

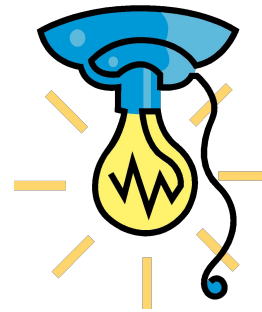
# Кодирование информации



# Кодирование информации

Сигналы, используемые для передачи информации:

- световые;
- звуковые;
- тепловые;
- электрические;
- в виде жеста;
- в виде движения;
- в виде слова и т. д.



Для того чтобы передача информации была успешной, приёмник должен не только получить сигнал, но и расшифровать его. Необходимо заранее договариваться, как понимать те или иные сигналы, т.е. требуется разработка **кода**.

# Кодирование информации

## Разнообразии используемых нами кодов

Нотные знаки кодируют музыкальные произведения:



Правила дорожного движения кодируются специальными знаками:



3.1



3.2



1.20



1.21

Свой код из шести цифр (индекс) имеет каждый населенный пункт:



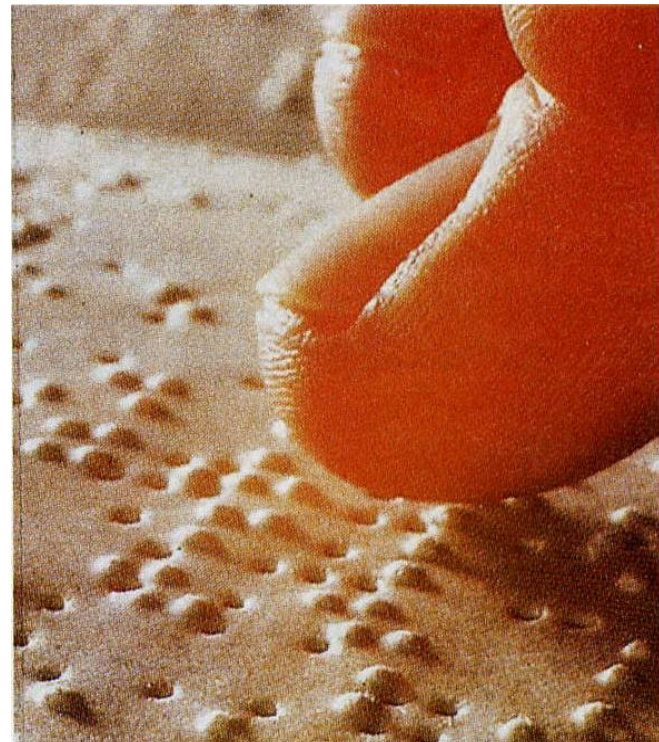
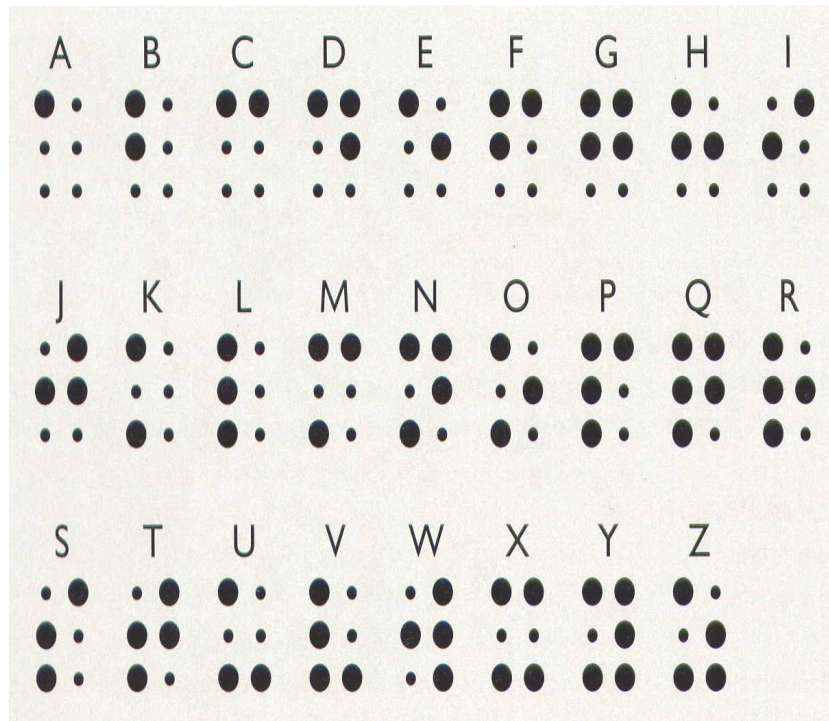
Товары маркируются специальными кодами:



# Кодирование информации

## Разнообразие используемых нами кодов

В середине XIX века французский педагог Луи Брайль придумал специальный способ представления информации для незрячих людей.






# Кодирование информации


## Способы кодирования

Одна и та же информация может быть представлена разными кодами.

Существуют **три** основных способа кодирования информации:

 **Графический** – с помощью рисунков и значков;

 **Числовой** – с помощью чисел;

 **Символьный** – с помощью символов того же алфавита, что и исходный текст.

# Кодирование информации

## Числовое кодирование

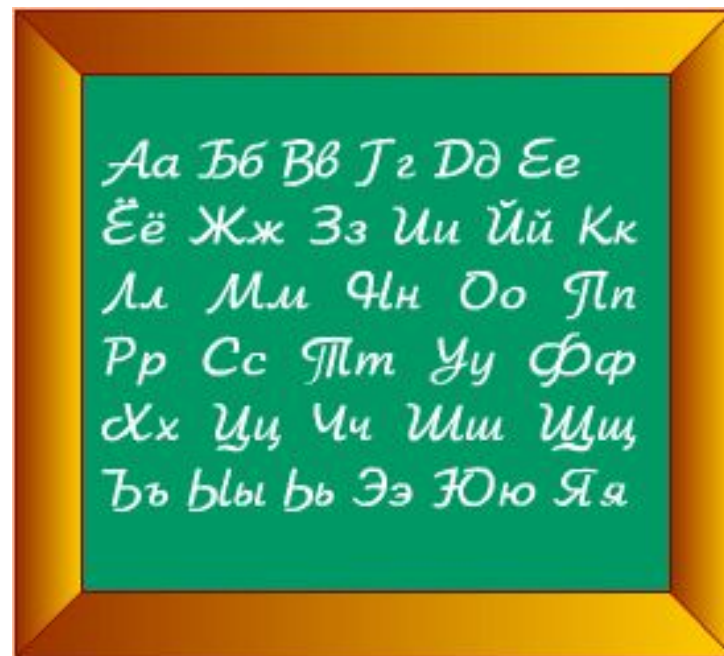
В алфавите любого разговорного языка буквы следуют друг за другом в определенном порядке. Это дает возможность присвоить каждой букве алфавита ее порядковый номер.

Например, числовое  
сообщение

**01112001030918**

соответствует слову

**АЛФАВИТ**



# Кодирование информации

## Символьное кодирование

Смысл этого способа заключается в том, что символы алфавита (буквы) заменяются символами (буквами) того же алфавита по определенному правилу.

Например,  $a \rightarrow б$ ,  $б \rightarrow в$ ,  $в \rightarrow г$  и т.д. Тогда слово **АЛФАВИТ** будет закодировано последовательностью **БМХБГЙУ**.

# Кодирование информации

## Графическое кодирование

Это кодирование информации при помощи разнообразных рисунков или значков:

а	б	в	г
д	е	ж	з
и	й	к	л
м	н	о	п
р	с	т	у
ф	х	ц	ч
ш	щ	ы	ь
э	ю	я	
пробел	·	точка	→



А	●■■
Б	■■■■●
В	■■■■■
Г	■■■■●
Д	■■■■■
Е	●■■■■
Ж	■■■■■
З	■■■■●
И	■■■■■
К	■■■■■
Л	■■■■●
М	■■■■■
Н	■■■■■
О	■■■■■

П	■■■■●
Р	■■■■■
С	■■■■■
Т	■■■■■
У	■■■■■
Ф	■■■■■
Х	■■■■■
Ц	■■■■■
Ч	■■■■■
Ш	■■■■■
Щ	■■■■■
Э	■■■■■
Ю	■■■■■
Я	■■■■■

Ъ	■■■■■
Ы	■■■■■
И	■■■■■
1	■■■■■
2	■■■■■
3	■■■■■
4	■■■■■
5	■■■■■
6	■■■■■
7	■■■■■
8	■■■■■
9	■■■■■
0	■■■■■



# Кодирование информации

## Графическое кодирование ВМФ России

Специальные сигнальные флаги появились в России ещё в 1696 г. В СССР существовали 32 буквенных, 10 цифровых флагов, 4 дополнительных и 13 специальных флагов. Эта же система с незначительными изменениями используется в ВМФ России.

А	Б	В	Г	Д	Е	Ж
З	И	И	К	Л	М	Н
О	П	Р	С	Т	У	Ф
Х	Ц	Ч	Ш	Щ	Ъ	Ы
Ь	Э	Ю	Я			

# Общие вопросы кодирования информации

Задача кодирования информации представляется как некоторое преобразование числовых данных в заданной системе счисления.

Двоично-десятичный код				Десятичный код
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9

Так как любая позиционная система не несет в себе избыточности информации и все кодовые комбинации являются разрешенными, **использовать такие системы для контроля правильности передачи не представляется возможным.**

# Общие вопросы кодирования информации

*Систематический код* - код, содержащий в себе кроме информационных еще и контрольные разряды.

**ВАСЯ**  $\Rightarrow$  **0101 00 1001 00 1011 10 0001 01**

В контрольные разряды записывается некоторая информация об исходном числе. Поэтому можно говорить, что систематический код обладает **избыточностью**. При этом абсолютная избыточность будет выражаться количеством контрольных разрядов  $k$ , а относительная избыточность - отношением  $k/n$ , где  $n = m + k$  - общее количество разрядов в кодовом слове ( $m$  - количество информационных разрядов).

**Абсолютная избыточность = 2, относительная избыточность 2/6**

# Общие вопросы кодирования информации

Понятие корректирующей способности кода обычно связывают с возможностью обнаружения и исправления ошибки. Количественно корректирующая способность кода определяется вероятностью обнаружения или исправления ошибки. Если имеем  $n$ -разрядный код и вероятность искажения одного символа  $p$ , то вероятность того, что искажены  $k$  символов, а остальные  $n - k$  символов не искажены, по теореме умножения вероятностей будет

$$w = p^k(1-p)^{n-k}.$$

Число кодовых комбинаций, каждая из которых содержит  $k$  искаженных элементов, равна числу сочетаний из  $n$  по  $k$ :

$$C_n^k = \frac{n!}{k!(n-k)!}$$

Тогда полная вероятность искажения информации

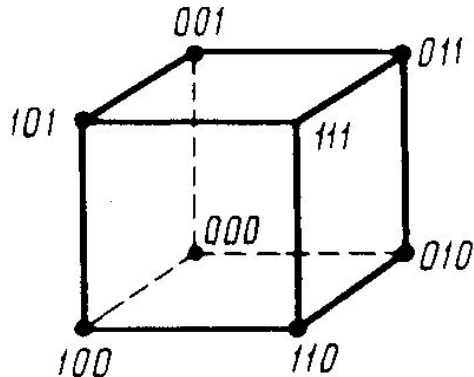
$$P_{\Sigma} = \sum_{i=1}^k \frac{n!}{i!(n-i)!} p^i (1-p)^{n-i}$$

# Общие вопросы кодирования информации

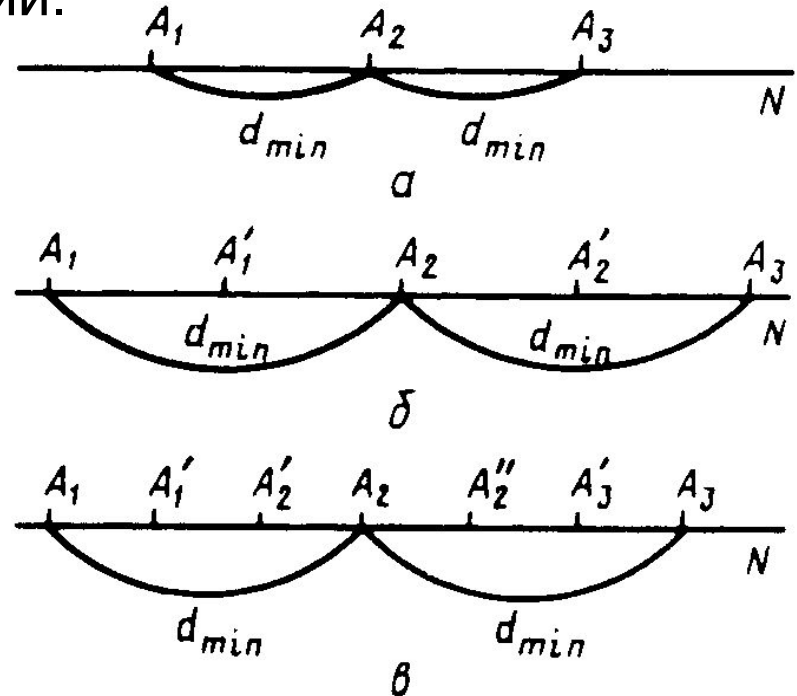
Кодовое расстояние  $d(A, B)$  для кодовых комбинаций  $A$  и  $B$  определяется как вес третьей кодовой комбинации, которая получается поразрядным сложением исходных комбинаций по модулю 2 (операция XOR) или количество различающихся разрядов в двух кодовых комбинациях.

Вес кодовой комбинации  $V(A)$  - количество единиц, содержащихся в кодовой комбинации.

*Кодовое расстояние = сумма длин ребер между соответствующими вершинами куба*



Геометрическое представление кодов



Кодовые расстояния



# Общие вопросы кодирования

В тех случаях, когда необходимо не только обнаружить ошибку, но и исправить ее (т. е. указать место ошибки), минимальное кодовое расстояние должно быть

В теории кодирования  $d_{\min} \geq 2t + 1$  показано, что систематический код способен обнаружить ошибки только когда минимальное кодовое расстояние для него больше или равно  $2t$ , т.е.

$$d_{\min} \geq 2t,$$

где  $t$  - кратность обнаруживаемых ошибок (в случае одиночных ошибок  $t = 1$  и т. д.).

Это означает, что между соседними разрешенными кодовыми словами должно существовать по крайней мере одно кодовое слово

# Общие вопросы кодирования

Допустим, имеется информация набор кодовых комбинаций:

0 0 0

0 0 1

0 1 0

0 1 1

1 0 0

1 0 1

1 1 0

1 1 1

Геометрическая модель этого кода – куб. Для рассматриваемого кода  $d_{\min} = 1$ . Учитывая, что рассматриваемый код по построению является избыточным, можно утверждать, что **любой избыточный код имеет  $d_{\min} = 1$  и наоборот, если  $d_{\min} = 1$ , код является избыточным.**

# Общие вопросы кодирования

Для построения **информации** избыточного кода, который может обнаруживать одну ошибку, нужно отобрать рабочие комбинации на расстоянии  $d(A, B) \geq 2$ .

В рассматриваемом коде можно выбрать следующие комбинации:

$$\left. \begin{array}{l} 010 \\ 100 \\ 111 \\ 001 \end{array} \right\} M_p = 4$$

где  $M_p$  – число разрешенных (или рабочих) комбинаций.

**Избыточность полученного кода**

$$R = k/n = (n-m)/n = 1/3 \approx 33\%.$$

# Общие вопросы кодирования

Если требуется обнаружить две ошибки, то рабочих комбинаций будет только две, например

$$\left. \begin{array}{l} 000 \\ 111 \end{array} \right\} M_p = 2$$

$$\left. \begin{array}{l} 010 \\ 101 \end{array} \right\} M_p = 2$$

минимальное кодовое расстояние в этом случае  $d_{\min} = 3$ , избыточность

$$R = k/n = (n-m)/n = 2/3 \approx 67\%.$$

Если требуется обнаруживать три ошибки,  $d_{\min} \geq 4$ , что невозможно обеспечить в рассматриваемом коде, так как кодовое расстояние  $d(A, B) \leq 3$ .

# Общие вопросы кодирования

## информации Возможны несколько стратегий борьбы с

### ошибками:

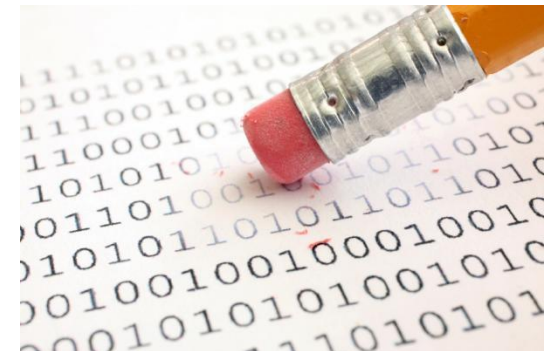
обнаружение  
ошибок в блоках  
данных и  
автоматический  
запрос повторной  
передачи  
повреждённых  
блоков



обнаружение  
ошибок в блоках  
данных и  
отбрасывание  
повреждённых  
блоков (потокковые  
мультимедиа-  
системы)



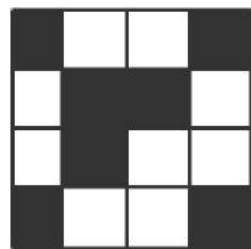
исправление  
ошибок



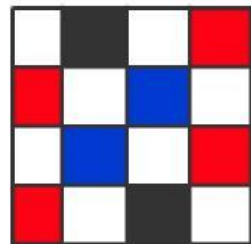


# Общие вопросы кодирования информации

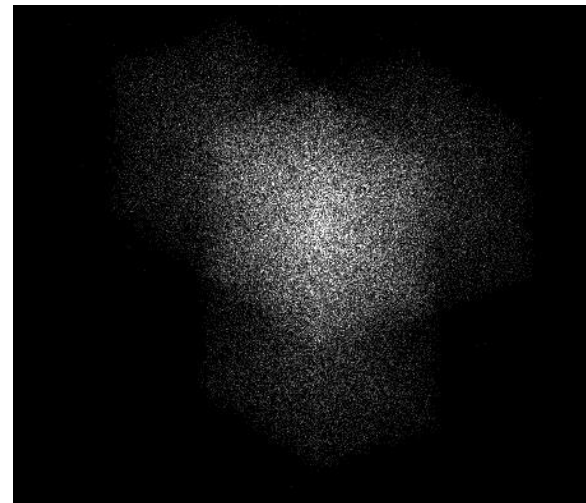
Эффективное кодирование базируется на основной теореме Шеннона для каналов без шума, в которой доказано, что сообщения, составленные из букв некоторого алфавита, можно закодировать так, что среднее число двоичных символов на букву будет сколь угодно близко к энтропии источника этих сообщений, но не меньше этой величины.



1 0 0 1  
0 1 1 0  
0 1 0 0  
1 0 0 1



00 11 00 01  
01 00 10 00  
00 10 00 01  
01 00 11 00



# Эффективное кодирование информации

## Методика кодирования Шеннона-Фано

Код строится следующим образом: буквы алфавита сообщений выписываются в таблицу в порядке убывания вероятностей. Затем они разделяются на две группы так, чтобы суммы вероятностей в каждой из групп 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 останется по одной букве.

Буквы	Вероятности	Кодовые комбинации
$z_1$	0,22	11
$z_2$	0,20	101
$z_3$	0,16	100
$z_4$	0,16	01
$z_5$	0,10	001
$z_6$	0,10	0001
$z_7$	0,04	00001
$z_8$	0,02	00000

$$H(z) = -\sum_{i=1}^8 p(z_i) \log p(z_i) \approx 2,76$$

Среднее число символов на букву

$$l_{cp} = -\sum_{i=1}^8 p(z_i) n(z_i) \approx 2,84,$$

# Эффективное кодирование информации

## Методика кодирования Шеннона-Фано

Множество вероятностей в предыдущей таблице можно было разбить иным образом:

Буквы	Вероятности	Кодовые комбинации
$z_1$	0,22	11
$z_2$	0,20	10
$z_3$	0,16	011
$z_4$	0,16	010
$z_5$	0,10	001
$z_6$	0,10	0001
$z_7$	0,04	00001
$z_8$	0,02	00000

Буквы	Вероятности	Кодовые комбинации
$z_1$	0,22	11
$z_2$	0,20	101
$z_3$	0,16	100
$z_4$	0,16	01
$z_5$	0,10	001
$z_6$	0,10	0001
$z_7$	0,04	00001
$z_8$	0,02	00000

Предыдущий вариант

$$H(z) = - \sum_{i=1}^8 p(z_i) \log p(z_i) \approx 2,76$$

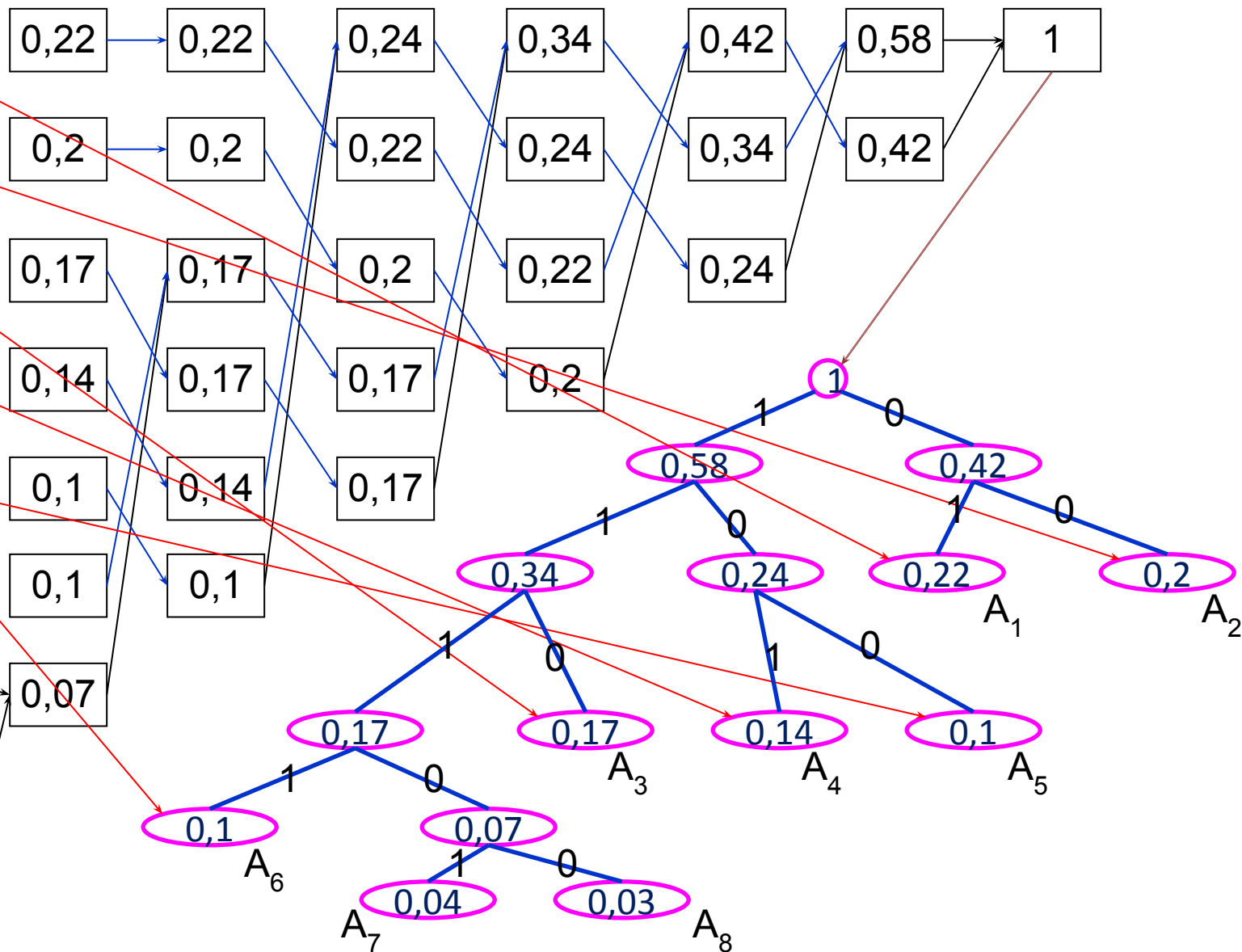
Среднее число символов на букву

$$l_{cp} = - \sum_{i=1}^8 p(z_i) n(z_i) \approx 2,8$$

# Эффективное кодирование информации

## Методика кодирования Хаффмана (Huffman)

A	P
A <sub>1</sub>	0,22 <b>01</b>
A <sub>2</sub>	0,2 <b>00</b>
A <sub>3</sub>	0,17 <b>110</b>
A <sub>4</sub>	0,14 <b>101</b>
A <sub>5</sub>	0,1 <b>100</b>
A <sub>6</sub>	0,1 <b>1111</b>
A <sub>7</sub>	0,04 <b>11101</b>
A <sub>8</sub>	0,03 <b>11100</b>



# Эффективное кодирование информации

## Методика кодирования по четности-нечетности

Если в математическом коде выделен один контрольный разряд ( $k = 1$ ), то к каждому двоичному числу добавляется один избыточный разряд и в него записывается 1 или 0 с таким условием, чтобы сумма цифр в каждом числе была по модулю 2 равна 0 для случая четности или 1 для случая нечетности. Появление ошибки в кодировании обнаружится по нарушению четности (нечетности). При таком кодировании допускается, что может возникнуть только одна ошибка.

Число	Контрольный разряд	Проверка
10101011	1	0
11001010	0	0
10010001	1	0
11001011	0	1-нарушение

Такое кодирование имеет минимальное кодовое расстояние, равное 2.



# Эффективное кодирование информации

## Методика кодирования по четности-нечетности

Можно представить и несколько видоизмененный способ контроля по методу четности - нечетности. Длинное число разбивается на группы, каждая из которых содержит  $l$  разрядов. Контрольные разряды выделяются всем группам по строкам и по столбцам согласно следующей схеме:

$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$k_1$
$a_6$	$a_7$	$a_8$	$a_9$	$a_{10}$	$k_2$
$a_{11}$	$a_{12}$	$a_{13}$	$a_{14}$	$a_{15}$	$k_3$
$a_{16}$	$a_{17}$	$a_{18}$	$a_{19}$	$a_{20}$	$k_4$
$a_{21}$	$a_{22}$	$a_{23}$	$a_{24}$	$a_{25}$	$k_5$
$k_6$	$k_7$	$k_8$	$k_9$	$k_{10}$	

Проверка

$$\sum_i (a_i + k_i)$$

# Эффективное кодирование информации

## Методика кодирования Хэмминга

Предположим, что имеется код, содержащий  $m$  информационных разрядов и  $k$  контрольных разрядов. Запись на  $k$  позиций определяется при проверке на четность каждой из проверяемых  $k$  групп информационных символов. Пусть было проведено  $k$  проверок. Если результат проверки свидетельствует об отсутствии ошибки, то запишем 0, если есть ошибка, то запишем 1. Запись полученной последовательности символов образует двоичное, контрольное число, указывающее номер позиции, где произошла ошибка. При отсутствии ошибки в данной позиции последовательность будет содержать только нули. Полученное таким образом число описывает  $n = (m + k + 1)$  событий. Следовательно, справедливо неравенство

$$2^k \geq (m + k + 1).$$

**Пример кода:**  $\underbrace{1001011001}_m \underbrace{0100}_k$   $2^k \geq (m + k + 1) \Rightarrow 16 \geq 15$

# Эффективное кодирование информации

## Методика кодирования Хэмминга

Полученное таким образом число описывает  $n = (m + k + 1)$  событий. Следовательно, справедливо неравенство

$$2^k \geq (m + k + 1).$$

Определить максимальное значение  $m$  для данного  $k$  можно из следующего:

$n...$	1	2	3	4...	8...	15	16...	31	32...	63	64
$m...$	0	0	1	1...	4...	11	11...	26	26...	57	57
$k...$	1	2	2	3...	4...	4	5...	5	6...	6	7

**Пример кода:** **1001011001** **0100**  $2^k \geq (m + k + 1) \Rightarrow 16 \geq 15$   
**m** **k**

# Эффективное кодирование информации

## Методика кодирования Хэмминга

Определим теперь позиции, которые надлежит проверить в каждой из  $k$  проверок. Если в кодовой комбинации ошибок нет, то контрольное число содержит только нули. Если в первом разряде контрольного числа стоит 1, то, значит, в результате первой проверки обнаружена ошибка. Имея таблицу двоичных эквивалентов для десятичных чисел, можно сказать, что, например, первая проверка охватывает позиции 1, 3, 5, 7, 9 и т. д., вторая проверка — позиции 2, 3, 6, 7, 10.

Проверка    Проверяемые разряды

1...	1,3,5,7,9,11,13,15...
2...	2,3,6,7, 10, 11, 14, 15, 18, 19,22,23...
3...	4, 5, 6, 7, 12, 13, 14, 15, 20, 21, 22, 23...
4...	8,9,10,11,12,13,14,15,24...

# Эффективное кодирование информации

## Методика кодирования Хэмминга

Кодирование информации по методу Хэмминга для 7-миразрядного кода  $n=7$ ,  $m=4$ ,  $k=3$  и контрольными будут разряды 1, 2, 4

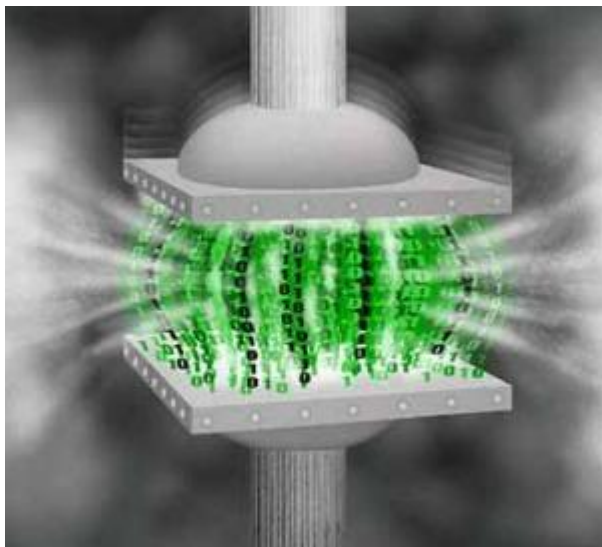
Разряды двоичного кода							Кодируемая десятичная информация	Проверка	Проверяемые разряды
1	2	3	4	5	6	7			
$k_1$	$k_2$	$m_1$	$k_3$	$m_2$	$m_3$	$m_4$			
0	0	0	0	0	0	0	0		
1	1	0	1	0	0	1	1		
0	1	0	1	0	1	0	2		
1	0	0	0	0	1	1	3		
1	0	0	1	1	0	0	4	000	
0	1	0	0	1	0	1	5		
1	1	0	0	1	1	0	6		
0	0	0	1	1	1	1	7		
1	1	1	0	0	0	0	8		
0	0	1	1	1	0	1	9	$101_2 = 5_{10}$	
1	0	1	1	0	1	0	10		
0	1	1	0	0	1	1	11		
0	1	1	1	1	0	0	12		
1	0	1	0	1	0	1	13		
0	0	1	0	1	1	0	14		
1	1	1	1	1	1	1	15		

# Эффективное кодирование информации

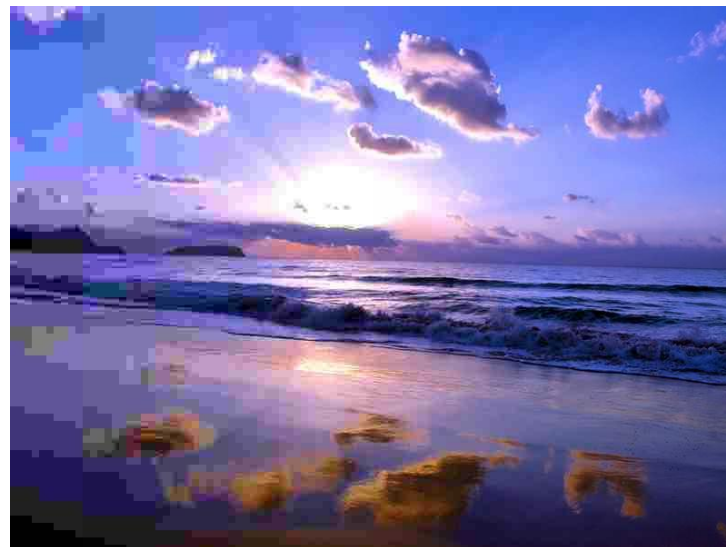
## Алгоритмы сжатия информации

Обычно выделяют два класса алгоритмов сжатия

### Сжатие без



### Сжатие с



# Эффективное кодирование информации

## Алгоритмы сжатия информации

Классический алгоритм Лемпела-Зива – LZ77, названный так по году своего опубликования, предельно прост. Он формулируется следующим образом : "если в прошедшем ранее выходном потоке уже встречалась подобная последовательность байт, причем запись о ее длине и смещении от текущей позиции короче чем сама эта последовательность, то в выходной файл записывается ссылка (смещение, длина), а не сама последовательность".

"КОЛОКОЛ\_ОКОЛО\_КОЛОКОЛЬНИ"

"КОЛО(-4,3)\_(-5,4)О\_(-14,7)ЬНИ"

Алгоритм RLE (англ. Run Length Encoding)

"AAAAAAAAA"

"(A,7)"

