Глава 2

Кодировании информации

Параграф 1

Рельефно-точечная система Брайля как пример кодирования информации

Коротко опишем историю создания рельефно-точечной системы Брайля. В 1828 году Луи Брайль окончил Парижский институт слепых и был оставлен в нем преподавателем математики и музыки. В этом же году он завершил работу над системой рельефно-точечных обозначений.

В 1829 году Луи Брайль впервые опубликовал эту систему. Она была изложена в брошюре «Способ написания слов, музыки и песнопений при помощи точек». Однако, предложенная Брайлем система была отклонена педагогическим советом Парижского института слепых.

В 1837 году стараниями Луи Брайля и его учеников была устроена небольшая печатня, в которой полукустарным способом рельефным шеститочием была набрана первая брайлевская книга «Краткая история Франции». В этом же году он опубликовал книгу «Методы написания слов, нот и церковных хоровых пений посредством точек для использования слепыми».

В 1838 году по Брайлю был издан учебник арифметики. Только в 1852 году, через 15 лет после издания первой книги, когда изобретателя уже не было в живых (он умер 6 января того же года), вышла в свет первая типографская книга, напечатанная шрифтом Брайля. В 1861 году был издан букварь шрифтом Брайля на французском языке, получивший название “Методика чтения”. Только после смерти Луи Брайля система его письма получила официальное признание.

В настоящее время система рельефно-точечных обозначений Брайля используется в подавляющем большинстве стран мира. С 1950 года по инициативе ЮНЕСКО стала проводиться работа по применению системы Брайля для восточных языков и ее распространению среди слепых Африки.

Основой рельефно-точечной системы Брайля является шеститочие. Из различных комбинаций шести точек составлены все брайлевские знаки (буквы, цифры, знаки препинания, математические, физические, химические знаки и обозначения, а также знаки нотной системы).

Точки шеститочия занумерованы следующим образом: точки левой (при чтении) половины клетки в порядке сверху вниз имеют номера 1, 2, 3, а точки правой половины также в порядке сверху вниз имеют номера 4, 5, 6.

Так как брайлевское письмо на приборе осуществляется с обратной стороны листа, то нумерация точек при чтении зеркальна по отношению к нумерации точек шеститочия при письме.

Шесть точек брайлевского шеститочия позволяют образовать 63 комбинации. Эти комбинации дают возможность использовать систему обозначений не только для алфавитов большинства языков, но и математических знаков и знаков нотного письма.

Записывая номера точек, образующих буквы русского алфавита, можно закодировать любое текстовое сообщение. Например, слово «информатика» будет закодировано следующей последовательностью цифр:

24, 1345, 124,135, 1235, 134, 1, 2345, 24, 13, 1 (проверьте это).

Брайлевские дисплеи и принтеры могут отображать информацию в компьютерном восьмиточечном брайле. Используя 8 точек можно получить уже 255 различных комбинаций. В восьмиточие седьмая точка располагается под третьей, а восьмая под шестой. Брайлевские принтеры (о которых более подробно будет рассказано в главе5) используют систему кодирования, в которой каждый бит байта отвечает за одну конкретную точку. Т.е., если на месте некоторого бита стоит 1, то принтер пробивает соответствующую точку, а если 0, то точка не пробивается. Очевидно, что такой способ хранения «брайлевской» информации в компьютере значительно более эффективен, чем код, основанный на номерах точек.

Из истории тифлопедагогики известны неоднократные попытки оптимизации системы Брайля. Желание сделать это было обусловлено с одной стороны, потребностью создать интернациональную систему письма для слепых на основе системы Брайля. С другой стороны, предпринимались попытки «минимизировать код», т.е. изменить систему Брайля так, чтобы буквы, чаще всего встречающиеся в текстах, обозначались наименьшим количеством точек. Предполагалось, что, поскольку, буквы, чаще всего встречающиеся, обозначаются минимальным количеством точек, то затраты времени и физических усилий при письме будут минимальны.

На основе современных алгоритмов можно разработать наиболее экономичный код для записи по рельефно-точечной системе. Самым экономичным будет код, в котором каждый элементарный символ будет передавать максимальную информацию. Элементарный символ в случае системы Брайля это одна точка шеститочия (т.е. один двоичный символ). Подобную задачу можно решить, например, с помощью «кода Шеннона-Фано». Принцип построения его в том, что кодируемые символы (буквы или комбинации букв) разделяются на две приблизительно равновероятные группы: для первой группы символов на первом месте комбинации ставится 0, для второй 1. далее каждая группа снова делится на две приблизительно равновероятные подгруппы; для символов первой подгруппы на втором месте ставиться 0; для второй подгруппы – единица и т. д.

Алгоритм Шеннона — Фано — один из первых алгоритмов сжатия, который сформулировали американские учёные Шеннон и Фано. Данный метод сжатия имеет большое сходство с алгоритмом Хаффмана, который появился на несколько лет позже и является логическим продолжением алгоритма Шеннона. Алгоритм использует коды переменной длины: часто встречающийся символ кодируется кодом меньшей длины, редко встречающийся — кодом большей длины.

Коды Шеннона — Фано являются префиксными, т.е. никакое кодовое слово не является началом любого другого. Это свойство позволяет однозначно декодировать любую последовательность кодовых слов. Подробнее эти алгоритмы здесь рассматриваться не будут.

Контрольные вопросы

1. Расскажите об истории создания рельефно-точечной системы Брайля.

2. Какие коды называются префексными?

3. Является ли код, состоящий в кодировании букв номерами точек их брайлевского написания префексным?

4. Как вы думаете, почему система рельефно-точечных обозначений Брайля так и не была минимизирована?

Параграф 2

Представление информации в компьютере

Как уже говорилось выше, в компьютере для представления информации используется двоичное кодирование, так как удалось создать надежно работающие технические устройства, которые могут со стопроцентной надежностью сохранять и распознавать не более двух различных состояний (цифр), например:

- электромагнитные реле (замкнуто/разомкнуто), широко использовались в конструкциях первых ЭВМ;

- участок поверхности магнитного носителя информации (намагничен/размагничен);

- участок поверхности лазерного диска (отражает/не отражает);

- триггер, может устойчиво находиться в одном из двух состояний, широко используется в оперативной памяти компьютера.

Все виды информации в компьютере кодируются на машинном языке, в виде последовательностей нулей и единиц. Алфавит такого кодирования состоит из двух символов 0 и 1. В теории кодирования алфавитом называется набор символов, используемых для записи кода.

Цифры двоичного кода можно рассматривать как два равновероятных состояния (события). При записи двоичной цифры реализуется выбор одного из двух возможных состояний и, следовательно, она несет количество информации, равное 1 биту. Таким образом, две цифры несут информацию в 2 бита, три цифры - в 3 бита и так далее.

Количество информации в битах равно количеству цифр двоичного машинного кода. Подробнее о подходах к измерению количества информации говорилось в параграфе 4 главы 1.

Двоичная система кодирования сообщений применяется также и при передаче по каналам связи Это значит, что каждому символу, используемому для обычной записи сообщения (например, букве русского языка), сопоставлена последовательность, состоящая из 0 и 1. Именно такое сопоставление представлено, например, кодовыми таблицами, задающими ASCII-кодирование. Более подробно о таблицах кодирования символов будет рассказано в параграфе 3 этой главы.

В расширении CP-1251 символы “а”, “и”, “п”, “р” задаются кодами 11100000, 11101000, 11101111, 11110000 соответственно. Таким образом, слово “пир” будет закодировано последовательностью 111011111110100011110000.

Предположим, что при передаче этого кодового сообщения произошла ошибка и вместо одной из единиц в результате каких-либо помех был передан 0. Т.е. на приемник информации поступила последовательность 111011111110000011110000. Тогда эта последовательность будет декодирована как “пар” (проверьте это!).

К подобной ошибке мог привести обыкновенный технический сбой, например, перепад в напряжении. Воздействие, приводящее к искажению передаваемой информации, обычно называют шумом.

Если сообщение носит “бытовой” характер, подобная ошибка может и не привести к тяжелым последствиям. Однако, если речь идет о передаче команд управления космическим кораблем или атомной электростанцией, то последствия могут быть весьма серьезными.

Существует метод, позволяющий исправлять ошибки, возникающие при передаче информации. Чтобы продемонстрировать идею кода, позволяющего обнаруживать ошибки, предположим, что все передаваемые сообщения - это числовые данные, записываемые цифрами обычной десятичной системы счисления. Каждую цифру будем кодировать ее представлением в двоичной системе счисления. Результат такого кодирования представлен в двух первых столбцах следующей таблицы (на третий столбец пока не обращаем внимание):

0 0000 00000

1 0001 00011

2 0010 00101

3 0011 00110

4 0100 01001

5 0101 01010

6 0110 01100

7 0111 01111

8 1000 10001

9 1001 10010

Очевидно, что предложенное кодирование не позволяет обнаружить ошибку. Например, если вместо 0010 пришло ошибочное 0011, то в такое сообщение можно поверить как в правильное.

Теперь добавим в конце кода каждой цифры еще один двоичный символ. Получившийся таким образом расширенный код представлен в третьем столбце таблицы.

Последний символ в код приписывается по следующему правилу: если в исходном четырехсимвольном коде четное число единиц, то пишем 0, если нечетное, то пишем 1. В получившемся коде для любого символа количество единиц всегда четно. Поэтому, если при передаче сообщения в каком-то месте произошла ошибка, т.е. 0 заменился на 1 или наоборот, то количество единиц в таком коде соответствующего символа стало нечетным, и это легко обнаруживается. Например, пришло сообщение: 100100011010011. Разбиваем его на три группы по 5 символов: 10010 00110 10011. Сразу видно: первые две группы правильные, а третья — нет.

Чтобы описать способность кода к распознаванию ошибок, используют понятие расстояния между словами. Пусть даны слова над одним алфавитом. Тогда расстоянием между словами называется количество позиций, в которых символы одного слова не совпадают с символами второго. Например, расстояние между словами "стог" и “снег" равно 2 - они отличаются во второй и третьей позициях. А между словами 1001001 и 0100001 расстояние равно 3, поскольку они отличаются в первой, второй и четвертой позициях. Расстояние между словами называют расстоянием Хэмминга.

Обычно под расстоянием понимают некоторую геометрическую величину, характеризующую удаленность объектов друг от друга. Однако, на практике расстояние представляют различными величинами. На плоскости под расстоянием обычно понимают длину отрезка, соединяющего две точки. На поверхности земного шара, естественно расстоянием считать дугу большого круга, проходящего через две рассматриваемые точки. В городе расстояние между двумя точками измеряют вдоль улиц, по которым можно добраться из одной точки в другую и т.д.

Математики выяснили, что общими для всех разновидностей расстояний являются три свойства (три аксиомы метрики). Обозначим через D(A, B) функцию, которая двум объектам (например, точкам или словам) сопоставляет неотрицательное число. Эта функция должна удовлетворять следующим аксиомам:

1. D(A, B) = 0 тогда и только тогда, когда A = B (равенство здесь означает совпадение объектов);

2. D(A, B) = D(B, A);

3. D(A, B) < D(A, C) + D(C, B), каким бы ни был объект C.

Смысл первого свойства очевиден. Второе свойство утверждает, что объект A удален от объекта B так же, как объект B удален от объекта A. наконец, третье свойство говорит, что дорога через третий объект C всегда длиннее, нежели прямой путь. Третье свойство обычно называют неравенством треугольника за его естественную геометрическую аналогию: сумма двух сторон треугольника больше третьей стороны.

В математике принято любую функцию, обладающую указанными тремя свойствами, называть расстоянием (или метрикой). Расстояние Хэмминга обладает всеми тремя свойствами. Попробуйте доказать это.

Теперь опять увеличим рассматриваемый выше код, добавив еще две двоичных цифры в конце. Приведем новую таблицу кодов, в которой два первых столбца те же, а в третьем приведен семибитовый расширенный код:

0 0000 0000000

1 0001 0001111

2 0010 0010110

3 0011 0011001

4 0100 0100101

5 0101 0101010

6 0110 0110011

7 0111 0111100

8 1000 1000011

9 1001 1001100

При таком кодировании минимальное расстояние между кодовыми словами равно 3. Это означает, что если при передаче сообщения произошло две ошибки, то все равно принятое слово не совпадет ни с одним кодовым словом. Таким образом, будет выявлено ошибочно переданное слово. Более того, весьма маловероятно, чтобы в семибитовом слове оказалось сразу две ошибки, поэтому, получив слово с ошибкой, можно найти ближайшее к нему слово (т.е. отличающееся только на один символ) и исправить ошибку. Заметим, что поскольку расстояние между любыми двумя кодовыми словами не меньше 3, кодовое слово, ближайшее к ошибочному, будет единственным.

Пусть, например, получено сообщение 001011101100101010111. Разобьем его на группы по 7 символов, получим: 0010111, 0110010, 1010111. Во второй таблице нет кодового слова, соответствующего первой группе символов. Но на расстоянии 1 от него находится код 0010101, значит, допущена одна ошибка, а исходно была передана цифра 2. Вторая группа является кодовым словом, она соответствует цифре 6. А третья группа снова ошибочна. Ближайшее к ней кодовое слово - 1000111. Это код цифры 8. Значит, было передано число 268 (проверьте это по таблице).

Таким образом, построенный код гарантированно исправляет одну ошибку в закодированном слове. Причем, исправление производится по четкому алгоритму, что позволяет автоматизировать процесс исправления ошибок. Такой код получил название кода Хэмминга.

Разработанная математиками теория кодирования позволяет строить коды с заданным минимальным расстоянием между кодовыми словами. Так, в европейских системах связи широко используется 235-битовый код, расширенный с помощью дополнительных 20 двоичных символов. Минимальное расстояние между словами этого кода равно 7. Такой код гарантированно обнаруживает 6 ошибок и исправляет слова, в которых допущено не более 3 ошибок. В течение многих лет эксплуатации этих систем не было случая, чтобы ошибка прошла незамеченной.

В конце параграфа приведем два примера решения задач на кодирование информации и на перевод из одной системы счисления в другую.

пример 1. Для кодирования букв О, В, Д, П, А решили использовать двоичное представление чисел 0, 1, 2, 3 и 4 соответственно (в записи используется минимум 2 разряда с сохранением одного незначащего нуля в случае одноразрядного представления). Закодируйте слово «ВОДОПАД» таким способом и результат запишите восьмеричным кодом.

Решение. Сначала следует представить числа от 0 до 4 в двоичном коде:

0 = 02;

1= 12;

2 = 102;

3 = 112;

4 = 1002.

выпишем двоичный код каждой буквы, согласно условию задачи:

О = 00;

В = 01;

Д = 10;

П = 11;

А = 100.

Кодируем последовательность букв «ВОДОПАД» и получаем код:

010010001110010.

Теперь разобьём этот код на тройки справа налево и переведём полученные группы в десятичный код, зная, что восьмеричное представление совпадает с десятичным при разбиении тройками.

010 010 001 110 010 = 210 210 110 610 210 =28 28 18 68 28 .

Собирая восьмеричные знаки вместе получаем результат: 221628.

Ответ: 22162.

пример 2. Для передачи по каналу связи сообщения, состоящего только из символов А, Б, В и Г, используется посимвольное кодирование: А - 10, Б - 11, В - 110, Г - 0. Через канал связи передаётся сообщение: «ВАГБААГВ». Закодируйте сообщение данным кодом. Полученное двоичное число переведите в шестнадцатеричный вид.

Решение. Закодируем последовательность букв: ВАГБААГВ — 1101001110100110. Теперь разобьём это представление на четвёрки справа налево и переведём полученный набор чисел сначала в десятичный код, затем в шестнадцатеричный:

1101 0011 1010 0110 = 13 3 10 6 = D3A6.

Ответ: D3A6.

Контрольные вопросы

1. Что такое шум с точки зрения процесса передачи информации?

2. Что такое алфавит в теории кодирования?

3. Приведите примеры процессов передачи информации, в которых:

А) Ошибка не приведет к серьезным последствиям;

Б) Ошибка приведет к серьезным последствиям.

4. Что такое расстояние между словами?

5. Как можно исправить ошибочное слово используя расстояние между словами в коде Хэмминга?

Упражнения для самостоятельного выполнения

1. Найдите расстояние между словами каждой пары:

а) собака и корова;

б) паровоз и самовар;

в) 10010110 и 10110100.

2. Рассматривается множество всех пятисимвольных слов над алфавитом, состоящим из символов 0 и 1.

а) Перечислите все слова, находящиеся на расстоянии 1 от слова 10101;

б) Перечислите все слова, находящиеся на расстоянии 2 от слова 01010.

3. Получено сообщение, закодированное семибитовым кодом Хэмминга. Декодируйте его, используя вторую таблицу из текста параграфа, исправив, если необходимо, ошибки:

А) 0010111001000010000001000001;

Б) 1001101011010001011100001011.

4. Приведенное во второй таблице кодирование десяти цифр - это только часть кода, изобретенного Хэммингом для кодирования всевозможных четырехбитовых последовательностей. Минимальное расстояние между кодовыми словами в этом коде равно 3. Найдите кодовые слова для остальных шести четырехбитовых последовательностей так, чтобы минимальное расстояние между словами осталось равным 3.

5. Для кодирования букв Д, Х, Р, О, В решили использовать двоичное представление чисел 0, 1, 2, 3 и 4 соответственно (с использованием минимум двух разрядов с добавлением незначащего нуля). Закодируйте слово «ХОРОВОД» таким способом и результат запишите восьмеричным кодом.

6. Для кодирования букв О, К, Г, Д, Р решили использовать двоичное представление чисел 0, 1, 2, 3 и 4 соответственно (с сохранением одного незначащего нуля в случае одноразрядного представления). Закодируйте слово «ГОРОДОК» таким способом и результат запишите восьмеричным кодом.

7. Для передачи по каналу связи сообщения, состоящего только из символов А, Б, В и Г, используется неравномерный (по длине) код: А- 0, Б- 11, В- 100, Г- 011. Через канал связи передаётся сообщение: ГБАВАВГ. Закодируйте сообщение данным кодом. Полученное двоичное число переведите в восьмеричный вид.

8. Для передачи по каналу связи сообщения, состоящего только из символов А, Б, В и Г, используется неравномерный (по длине) код: А- 10, Б- 11, В- 110, Г- 0. Через канал связи передаётся сообщение: ВАГБААГВ. Закодируйте сообщение данным кодом. Полученное двоичное число переведите в восьмеричный вид.

9. Для кодирования букв А, Б, В, Г решили использовать двухразрядные последовательные двоичные числа (от 00 до 11 соответственно). Закодируйте таким образом последовательность символов ГБВА и запишите результат шестнадцатеричным кодом.

10. Для кодирования букв A, B, C, D используются четырехразрядные последовательные двоичные числа, начинающиеся с 1 (от 1001 до 1100 соответственно). Закодируйте таким образом последовательность символов CADB и запишите результат в шестнадцатеричном коде.

Параграф 3

Кодирование текстовой информации

Информация, выраженная с помощью естественных и формальных языков в письменной форме, обычно называется текстовой информацией. Для обработки текстовой информации на компьютере она должна быть представлена в двоичном коде.

Человек различает знаки по их начертанию, а компьютер - по их двоичным кодам. При вводе в компьютер текстовой информации происходит ее двоичное кодирование, изображение знака преобразуется в его двоичный код. Пользователь нажимает на клавиатуре клавишу со знаком, и в компьютер поступает определенная последовательность электрических импульсов (двоичный код знака).

В процессе вывода знака на экран компьютера производится обратное перекодирование, т. е. преобразование двоичного кода знака в его изображение.

Присваивание знаку конкретного двоичного кода - это вопрос соглашения, которое фиксируется в кодовой таблице (или кодовой странице code page). В существующих кодовых таблицах первые 32 кода (десятичные коды с 0 по 31) соответствуют не знакам, а операциям (перевод строки, прокрутка страницы принтером и т.д.). Десятичным кодом 32 закодирован символ пробела. Пробел – это символ, а не его отсутствие.

В США, в 1963 году была разработана и стандартизована Таблица ASCII (American standard code for information interchange). Название «ASCII» по-русски часто произносится как «аски».

Для кодирования каждого знака таблицы ASCII требуется количество информации, равное 8 битам, т. е. длина двоичного кода каждого знака составляет восемь двоичных знаков, т.е. один байт. Каждому знаку ставится в соответствие уникальный двоичный код из интервала от 00000000 до 11111111 (в десятичном представлении от 0 до 255).

Таблица ASCII определяет коды для символов:

десятичных цифр;

латинского алфавита (большие и малые буквы);

национального алфавита (например, русского);

знаков препинания;

управляющих символов.

Десятичные коды с 32 по 127 являются интернациональными и соответствуют буквам латинского алфавита (большим и малым), цифрам, скобкам, специальным символам (например, @), знакам арифметических операций и знакам препинания.

Десятичные коды с 128 по 255 являются национальными, т. е. в различных национальных кодировках одному и тому же коду соответствуют разные знаки.

До недавнего времени существовало несколько различных кодовых таблиц для русских букв (Windows CP1251, MS-DOS CP866, КОИ-8, Mac, ISO), поэтому тексты, созданные в одной кодировке, не могли правильно отображаться в другой. Например, в кодировке Windows (кодовая страница CP1251) последовательность числовых кодов 221, 194, 204 образует слово "ЭВМ", тогда как в других кодировках это будет бессмысленный набор символов.

В большинстве случаев пользователь не должен заботиться о перекодировках текстовых документов, так как это делают специальные программы-конверторы, встроенные в операционную систему и приложения.

В последние годы широкое распространение получил новый международный стандарт кодирования текстовых символов Unicode, который отводит на каждый символ 2 байта (16 битов). ▌По формуле можно определить количество символов, которые можно закодировать согласно этому стандарту:

N = 2i = 216 = 65536

Такого количества символов оказалось достаточно, чтобы закодировать не только русский и латинский алфавиты, цифры, знаки и математические символы, но и греческий, арабский, иврит и другие алфавиты.

В конце параграфа приведем пример решения задачи на кодирование текстовой информации.

пример 1. Текстовый документ, состоящий из 3072 символов, хранился в 8-битной кодировке КОИ-8. Этот документ был преобразован в 16-битную кодировку Unicode. Укажите, какое дополнительное количество Кбайт потребуется для хранения документа. В ответе запишите только число.

решение. Первоначально для кодирования текста была использована кодировка, при которой каждый символ занимает объём памяти, равный 8 бит или 1байт. Вычислим объем информации в кодировке КОИ-8:

3072\*1 байт = 3072 байта.

Вычислим объем информации в 16-битной кодировке (один символ занимает память, равную двум байтам):

3072\*2 байта = 6144 байта.

Вычислим дополнительный объем памяти, который потребуется для хранения перекодированного файла:

6144 - 3072 = 3072 байта.

Переведем данную величину в килобайты:

3072 : 1024 = 3 Кбайта.

Ответ: 3.

Контрольные вопросы

1. Что такое текстовая информация?

2. Зачем нужно двоичное кодирование текстовой информации?

3. Что такое кодовая таблица?

4. Расскажите о кодировке ASCII.

5. С какой целью ввели стандарт кодирования Unicode?

Упражнения для самостоятельного выполнения

1. В кодировке стандарта Unicode на каждый символ отводится два байта. Определите информационный объем слова из двадцати четырех символов в этой кодировке.

2. Автоматическое устройство осуществило перекодировку текстового сообщения из 16-битного кода в 8-битную кодировку. При этом сообщение уменьшилось на 20 байт. Какова длина сообщения в символах?

3. Во сколько раз уменьшится информационный объем текста из двадцати четырех символов при его преобразовании из 16-битной кодировки в 1-байтную кодировку?

4. Автоматическое устройство осуществило перекодировку текстового сообщения на русском языке, первоначально записанного в 16-битном стандарте Unicode, в 8-битную кодировку КОИ-8. При этом сообщение уменьшилось на 800 бит. Какова длина сообщения в символах?

5. Автоматическое устройство осуществило перекодировку текстового сообщения на русском языке, первоначально записанного в 16–битном стандарте Unicode, в 8–битную кодировку Windows CP1251, при этом информационный объем сообщения составил 60 байт. Определите информационный объем сообщения до перекодировки.

Параграф 4

Кодирование звуковой информации

Как известно из курса физики, звук представляет собой распространяющуюся в воздухе, воде или другой среде волну с непрерывно меняющейся амплитудой и частотой.

Человек воспринимает звуковые волны (колебания воздуха) с помощью слуха в форме звука различных громкости и тона. Чем больше амплитуда звуковой волны, тем громче звук, чем больше частота волны, тем выше тон звука.

Человеческое ухо воспринимает звук с частотой от 20 колебаний в секунду (низкий звук) до 20000 колебаний в секунду (высокий звук).

Человек может воспринимать звук в огромном диапазоне амплитуд (интенсивностей), в котором максимальная амплитуда (интенсивность) больше минимальной в 1014 раз (в сто тысяч миллиардов раз). Для измерения громкости звука применяется специальная единица "децибел" (дбл). Уменьшение или увеличение громкости звука на 10 дбл соответствует уменьшению или увеличению интенсивности звука в 10 раз.

Для того чтобы компьютер мог обрабатывать звук, непрерывный звуковой сигнал должен быть преобразован в цифровую дискретную форму с помощью временной дискретизации.

Напомним, что преобразование непрерывного объекта в множество отделимых друг от друга частей (т.е. в дискретное множество) называется дискретизацией.

Непрерывная звуковая волна разбивается на отдельные маленькие временные (временн’ые ударение на «ы») участки, для каждого такого участка устанавливается определенная величина амплитуды (интенсивности) звука. Таким образом, непрерывная зависимость громкости звука от времени A(t) заменяется на дискретную последовательность уровней громкости. На графике это выглядит как замена гладкой кривой на последовательность "ступенек".

Для записи аналогового звука и его преобразования в цифровую форму используется микрофон, подключенный к звуковой плате. Качество полученного цифрового звука зависит от количества измерений уровня громкости звука в одну секунду, т. е. частоты дискретизации. Чем большее количество измерений производится за одну секунду (чем больше частота дискретизации), тем точнее "лесенка" цифрового звукового сигнала повторяет кривую аналогового сигнала.

Частота дискретизации звука - это количество измерений громкости звука за одну секунду.

Обычно в компьютерных программах обработки звуковой информации частота дискретизации лежит в диапазоне от 8000 до 48000 измерений громкости звука за одну секунду.

Каждой "ступеньке" присваивается определенное значение уровня громкости звука. Уровни громкости звука можно рассматривать как набор возможных состояний N, для кодирования которых необходимо определенное количество информации I, которое называется глубиной кодирования звука. Таким образом, глубина кодирования звука - это количество информации, которое необходимо для кодирования дискретных уровней громкости цифрового звука.

Если известна глубина кодирования, то количество уровней громкости цифрового звука можно рассчитать по формуле N = 2I. Пусть глубина кодирования звука составляет 16 бит, тогда количество уровней громкости звука равно:

N = 2I = 216 = 65536

В процессе кодирования каждому уровню громкости звука присваивается свой 16-битовый двоичный код, наименьшему уровню звука будет соответствовать код 0000000000000000, а наибольшему - 1111111111111111.

Чем больше частота и глубина дискретизации звука, тем более качественным будет звучание оцифрованного звука. Самое низкое качество оцифрованного звука, соответствующее качеству телефонной связи, получается при частоте дискретизации 8000 раз в секунду, глубине дискретизации 8 битов и записи одной звуковой дорожки (режим "моно"). Самое высокое качество оцифрованного звука, соответствующее качеству аудио-CD, достигается при частоте дискретизации 48000 раз в секунду, глубине дискретизации 16 битов и записи двух звуковых дорожек (режим "стерео").

Заметим, что чем выше качество цифрового звука, тем больше информационный объем звукового файла. Можно оценить информационный объем цифрового стереозвукового файла длительностью звучания 1 секунда при среднем качестве звука (16 битов, 24000 измерений в секунду). Для этого глубину кодирования необходимо умножить на количество измерений в 1 секунду и умножить на 2 (стереозвук):

16\*24000\*2 = 768000 бит = 96000 байт = 93,75 Кбайт

Звуковые редакторы позволяют не только записывать и воспроизводить звук, но и редактировать его. Оцифрованный звук представляется в звуковых редакторах в наглядной форме, поэтому операции копирования, перемещения и удаления частей звуковой дорожки можно легко осуществлять с помощью соответствующих клавиатурных команд. Кроме того, можно накладывать звуковые дорожки друг на друга (микшировать звуки) и применять различные акустические эффекты (эхо, воспроизведение в обратном направлении и др.).

Звуковые редакторы позволяют изменять качество цифрового звука и объем звукового файла путем изменения частоты дискретизации и глубины кодирования. Оцифрованный звук можно сохранять без сжатия в звуковых файлах в универсальном формате WAV или в формате со сжатием mp3. При сохранении звука в форматах со сжатием отбрасываются "избыточные" для человеческого восприятия звуковые частоты с малой интенсивностью, совпадающие по времени со звуковыми частотами с большой интенсивностью. Применение такого формата позволяет сжимать звуковые файлы в десятки раз, однако приводит к необратимой потере информации, т.е. файлы не могут быть восстановлены в первоначальном виде.

В конце параграфа приведем пример решения задач на кодирование звуковой информации.

пример 1. Производилась двухканальная (стерео) звукозапись с частотой дискретизации 64 кГц и 24-битным разрешением (глубина кодирования). В результате был получен файл размером 48 Мбайт, сжатие данных не производилось. Определите приблизительно, сколько времени (в минутах) проводилась запись. В качестве ответа укажите ближайшее к времени записи целое число.

решение. Так как частота дискретизации 64 кГц, то за одну секунду запоминается 64000 значений сигнала. Глубина кодирования 24 бита, т. е. 3 байта, это объем памяти, который требуется для хранения данных одного замера сигнала. Объем памяти для хранения данных двух каналов в два раза больше памяти, которая требуется при одноканальной записи.

Для вычисления объема файла в байтах необходимо перемножить количество каналов на глубину звука в байтах на частоту дискретизации и на длительность звучания файла в секундах.

Если требуется найти время, то разделим объем файла на известные параметры звукового файла, переписанные для удобства в виде степени чисел 2, 3 и 5:

Частота дискретизации: 26\*(2\*5)3 = 29\*53 байт

Объем файла: 48 Мбайт = 48\*220 байт = 224\*3 байт.

Количество каналов: 21 .

T = (3\*224) / (29\*53\*3\*21) =(224 / 210)\*(1 / 53) = 214 / 53 секунд.

Для выяснения количества минут, разделим эту величину на 60. заметим, что 60 = 22\*3\*5.

T = 214 : (22\*3\*5\*53) = 212 : (3\*54) = 4096 : 1875 минут.

Эта величина больше числа 2 и меньше 2,5. Округляя до целых, получаем результат 2 минуты.

Ответ: 2.

Контрольные вопросы

1. Что такое дискретизация?

2. Что такое частота дискретизации?

3. Что такое глубина кодирования?

4. Как частота дискретизации и глубина кодирования влияют на качество цифрового звука?

Упражнения для самостоятельного выполнения

1. Звуковая плата производит двоичное кодирование аналогового звукового сигнала. Какое количество информации необходимо для кодирования каждого из 65 536 возможных уровней интенсивности сигнала?

2. Оценить информационный объем цифровых звуковых файлов длительностью 10 секунд при глубине кодирования и частоте дискретизации звукового сигнала, обеспечивающих минимальное и максимальное качество звука:

а) моно, 8 битов, 8000 измерений в секунду;

б) стерео, 16 битов, 48000 измерений в секунду.

3. Производится четырёхканальная (квадро) звукозапись с частотой дискретизации 32 кГц и 32-битным разрешением. Запись длится 2 минуты, её результаты записываются в файл, сжатие данных не производится. Определите приблизительно размер полученного файла (в Мбайт). В качестве ответа укажите ближайшее к размеру файла целое число, кратное 10.

4. Производится двухканальная (стерео) звукозапись с частотой дискретизации 32 кГц и 32-битным разрешением. Результаты записи записываются в файл, сжатие данных не производится; размер полученного файла — 45 Мбайт. Определите приблизительно время записи (в минутах). В качестве ответа укажите ближайшее к времени записи целое число.

5. Производится двухканальная (стерео) звукозапись с частотой дискретизации 32 кГц и 32-битным разрешением. Запись длится 2 минуты, её результаты записываются в файл, сжатие данных не производится. Определите приблизительно размер полученного файла в Мбайт. В качестве ответа укажите ближайшее к размеру файла целое число, кратное 10.

6. Производится двухканальная (стерео) звукозапись с частотой дискретизации 32 кГц и 32-битным разрешением. Результаты записи записываются в файл, сжатие данных не производится; размер полученного файла — 60 Мбайт. Определите приблизительно время записи в минутах. В качестве ответа укажите ближайшее к времени записи целое число.

7. Производится одноканальная (моно) звукозапись с частотой дискретизации 11 кГц и глубиной кодирования 24 бита. Запись длится 7 минут, ее результаты записываются в файл, сжатие данных не производится. Определите объем файла в Мбайтах. В качестве ответа укажите ближайшее к объему файла целое число.

8. Производится одноканальная (моно) звукозапись с частотой дискретизации 16 кГц и 32-битным разрешением. Запись длится 4 минуты, ее результаты записываются в файл, сжатие данных не производится. Определите объем файла в Мбайтах. В качестве ответа укажите ближайшее к объему файла целое число.

9. Производится одноканальная (моно) звукозапись с частотой дискретизации 8 кГц и глубиной кодирования 16 бит. Запись длится 2 минуты, ее результаты записываются в файл, сжатие данных не производится. Определите объем файла в Мбайтах. В качестве ответа укажите ближайшее к объему файла целое число.

10. Производится одноканальная (моно) звукозапись с частотой дискретизации 48 кГц и 16-битным разрешением. Запись длится 2 минуты, ее результаты записываются в файл, сжатие данных не производится. Определите объем файла в Мбайтах. В качестве ответа укажите ближайшее к объему файла целое число.

11. Производится двухканальная (стерео) звукозапись с частотой дискретизации 11 кГц и глубиной кодирования 16 бит. Запись длится 6 минут, ее результаты записываются в файл, сжатие данных не производится. Определите объем файла в Мбайтах. В качестве ответа укажите ближайшее к объему файла целое число.

12. Производится двухканальная (стерео) звукозапись с частотой дискретизации 48 кГц и 24-битным разрешением. Запись длится 1 минуту, ее результаты записываются в файл, сжатие данных не производится. Определите объем файла в Мбайтах. В качестве ответа укажите ближайшее к объему файла целое число.

13. Двухканальная (стерео) звукозапись с частотой дискретизации 16 кГц и 24 битным разрешением велась в течение 16 минут. Сжатие данных не производилось. Определите объем файла в Мбайтах. В качестве ответа укажите ближайшее к объему файла целое число.

14. Производилась двухканальная (стерео) звукозапись с частотой дискретизации 48 кГц и 24-битным разрешением. В результате был получен файл размером 5625 Мбайт, сжатие данных не производилось. Определите приблизительно, сколько времени (в минутах) производилась запись. В качестве ответа укажите ближайшее к времени записи целое число, кратное 5.

15. Производится четырёхканальная (квадро) звукозапись с частотой дискретизации 32 кГц и 24-битным разрешением. Запись длится 2 минуты, её результаты записываются в файл, сжатие данных не производится. Определите объем файла в Мбайтах. В качестве ответа укажите ближайшее к объему файла целое число.

16. Производится двухканальная (стерео) звукозапись с частотой дискретизации 32 кГц и 32-битным разрешением. Запись длится 3 минуты, её результаты записываются в файл, сжатие данных не производится. Определите приблизительно размер полученного файла (в Мбайт). В качестве ответа укажите ближайшее к размеру файла целое число, кратное пяти.

Параграф 5

Кодирование графической информации

Графическая информация может быть представлена в аналоговой и дискретной формах. Примером аналогового представления графической информации может служить живописное полотно, цвет которого изменяется непрерывно, а дискретного - изображение, напечатанное с помощью струйного принтера и состоящее из отдельных точек разного цвета.

Графические изображения из аналоговой (непрерывной) формы в цифровую (дискретную) преобразуются путем пространственной дискретизации. Пространственную дискретизацию изображения можно сравнить с построением изображения из мозаики (большого количества маленьких разноцветных стекол). Изображение разбивается на отдельные маленькие элементы (точки, или пиксели), причем каждый элемент может иметь свой цвет.

Пиксель - минимальный участок изображения, для которого независимым образом можно задать цвет.

В результате пространственной дискретизации графическая информация представляется в виде растрового изображения, которое формируется из определенного количества строк, содержащих, в свою очередь, определенное количество точек.

Важнейшей Характеристикой качества растрового изображения является разрешающая способность. Разрешающая способность растрового изображения определяется количеством точек как по горизонтали, так и по вертикали на единицу длины изображения. Чем меньше размер точки, тем больше разрешающая способность (больше строк растра и точек в строке) и, соответственно, выше качество изображения. Величина разрешающей способности обычно выражается в dpi (dot per inch - точек на дюйм), т. е. в количестве точек в полоске изображения длиной один дюйм (1 дюйм = 2,54 см)

В процессе дискретизации могут использоваться различные палитры цветов, т. е. наборы цветов, в которые могут быть окрашены точки изображения. Каждый цвет можно рассматривать как возможное состояние точки. Количество цветов N в палитре и количество информации I, необходимое для кодирования цвета каждой точки, связаны между собой и могут быть вычислены по формуле:

N=2I

В простейшем случае (черно-белое изображение без градаций серого цвета) палитра цветов состоит всего из двух цветов (черного и белого). Каждая точка экрана может принимать одно из двух состояний - "черная" или "белая", следовательно, по приведенной формуле можно вычислить, какое количество информации необходимо, чтобы закодировать цвет каждой точки:

2 = 2I = 21

Следовательно I = 1 бит.

Количество информации, которое используется для кодирования цвета точки изображения, называется глубиной цвета.

Наиболее распространенными значениями глубины цвета при кодировании цветных изображений являются 4, 8, 16, 24 или 32 бита на точку. Зная глубину цвета, по выше приведенной формуле можно вычислить количество цветов в палитре.

В конце параграфа приведем пример решения задач на кодирование графической информации.

пример. Какой минимальный объём памяти (в Кбайт) нужно зарезервировать, чтобы можно было сохранить любое растровое изображение размером 128X128 пикселей при условии, что в изображении могут использоваться 256 различных цветов? В ответе запишите только целое число, единицу измерения писать не нужно.

решение. Информация о каждом пикселе содержит двоичный код, определяющий цвет пикселя. Для возможности сохранять информацию о различных 256 цветов, потребуется найти наименьшую степень числа 2, чтобы выполнялось неравенство:

2x >= 256

Такой степенью является число 8. Следовательно, для каждого пикселя выделяется объем памяти, равный 8 бит = 1 байт.

Подсчитаем количество пикселей:

128\*128 = 27\*27 = 214 пикселей

Перемножим количество пикселей на объем памяти, занимаемой информацией об одном пикселе:

214\*1 = 214 байт

Переведём объем памяти в килобайты, разделив число байт на 210 :

214 : 210 = 24 Кбайт = 16 Кбайт

Ответ: 16.

Контрольные вопросы

1. Объясните, как с помощью пространственной дискретизации происходит формирование растрового изображения.

2. В каких единицах выражается разрешающая способность растровых изображений?

3. Как связаны между собой количество цветов в палитре и глубина цвета?

Упражнения для самостоятельного выполнения

1. В процессе преобразования растрового графического изображения количество цветов уменьшилось с 65 536 до 16. Во сколько раз уменьшился его информационный объем?

2. Черно-белое (без градаций серого) растровое графическое изображение имеет размер 10 х 10 точек. Какой информационный объем имеет изображение?

3. Цветное (с палитрой из 256 цветов) растровое графическое изображение имеет размер 10 х 10 точек. Какой информационный объем имеет изображение?

4. Какой минимальный объём памяти (в Кбайт) нужно зарезервировать, чтобы можно было сохранить любое растровое изображение размером 512x512 пикселей при условии, что в изображении могут использоваться 256 различных цветов? В ответе запишите только целое число.

5. Какой минимальный объём памяти (в Кбайт) нужно зарезервировать, чтобы можно было сохранить любое растровое изображение размером 1024x1024 пикселей при условии, что в изображении могут использоваться 16 различных цветов? В ответе запишите только целое число, единицу измерения писать не нужно.

6. Какой минимальный объём памяти (в Мбайт) нужно зарезервировать, чтобы можно было сохранить любое растровое изображение размером 2048x1024 пикселей при условии, что в изображении могут использоваться 16 различных цветов? В ответе запишите только целое число, единицу измерения писать не нужно.

7. Какой минимальный объём памяти (в Кбайт) нужно зарезервировать, чтобы можно было сохранить любое растровое изображение размером 320x640 пикселей при условии, что в изображении могут использоваться 256 различных цветов? В ответе запишите только целое число, единицу измерения писать не нужно.

8. Для хранения произвольного растрового изображения размером 1024X1024 пикселей отведено 512 Кбайт памяти, при этом для каждого пикселя хранится двоичное число — код цвета этого пикселя. Для каждого пикселя для хранения кода выделено одинаковое количество бит. Сжатие данных не производится. Какое максимальное количество цветов можно использовать в изображении?