность переходить из одного состояния в другое (например, $x_1 \rightarrow x_2 \rightarrow x_3$).

Полная информационная избыточность выражается через составляющие избыточности, связанные с: а) неравномерным распределением символов в сообщении; б) статистической связью между символами сообщения.

Пропускная способность канала – максимальная скорость передачи информации по данному каналу связи.

Прогнозирование – предсказание выхода Y по входу U .

Равновесие – это способность системы в отсутствие внешних возмущающих воздействий (или при постоянных воздействиях) сохранить свое состояние сколь угодно долго.

Распознавание – определение U по известному выходу Y .

Структурная мера информации – использует простой подсчет информационных элементов.

Статистическая мера информации – оперирует понятием энтропия.

Семантическая мера учитывает целесообразность, ценность полученной информации.

Синдром – проверочный вектор.

Стационарность процесса – его существование и статистическая однородность во всем диапазоне от $-\infty$ до $+\infty$.

Сигнал – физический процесс, отображающий передаваемое сообщение.

Символ – при большом объеме алфавита часто прибегает к представлению букв в другом алфавите с меньшим числом букв, которые называются символами.

Структура – это совокупность элементов и связей между ними.

Связь – обеспечивает возникновение и сохранение структуры и целостных свойств системы

Состояние – мгновенный «срез» системы, остановку в ее развитии.

Структурированность – возможность расчленения системы на составляющие компоненты

Система управления файлами – комплекс программных модулей, обеспечивающих работу с файлами в конкретной операционной системе

Управление – определение U для известного значения X .

Устойчивость – способность системы возвращаться в состояние равновесия после того, как она была из этого состояния выведена под влиянием внешних возмущающих воздействий

Файловая система – определяет принципы доступа к данным, организованных в файлы.

Цель – образ несуществующего, но желаемого, с точки зрения задачи или рассматриваемой проблемы, состояния среды, т. е. такого состояния, которое позволяет решать проблему при данных ресурсах.

Эмерджентность – цели (функции) компонентов системы не всегда совпадают с целями (функциями) системы;

Элемент – простейшая неделимая часть системы.

Файловая система – определяет принципы доступа к данным, организованных в файлы.

Эпсилон-энтропия – минимальное количество информации в одной случайной величине Z относительно другой V , при котором удовлетворяем заданное требование к верности воспроизведения величины V .

Приложение 1.

Двоичные логарифмы чисел от 1 до 128

p	$-\log_2 p$	$-p \log_2 p$	n	$-q \log_2 q$	$-\log_2 q$	q
0,0005	10,965784	0,005483	0,006204	0,000721	0,000722	0,9995
0,0010	9,965784	0,009966	0,011408	0,001442	0,001443	0,9990
0,0015	9,380882	0,014071	0,016234	0,002162	0,002166	0,9985
0,0020	8,965784	0,017932	0,020814	0,002882	0,002888	0,9980
0,0025	8,643956	0,021610	0,025212	0,003602	0,003611	0,9975
0,0030	8,380882	0,025142	0,029464	0,004322	0,004335	0,9970
0,0035	8,158429	0,028555	0,033595	0,005041	0,005058	0,9965
0,0040	7,965784	0,031863	0,037622	0,005759	0,005782	0,9960
0,0045	7,795859	0,035081	0,041559	0,006477	0,006507	0,9955
0,0050	7,643856	0,038219	0,045415	0,007195	0,007232	0,9950
0,0055	7,506353	0,041285	0,049198	0,007913	0,007957	0,9945
0,0060	7,380822	0,044285	0,052915	0,008630	0,008682	0,9940
0,0065	7,265345	0,047225	0,056572	0,009347	0,009408	0,9935
0,0070	7,158429	0,050109	0,060172	0,010063	0,010134	0,9930
0,0075	7,058894	0,052942	0,063721	0,010780	0,010861	0,9925
0,0080	6,955784	0,055726	0,067222	0,011495	0,011588	0,9920
0,0085	6,878321	0,058466	0,070676	0,012211	0,012315	0,9915
0,0090	6,795859	0,061163	0,074088	0,012926	0,013043	0,9910
0,0095	6,717857	0,063820	0,077460	0,013640	0,013771	0,9905
0,0100	6,643856	0,066439	0,080793	0,014355	0,014500	0,9900
0,0110	6,506353	0,071570	0,087352	0,015782	0,015958	0,9890
0,0120	6,380822	0,076570	0,093778	0,017208	0,017417	0,9880
0,0130	6,265345	0,081449	0,100082	0,018633	0,018878	0,9870
0,0140	6,158429	0,086218	0,106274	0,020056	0,020340	0,9860
0,0150	6,058894	0,090883	0,112361	0,021477	0,021804	0,9850
0,0160	5,965784	0,095453	0,118350	0,022897	0,023270	0,9840
0,0170	5,878321	0,099931	0,12248	0,024316	0,024737	0,9830
0,0180	5,795859	0,104325	0,130059	0,025733	0,026205	0,9820
0,0190	5,717857	0,108639	0,135788	0,027149	0,027675	0,9810
0,0200	5,643856	0,112877	0,141441	0,028563	0,029146	0,9800
0,0210	5,573467	0,117043	0,147019	0,029976	0,030619	0,9790
0,0220	5,506353	0,121140	0,152527	0,031388	0,032094	0,9780
0,0230	5,442222	0,125171	0,157969	0,032797	0,033570	0,9770
0,0240	5,380822	0,129140	0,163346	0,034206	0,035047	0,9760
0,0250	5,321928	0,133048	0,168661	0,035513	0,036526	0,9750
0,0260	5,265345	0,136899	0,173917	0,037018	0,038006	0,9740
0,0270	5,210897	0,140694	0,179116	0,038422	0,039488	0,9730
0,0280	5,158428	0,144436	0,184216	0,040972	0,040972	0,9720
0,0290	5,107803	0,148126	0,189352	0,041226	0,042457	0,9710
0,0300	5,058894	0,151767	0,194392	0,042625	0,043943	0,9700
0,0310	5,011588	0,155359	0,199382	0,044023	0,045431	0,9690
0,0320	4,965784	0,158905	0,204325	0,045420	0,046921	0,9680
0,0330	4,921390	0,162406	0,209220	0,046815	0,048412	0,9670
0,0340	4,878321	0,165863	0,214071	0,048208	0,049904	0,9660
0,0350	4,836501	0,169278	0,218878	0,049600	0,051399	0,9650
0,0360	4,795859	0,172651	0,223642	0,050991	0,052895	0,9640
0,0370	4,756331	0,175984	0,228364	0,052380	0,054392	0,9630
0,0380	4,717857	0,179279	0,233046	0,053767	0,056891	0,9620

Продолжение приложения 1.

0,0390	4,680382	0,182535	0,237688	0,055153	0,057392	0,9610
0,0400	4,643856	0,185754	0,242292	0,056553	0,058894	0,9600
0,0410	4,608232	0,188938	0,246858	0,057921	0,060397	0,9590
0,0420	4,573467	0,192086	0,251388	0,059303	0,061902	0,9580
0,0430	4,539519	0,195199	0,255882	0,060683	0,063409	0,9570
0,0440	4,506353	0,198280	0,260341	0,062061	0,064917	0,9560
0,0450	4,473931	0,201327	0,264765	0,063438	0,066427	0,9550
0,0460	4,442222	0,204342	0,269156	0,064814	0,067939	0,9540
0,0470	4,411195	0,207326	0,273514	0,066188	0,069452	0,9530
0,0480	4,380822	0,210279	0,277840	0,067560	0,070967	0,9520
0,0490	4,351074	0,213203	0,282134	0,068931	0,072483	0,9610
0,0500	4,321928	0,216096	0,286397	0,070301	0,074001	0,9500
0,0520	4,265345	0,221798	0,294833	0,073035	0,077041	0,9480
0,0530	4,237864	0,224607	0,299007	0,074400	0,078564	0,9470
0,0540	4,210897	0,227388	0,303152	0,075763	0,079078	0,9460
0,0550	4,184425	0,230143	0,307268	0,077125	0,081614	0,9450
0,0560	4,158429	0,232872	0,311357	0,078485	0,083141	0,9440
0,0570	4,132894	0,235575	0,315419	0,079844	0,084670	0,9430
0,0580	4,107803	0,238253	0,319454	0,081201	0,086201	0,9420
0,0590	4,083141	0,240905	0,323462	0,082557	0,087733	0,9410
0,0600	4,058894	0,243534	0,327445	0,083911	0,089267	0,9400
0,0625	4,00000	0,250000	0,337290	0,087290	0,093109	0,9375
0,0650	3,944316	0,256322	0,346981	0,090659	0,096962	0,9350
0,0675	3,888969	0,262505	0,356524	0,094019	0,100824	0,9325
0,0700	3,936501	0,268555	0,365924	0,097369	0,104698	0,9300
0,0725	3,785875	0,274476	0,375185	0,100709	0,108581	0,9275
0,0750	3,736966	0,280272	0,384312	0,104039	0,112475	0,9250
0,0775	3,689660	0,285949	0,393308	0,107360	0,116379	0,9225
0,0800	3,643856	0,291508	0,402179	0,110671	0,120294	0,9200
0,0825	3,599462	0,296956	0,410227	0,113972	0,124220	0,9175
0,0850	3,556393	0,302293	0,419556	0,117263	0,128156	0,9150
0,0875	3,514573	0,307525	0,428070	0,120544	0,132104	0,9125
0,0900	3,473931	0,312654	0,436470	0,123816	0,136062	0,9100
0,0925	3,434402	0,317682	0,444760	0,127078	0,140030	0,9075
0,0950	3,395929	0,322613	0,452943	0,130329	0,144010	0,9050
0,0975	3,358454	0,327449	0,461020	0,133571	0,148001	0,9025
0,1000	3,321928	0,332193	0,468986	0,136803	0,152003	0,9000
0,1025	3,286304	0,336846	0,476871	0,140024	0,156016	0,8975
0,1050	3,251539	0,341412	0,484648	0,143236	0,160040	0,8950
0,1075	3,217591	0,345891	0,492329	0,146438	0,164076	0,8925
0,1100	3,184425	0,350287	0,499916	0,149629	0,168123	0,8900
0,1125	3,152003	0,354600	0,507411	0,152811	0,172181	0,8875
0,1150	3,120294	0,358834	0,514816	0,155982	0,176251	0,8850
0,1175	3,089267	0,362989	0,522132	0,159143	0,180332	0,8825
0,1200	3,058894	0,367067	0,529361	0,162294	0,184425	0,8800
0,1225	3,029146	0,371070	0,536505	0,165434	0,188529	0,8775

Продолжение приложения 1.

0,1250	3,000000	0,375000	0,543564	0,168564	0,192645	0,8750
0,1275	2,911431	0,378857	0,550542	0,171684	0,196773	0,8725
0,1300	2,943416	0,382644	0,557438	0,174794	0,200913	0,8700
0,1325	2,915936	0,386361	0,564255	0,177893	0,205064	0,8675
0,1350	2,888969	0,390011	0,570993	0,180982	0,209228	0,8650
0,1375	2,862496	0,393593	0,577654	0,184061	0,213404	0,8625
0,1400	2,836501	0,397110	0,584239	0,187129	0,217591	0,8600
0,1425	2,810966	0,400563	0,590749	0,190186	0,221791	0,8575
0,1450	2,785875	0,403952	0,597185	0,193233	0,226004	0,8550
0,1475	2,761213	0,407279	0,603549	0,196270	0,230228	0,8525
0,1500	2,736966	0,410545	0,609840	0,199295	0,234465	0,8500
0,1525	2,713119	0,413751	0,616061	0,203311	0,238715	0,8475
0,1550	2,689660	0,416897	0,622213	0,205315	0,142977	0,8450
0,1575	2,666576	0,419986	0,628295	0,208309	0,247251	0,8425
0,1600	2,643856	0,423017	0,634310	0,211293	0,251539	0,8400
0,1625	2,621488	0,425992	0,640257	0,214265	0,255839	0,8375
0,1650	2,599462	0,428911	0,646138	0,217227	0,260152	0,8350
0,1675	2,577767	0,431776	0,651954	0,220178	0,264678	0,8325
0,1700	2,556393	0,434587	0,657705	0,223118	0,268817	0,8300
0,1725	2,535332	0,437345	0,663392	0,226047	0,273169	0,8275
0,1750	2,514573	0,440050	0,669016	0,228966	0,277534	0,8250
0,1775	2,494109	0,442704	0,674577	0,231873	0,281912	0,8225
0,1800	2,473931	0,445308	0,680077	0,234769	0,286304	0,8200
0,1825	2,454032	0,337861	0,685516	0,237655	0,290709	0,8175
0,1850	2,434403	0,450365	0,690894	0,240529	0,295128	0,8150
0,1875	2,415037	0,452820	0,696212	0,243393	0,299560	0,8125
0,1900	2,395929	0,455226	0,701471	0,246245	0,304006	0,8100
0,1925	2,377070	0,457586	0,706672	0,249086	0,308466	0,8075
0,1950	2,358454	0,459899	0,711815	0,251916	0,312939	0,8050
0,1975	2,340075	0,462165	0,716900	0,254735	0,317427	0,8025
0,2000	2,321928	0,464386	0,721928	0,257542	0,321928	0,8000
0,2050	2,286304	0,468692	0,731816	0,262124	0,330973	0,7250
0,2100	2,251539	0,472883	0,741483	0,268660	0,340075	0,7900
0,2150	2,217591	0,476782	0,750932	0,274151	0,349235	0,7850
0,2200	2,184425	0,480573	0,760167	0,279594	0,358454	0,7800
0,2250	2,152003	0,484201	0,769193	0,284992	0,367732	0,7750
0,2300	2,120294	0,487668	0,778011	0,290344	0,377070	0,7700

Приложение 2.

Энтропия Н при вероятностях р и q двоичного события

p	$-\log_2 p$	$-p \log_2 p$	H	$-q \log_2 q$	$-q \log_2 q$	q
0,0005	10,965784	0,005483	0,006204	0,000721	0,000722	0,9995
0,0010	9,965784	0,009966	0,011408	0,001442	0,001443	0,9990
0,0015	9,380882	0,014071	0,016234	0,002162	0,002166	0,9985
0,0020	8,965784	0,017932	0,020814	0,002882	0,002888	0,9980
0,0025	8,643956	0,021610	0,025212	0,003602	0,003611	0,9975
0,0030	8,380882	0,025142	0,029464	0,004322	0,004335	0,9970
0,0035	8,158429	0,028555	0,033595	0,005041	0,005058	0,9965
0,0040	7,965784	0,031863	0,037622	0,005759	0,005782	0,9960
0,0045	7,795859	0,035081	0,041559	0,006477	0,006507	0,9955
0,0050	7,643856	0,038219	0,045415	0,007195	0,007232	0,9950
0,0055	7,506353	0,041285	0,049198	0,007913	0,007957	0,9945
0,0060	7,380822	0,044285	0,052915	0,008630	0,008682	0,9940
0,0065	7,265345	0,047225	0,056572	0,009347	0,009408	0,9935
0,0070	7,158429	0,050109	0,060172	0,010063	0,010134	0,9930
0,0075	7,058894	0,052942	0,063721	0,010780	0,010861	0,9925
0,0080	6,955784	0,055726	0,067222	0,011495	0,011588	0,9920
0,0085	6,878321	0,058466	0,070676	0,012211	0,012315	0,9915
0,0090	6,795859	0,061163	0,074088	0,012926	0,013043	0,9910
0,0095	6,717857	0,063820	0,077460	0,013640	0,013771	0,9905
0,0100	6,643856	0,066439	0,080793	0,014355	0,014500	0,9900
0,0110	6,506353	0,071570	0,087352	0,015782	0,015958	0,9890
0,0120	6,380822	0,076570	0,093778	0,017208	0,017417	0,9880
0,0130	6,265345	0,081449	0,100082	0,018633	0,018878	0,9870
0,0140	6,158429	0,086218	0,106274	0,020056	0,020340	0,9860
0,0150	6,058894	0,090883	0,112361	0,021477	0,021804	0,9850
0,0160	5,965784	0,095453	0,118350	0,022897	0,023270	0,9840
0,0170	5,878321	0,099931	0,12248	0,024316	0,024737	0,9830
0,0180	5,795859	0,104325	0,130059	0,025733	0,026205	0,9820
0,0190	5,717857	0,108639	0,135788	0,027149	0,027675	0,9810
0,0200	5,643856	0,112877	0,141441	0,028563	0,029146	0,9800
0,0210	5,573467	0,117043	0,147019	0,029976	0,030619	0,9790
0,0220	5,506353	0,121140	0,152527	0,031388	0,032094	0,9780
0,0230	5,442222	0,125171	0,157969	0,032797	0,033570	0,9770
0,0240	5,380822	0,129140	0,163346	0,034206	0,035047	0,9760
0,0250	5,321928	0,133048	0,168661	0,035513	0,036526	0,9750
0,0260	5,265345	0,136899	0,173917	0,037018	0,038006	0,9740
0,0270	5,210897	0,140694	0,179116	0,038422	0,039488	0,9730
0,0280	5,158428	0,144436	0,184216	0,040972	0,040972	0,9720
0,0290	5,107803	0,148126	0,189352	0,041226	0,042457	0,9710
0,0300	5,058894	0,151767	0,194392	0,042625	0,043943	0,9700
0,0310	5,011588	0,155359	0,199382	0,044023	0,045431	0,9690
0,0320	4,965784	0,158905	0,204325	0,045420	0,046921	0,9680
0,0330	4,921390	0,162406	0,209220	0,046815	0,048412	0,9670
0,0340	4,878321	0,165863	0,214071	0,048208	0,049904	0,9660
0,0350	4,836501	0,169278	0,218878	0,049600	0,051399	0,9650
0,0360	4,795859	0,172651	0,223642	0,050991	0,052895	0,9640
0,0370	4,756331	0,175984	0,228364	0,052380	0,054392	0,9630
0,0380	4,717857	0,179279	0,233046	0,053767	0,055891	0,9620

Продолжение приложения 2.

0,0390	4,680382	0,182535	0,237688	0,055153	0,057392	0,9610
0,0400	4,643856	0,185754	0,242292	0,056553	0,058894	0,9600
0,0410	4,608232	0,188938	0,246858	0,057921	0,060397	0,9590
0,0420	4,573467	0,192086	0,251388	0,059303	0,061902	0,9580
0,0430	4,539519	0,195199	0,255882	0,060683	0,063409	0,9570
0,0440	4,506353	0,198280	0,260341	0,062061	0,064917	0,9560
0,0450	4,473931	0,201327	0,264765	0,063438	0,066427	0,9550
0,0460	4,442222	0,204342	0,269156	0,064814	0,067939	0,9540
0,0470	4,411195	0,207326	0,273514	0,066188	0,069452	0,9530
0,0480	4,380822	0,210279	0,277840	0,067560	0,070967	0,9520
0,0490	4,351074	0,213203	0,282134	0,068931	0,072483	0,9610
0,0500	4,321928	0,216096	0,286397	0,070301	0,074001	0,9500
0,0520	4,265345	0,221798	0,294833	0,073035	0,077041	0,9480
0,0530	4,237864	0,224607	0,299007	0,074400	0,078564	0,9470
0,0540	4,210897	0,227388	0,303152	0,075763	0,079078	0,9460
0,0550	4,184425	0,230143	0,307268	0,077125	0,081614	0,9450
0,0560	4,158429	0,232872	0,311357	0,078485	0,083141	0,9440
0,0570	4,132894	0,235575	0,315419	0,079844	0,084670	0,9430
0,0580	4,107803	0,238253	0,319454	0,081201	0,086201	0,9420
0,0590	4,083141	0,240905	0,323462	0,082557	0,087733	0,9410
0,0600	4,058894	0,243534	0,327445	0,083911	0,089267	0,9400
0,0625	4,00000	0,250000	0,337290	0,087290	0,093109	0,9375
0,0650	3,943416	0,256322	0,346981	0,090659	0,096962	0,9350
0,0675	3,888969	0,262505	0,356524	0,094019	0,100824	0,9325
0,0700	3,936501	0,268555	0,365924	0,097369	0,104698	0,9300
0,0725	3,785875	0,274476	0,375185	0,100709	0,108581	0,9275
0,0750	3,736966	0,280272	0,384312	0,104039	0,112475	0,9250
0,0775	3,689660	0,285949	0,393308	0,107360	0,116379	0,9225
0,0800	3,643856	0,291508	0,402179	0,110671	0,120294	0,9200
0,0825	3,599462	0,296956	0,410227	0,113972	0,124220	0,9175
0,0850	3,556393	0,302293	0,419556	0,117263	0,128156	0,9150
0,0875	3,514573	0,307525	0,428070	0,120544	0,132104	0,9125
0,0900	3,473931	0,312654	0,436470	0,123816	0,136062	0,9100
0,0925	3,434402	0,317682	0,444760	0,127078	0,140030	0,9075
0,0950	3,395929	0,322613	0,452943	0,130329	0,144010	0,9050
0,0975	3,358454	0,327449	0,461020	0,133571	0,148001	0,9025
0,1000	3,321928	0,332193	0,468986	0,136803	0,152003	0,9000
0,1025	3,286304	0,336846	0,476871	0,140024	0,156016	0,8975
0,1050	3,251539	0,341412	0,484648	0,143236	0,160040	0,8950
0,1075	3,217591	0,345891	0,492329	0,146438	0,164076	0,8925
0,1100	3,184425	0,350287	0,499916	0,149629	0,168123	0,8900
0,1125	3,152003	0,354600	0,507411	0,152811	0,172181	0,8875
0,1150	3,120294	0,358834	0,514816	0,155982	0,176251	0,8850
0,1175	3,089267	0,362989	0,522132	0,159143	0,180332	0,8825
0,1200	3,058894	0,367067	0,529361	0,162294	0,184425	0,8800
0,1225	3,029146	0,371070	0,536505	0,165434	0,188529	0,8775
0,1250	3,000000	0,375000	0,543564	0,168564	0,192645	0,8750
0,1275	2,911431	0,378857	0,550542	0,171684	0,196773	0,8725
0,1300	2,943416	0,382644	0,557438	0,174794	0,200913	0,8700
0,1325	2,915936	0,386361	0,564255	0,177893	0,205064	0,8675

Продолжение приложения 2.

0,1350	2,888969	0,390011	0,570993	0,180982	0,209228	0,8650
0,1375	2,862496	0,393593	0,577654	0,184061	0,213404	0,8625
0,1400	2,836501	0,397110	0,584239	0,187129	0,217591	0,8600
0,1425	2,810966	0,400563	0,590749	0,190186	0,221791	0,8575
0,1450	2,785875	0,403952	0,597185	0,193233	0,226004	0,8550
0,1475	2,761213	0,407279	0,603549	0,196270	0,230228	0,8525
0,1500	2,736966	0,410545	0,609840	0,199295	0,234465	0,8500
0,1525	2,713119	0,413751	0,616061	0,203311	0,238715	0,8475
0,1550	2,689660	0,416897	0,622213	0,205315	0,142977	0,8450
0,1575	2,666576	0,419986	0,628295	0,208309	0,247251	0,8425
0,1600	2,643856	0,423017	0,634310	0,211293	0,251539	0,8400
0,1625	2,621488	0,425992	0,640257	0,214265	0,255839	0,8375
0,1650	2,599462	0,428911	0,646138	0,217227	0,260152	0,8350
0,1675	2,577767	0,431776	0,651954	0,220178	0,264678	0,8325
0,1700	2,556393	0,434587	0,657705	0,223118	0,268817	0,8300
0,1725	2,535332	0,437345	0,663392	0,226047	0,273169	0,8275
0,1750	2,514573	0,440050	0,669016	0,228966	0,277534	0,8250
0,1775	2,494109	0,442704	0,674577	0,231873	0,281912	0,8225
0,1800	2,473931	0,445308	0,680077	0,234769	0,286304	0,8200
0,1825	2,454032	0,447861	0,685516	0,237655	0,290709	0,8175
0,1850	2,434403	0,450365	0,690894	0,240529	0,295128	0,8150
0,1875	2,415037	0,452820	0,696212	0,243393	0,299560	0,8125
0,1900	2,395929	0,455226	0,701471	0,246245	0,304006	0,8100
0,1925	2,377070	0,457586	0,706672	0,249086	0,308466	0,8075
0,1950	2,358454	0,459899	0,711815	0,251916	0,312939	0,8050
0,1975	2,340075	0,462165	0,716900	0,254735	0,317427	0,8025
0,2000	2,321928	0,464386	0,721928	0,257542	0,321928	0,8000
0,2050	2,286304	0,468692	0,731816	0,262124	0,330973	0,7250
0,2100	2,251539	0,472883	0,741483	0,268660	0,340075	0,7900
0,2150	2,217591	0,476782	0,750932	0,274151	0,349235	0,7850
0,2200	2,184425	0,480573	0,760167	0,279594	0,358454	0,7800
0,2250	2,152003	0,484201	0,769193	0,284992	0,367732	0,7750
0,2300	2,120294	0,487668	0,778011	0,290344	0,377070	0,7700
0,2350	2,089267	0,490978	0,786626	0,295648	0,386468	0,7650
0,2400	2,058894	0,494134	0,795040	0,300906	0,395929	0,7600
0,2450	2,029146	0,497141	0,803257	0,306116	0,405451	0,7550
0,2500	2,000000	0,500000	0,811278	0,311278	0,415037	0,7500
0,2550	1,971431	0,502715	0,819107	0,316392	0,424688	0,7450
0,2600	1,943416	0,505288	0,826746	0,321458	0,434403	0,7400
0,2650	1,915936	0,507723	0,834198	0,326475	0,444184	0,7350
0,2700	1,888969	0,510022	0,841465	0,331443	0,454032	0,7300
0,2750	1,862496	0,512187	0,848548	0,336362	0,463947	0,7250
0,2800	1,836501	0,514220	0,855451	0,341230	0,473931	0,7200
0,2850	1,810966	0,516125	0,862175	0,346049	0,483985	0,7150
0,2900	1,785875	0,517904	0,868721	0,350817	0,494109	0,7100
0,2950	1,761213	0,519558	0,875093	0,355535	0,504305	0,7050
0,3000	1,736966	0,521090	0,881291	0,360201	0,514573	0,7000
0,3050	1,713119	0,522501	0,887317	0,364816	0,524915	0,6950

Продолжение приложения 2.

0,3100	1,689660	0,523795	0,893173	0,369379	0,535332	0,6900
0,3150	1,665576	0,524972	0,898861	0,373890	0,545824	0,6850
0,3200	1,643856	0,526034	0,904381	0,378347	0,556393	0,6800
0,3250	1,621488	0,526984	0,909736	0,382752	0,567041	0,6750
0,3300	1,599462	0,527822	0,914926	0,387104	0,577767	0,6700
0,3350	1,577767	0,528552	0,919953	0,391402	0,588574	0,6650
0,3400	1,556393	0,529174	0,924819	0,395645	0,599462	0,6600
0,3450	1,535332	0,529689	0,929523	0,399834	0,610043	0,6550
0,3500	1,514573	0,530101	0,934068	0,403967	0,621488	0,6500
0,3550	1,494109	0,530409	0,938454	0,408046	0,632629	0,6450
0,3600	1,473931	0,530615	0,942683	0,412068	0,643856	0,6400
0,3650	1,454032	0,530722	0,946755	0,416034	0,655157	0,6350
0,3700	1,434403	0,530729	0,950672	0,419943	0,666576	0,6300
0,3750	1,415037	0,530639	0,954434	0,423795	0,678072	0,6250
0,3800	1,395929	0,530453	0,958042	0,427589	0,689660	0,6200
0,3850	1,377070	0,530172	0,961497	0,431323	0,701342	0,6150
0,3900	1,358454	0,529797	0,965800	0,435002	0,713119	0,6100
0,3950	1,340075	0,529330	0,967951	0,438621	0,724993	0,6050
0,4000	1,321928	0,528771	0,970951	0,442179	0,736966	0,6000
0,4050	1,304006	0,528122	0,973800	0,445678	0,749038	0,5950
0,4100	1,286304	0,527385	0,976500	0,449116	0,761213	0,5900
0,4150	1,268817	0,526559	0,979051	0,452493	0,773491	0,5850
0,4200	1,251539	0,525645	0,981454	0,455808	0,785875	0,5800
0,4250	1,234464	0,524648	0,983708	0,459061	0,798366	0,5750
0,4300	1,217591	0,523564	0,985815	0,462251	0,810966	0,5700
0,4350	1,200913	0,522397	0,987775	0,465378	0,823677	0,5650
0,4400	1,184425	0,521147	0,989588	0,468441	0,836501	0,5600
0,4450	1,168123	0,519815	0,991254	0,471439	0,849440	0,5550
0,4500	1,152003	0,518401	0,992774	0,474373	0,862496	0,5500
0,4550	1,136062	0,516908	0,994149	0,477241	0,875672	0,5450
0,4600	1,120294	0,515335	0,995378	0,480043	0,888969	0,5400
0,4650	1,104697	0,513684	0,996462	0,482778	0,902389	0,5350
0,4700	1,089267	0,511956	0,997402	0,485446	0,915936	0,5300
0,4750	1,074001	0,510150	0,998196	0,488046	0,929611	0,5250
0,4800	1,058894	0,508269	0,998846	0,490577	0,943416	0,5200
0,4850	1,043943	0,506313	0,999351	0,493038	0,957356	0,5150
0,4900	1,029156	0,504282	0,999711	0,495430	0,971431	0,5100
0,4950	0,014500	0,502177	0,999928	0,497751	0,985645	0,5050
0,5000	1,000000	0,500000	1,000000	0,500000	1,000000	0,5000

Приложение 3.

Таблица многочленов, не приводимых над полем $GF(2)$

№ п/п	Степень	Многочлен	Двоичная последовательность
1	1	$x + 1$	11
2	2	$x^2 + x + 1$	111
3	3	$x^3 + x + 1$	1011
4		$x^3 + x^2 + 1$	1101
5	4	$x^4 + x + 1$	10011
6		$x^4 + x^2 + 1$	11001
7		$x^4 + x^3 + x^2 + x + 1$	11111
8	5	$x^5 + x^2 + 1$	100101
9		$x^5 + x^3 + 1$	101001
10		$x^5 + x^3 + x^2 + x + 1$	101111
11		$x^5 + x^3 + x^2 + x + 1$	110111
12		$x^5 + x^4 + x^3 + x + 1$	111011
13		$x^5 + x^4 + x^3 + x^2 + 1$	111101
14	6	$x^6 + x + 1$	1000011
15		$x^6 + x^3 + 1$	1001001
16		$x^6 + x^4 + x^2 + x + 1$	1010111
17		$x^6 + x^4 + x^3 + x + 1$	1011011
18		$x^6 + x^5 + 1$	1100001
19		$x^6 + x^5 + x^2 + x + 1$	1100111
20		$x^6 + x^5 + x^2 + x^2 + 1$	1101101
21		$x^6 + x^5 + x^4 + x + 1$	1110011
22		$x^6 + x^5 + x^4 + x^2 + 1$	1110101
23		$x^7 + x + 1$	10000011
24		$x^7 + x^3 + 1$	10001001
25		$x^7 + x^3 + x^2 + x + 1$	10001111
26		$x^7 + x^4 + 1$	10010001
27		$x^7 + x^4 + x^3 + x^2 + 1$	10011101
28		$x^7 + x^5 + x^2 + x + 1$	10100111
29		$x^7 + x^5 + x^3 + x + 1$	10101011
30		$x^7 + x^5 + x^4 + x^3 + 1$	10111001
31		$x^7 + x^5 + x^4 + x^3 + x^2 + x + 1$	10111111
32	7	$x^7 + x^6 + 1$	11000001
33		$x^7 + x^6 + x^3 + x + 1$	11001011
34		$x^7 + x^6 + x^4 + x + 1$	11010011
35		$x^7 + x^6 + x^4 + x^2 + 1$	11010101
36		$x^7 + x^6 + x^5 + x^2 + 1$	11100101
37		$x^7 + x^6 + x^5 + x^3 + x^2 + x + 1$	11101111
38		$x^7 + x^6 + x^5 + x^4 + 1$	11110001
39		$x^7 + x^6 + x^5 + x^4 + x^2 + x + 1$	11110111

Продолжение приложения 2.

№ п/п	Степень	Многочлен	Двоичная последовательность
40	8	$x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + 1$	11111101
41		$x^8 + x^4 + x^3 + x + 1$	100011011
42		$x^8 + x^4 + x^3 + x^2 + 1$	100011101
43		$x^8 + x^5 + x^3 + x + 1$	100101011
44		$x^8 + x^5 + x^3 + x^2 + 1$	100101101
45		$x^8 + x^5 + x^4 + x^3 + 1$	100111001
46		$x^8 + x^5 + x^4 + x^3 + x^2 + x + 1$	100111111
47		$x^8 + x^6 + x^3 + x^2 + 1$	101001101
48		$x^8 + x^5 + x^4 + x^3 + x^2 + x + 1$	101011111
49		$x^8 + x^6 + x^5 + x + 1$	101100011
50		$x^8 + x^6 + x^5 + x^2 + 1$	101100101
51		$x^8 + x^6 + x^5 + x^3 + 1$	101101001
52		$x^8 + x^6 + x^5 + x^4 + 1$	101110001
53		$x^8 + x^6 + x^5 + x^4 + x^2 + x + 1$	101110111
54		$x^8 + x^6 + x^5 + x^4 + x^3 + x + 1$	101111011
55		$x^8 + x^7 + x^2 + x + 1$	110000111
56		$x^8 + x^7 + x^3 + x + 1$	110001011
57		$x^8 + x^7 + x^3 + x^2 + 1$	110001101
58		$x^8 + x^7 + x^4 + x^3 + x^2 + x + 1$	110011111
59		$x^8 + x^7 + x^5 + x + 1$	110100011
60		$x^8 + x^7 + x^5 + x^3 + 1$	110101001
61		$x^8 + x^7 + x^5 + x^4 + 1$	110110001
62		$x^8 + x^7 + x^5 + x^4 + x^3 + x^2 + 1$	110111101
63		$x^8 + x^7 + x^6 + x + 1$	111000011
64		$x^8 + x^7 + x^6 + x^3 + x^2 + 1$	111001111
65		$x^8 + x^7 + x^6 + x^4 + x^2 + x + 1$	111010111
66		$x^8 + x^7 + x^6 + x^4 + x^3 + x^2 + 1$	111011101
67		$x^8 + x^7 + x^6 + x^5 + x^3 + x + 1$	111100111
68		$x^8 + x^7 + x^6 + x^5 + x^4 + x + 1$	111110011
69		$x^8 + x^7 + x^6 + x^5 + x^4 + x^2 + 1$	111110101
70		$x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + 1$	111111001

СОДЕРЖАНИЕ

	ВВЕДЕНИЕ.....	3
1.	Информационные системы и теория систем	6
1.1.	Основные теории систем	6
2.	Информационные системы и теория информации понятия	23
2.1.	Основные понятия и определения информационной системы	26
2.2.	Виды информации. Этапы обращения информации	28
2.3.	История и перспективы развития информационных систем	32
3.	Представление данных в информационных системах	39
3.1.	Системы счисления	39
3.2.	Представление информации в компьютере	43
3.3.	Представление отрицательных чисел и графической информации	47
3.4.	Организация информации в компьютере ...	49
3.4.1.	Файлы	49
3.4.2.	Файловые системы	49
3.4.3.	Файловая система FAT	53
3.4.3.1.	Исследование структуры логического диска файловой системы FAT	53
3.4.4.	Организация данных на логических дисках системы	54
3.4.5.	Структура FS INFO	58
3.4.6.	Назначение и внутренняя организация таблиц размещения файлов	59
3.4.7.	Каталог (Directory)	59
3.4.8.	Длинное имя файла (LFN)	61

3.4.9.	Логическая структура жесткого диска. Организация данных на жестком диске	63
4.	Основные понятия, связанные с информацией. Информационные характеристики, определяющие качество ИС.....	68
4.1.	Структурные меры информации	68
4.1.1.	Геометрическая мера информации	68
4.1.2.	Комбинаторная мера информации	69
4.1.3.	Аддитивная мера (мера Хартли)	71
4.2.	Семантические меры информации	72
4.2.1.	Содержательность информации	73
4.2.2.	Целесообразность информации	74
4.2.3.	Динамическая энтропия	77
4.3.	Статистические меры информации	78
4.3.1.	Вероятность и информация	78
4.3.2.	Свойства энтропии ансамбля	82
4.3.3.	Условная энтропия и энтропия объединения	86
4.3.4.	Дифференциальная энтропия	92
4.3.5.	Эпсилон-энтропия	96
4.4.	Информационные потери при передаче по дискретным каналам связи с шумами	97
4.4.1.	Скорость передачи информации по дискретному каналу связи	102
4.4.2.	Пропускная способность дискретного канала без помех	103
4.4.3.	Пропускная способность дискретного канала с помехами.....	105
4.4.4.	Избыточность	110
5 .	Технологии работы с информацией	116
5.1.	Общие понятия теории кодирования информации	116
5.2.	Кодирование информации при передаче по дискретному каналу без помех	117

5.2.1.	Аналого-цифровые преобразователи (АЦП)	118
5.2.2.	Эффективное кодирование. Методика Шеннона–Фэно	120
5.3.	Методика Хаффмана	124
5.3.1.	Алгоритм LZW	126
5.3.2.	Арифметическое кодирование	128
5.4.	Помехоустойчивое кодирование	130
5.4.1.	Теоретические основы блоковых кодов	131
5.4.2.	Линейные групповые коды	134
5.4.3.	Циклическое кодирование	143
5.4.3.1.	Построение циклических кодов.	
	Обнаружение и исправление ошибок.....	143
	ЗАКЛЮЧЕНИЕ.....	151
	БИБЛИОГРАФИЧЕСКИЙ СПИСОК	152
	ГЛОССАРИЙ	155
	ПРИЛОЖЕНИЕ 1	159
	ПРИЛОЖЕНИЕ 2	162
	ПРИЛОЖЕНИЕ 3	166

Учебное издание

Яскевич Татьяна Владимировна
Айтчанов Бекмурза Хусаинович
Байматаева Шолпан Муратовна

ОСНОВЫ ИНФОРМАЦИОННЫХ СИСТЕМ

Учебное пособие

Нач. ОПНП РУ ДСО
Редактор
Компьютерная верстка

З. А. Губайдулина
З. А. Губайдулина
С. Е. Орысбекова

Подписано в печать 01.02.2017 г.

Тираж 300 экз. Формат 60х84х 1/16. Бумага типогр. № 1.
Уч.-изд.л. 10,6. Усл. п.л. 9,9. Заказ № 793. Цена договорная.

Издание Казахского национального исследовательского
технического университета имени К. И. Сатпаева
Департамент по связям с общественностью КазННТУ
Издательское управление
г. Алматы, ул. Сатпаева, 22

ISBN 978-601-323-081-8



9

7 8 6 0 1 3 2 3 0 8 1 8