Основы теории информации и кодирования. Измерение информации. Кодирование. Форматы файлов

Александра Игоревна Кононова: +7-977-977-97-29 (WhatsApp),
gitlab.com/illinc/raspisanie

МИЭТ

14 сентября 2025 г. — актуальную версию можно найти на
https://gitlab.com/illinc/otik

Регламент ОТИК (аналогичен ОЭВМ [и Асм] + семинары)

1. Материалы: `https://gitlab.com/illinc/otik`
   - регламент: `https://gitlab.com/illinc/otik/-/raw/master/otik-labs.pdf`
   - баллы за семинары и посещаемость семинаров выставляются на 9 и 15/17 неделях.
2. Поскольку у нас очное обучение — **все вопросы задаются очно на консультациях,** письменно преподаватель вам не отвечает.
3. Консультации Кононовой А. И. (плановые и дополнительные): `https://gitlab.com/illinc/raspisanie/-/issues/4`
4. Итоговая оценка ОТИК выставляется после экзамена (аналогично ОЭВМ [и Асм]):
   - досрочно оценки выставляет только Кононова А. И.;
   - по расписанию экзамена — все преподаватели, кто назначен в расписании на этот экзамен.

Предмет теории информации. Источник информации

Теория информации (ТИ) — математическая теория, посвящённая измерению информации, её потока, «размеров» канала связи и т. п., особенно применительно к средствам связи:

$$x \leftrightarrow X \quad \sim \quad I(x)$$

x — сообщение, $X = \{x, p(x)\}$ — источник (случайный процесс/случайная величина).

Дискретное x может состоять из символов или быть отдельным символом.

Информация — нематериальная сущность, при помощи которой с любой точностью можно описывать реальные (материальные), виртуальные (возможные) и понятийные сущности.

$I(x)$:

1. **Новизна** (неизмеряемость в быту).
2. **Объёмный** (длина — измерение в технике).
3. **Вероятностный** (снятая неопределённость — измерение в ТИ).

данные $\supseteq$ информация $\supset$ знания,

ОТИК: информация = данные + источник

Виды источников информации

1. Все → **дискретные (цифровые)**/непрерывные (аналоговые); дискретные → **качественные**/количественные.

   Элемент качественной информации — **символ** $a \in A$ (множество A — алфавит);
   конечная последовательность символов — **слово** $x \in A^+$ (строка, фраза).

2. Источники символов алфавита A (можно прочитать строку; дискретные качественные) →
   - **стационарные** — вероятность символа не зависит от времени/позиции: только от контекста;
   - нестационарные — при сдвиге вероятности меняются;

   стационарные $\supset$ марковские (М) $\supset$ стационарные без памяти (БП) $\ni$ равновероятный (РВ)

   1. **марковский** источник — а) вероятность символа определяется состоянием; б) состояние изменяется после порождения символа (новое состояние однозначно определяется предыдущим и порождённым символом); марковский источник порядка m — вероятность символа на i-м шаге зависит от m предыдущих символов: $i-1, i-2, \ldots, i-m$;
   2. **стационарный источник без памяти** — вероятность символа $a \in A$ постоянна (равна $p(a)$);
   3. **равновероятный источник** — вероятность символа $a \in A$ постоянна и одинакова для всех ($p(a) = \frac{1}{|A|}$).

   Источник без памяти может быть и нестационарным: $p(a, i)$! Но в этом курсе БП только стационарные.
   Для нестационарного с глубиной памяти m иногда используется термин «нестационарный марковский».

Энтропия и информация

1865–1901, статистическая термодинамика: мера уравновешенности — энтропия $H = k \cdot \ln N$

Средняя длина кода $code$ зависит от источника:

$$\left|code(X)\right| = \sum_{x \leftarrow\!\shortmid X} p_X(x) \cdot \left|code(x)\right|$$

1921, Роналд Фишер: для каждого источника X есть предел сжатия

$$\exists \inf_{code} |code(X)|$$

Для РВ X с $N = 2^r$ состояниями предел r бит

1928, Ральф Хартли:

а) назовём этот предел $I(X)$;

б) для РВ X с N состояниями $I(X) = \log_2 N = -\log_2 \frac{1}{N}$ бит

Для РВ X с N состояниями $\frac{1}{N} = p_X(x)$

1948, Клод Шеннон:

а) пусть $I_X(x) = -\log_2 p_X(x)$ бит;

б) усредняем по X как длину: $I(X) = \sum_{x \leftarrow\!\shortmid X} p_X(x) \cdot I_X(x)$

в) такая $I(X)$ действительно является $\inf_{code} |code(X)|$ (первая теорема Шеннона)

Энтропия и информация

1865 г. — Рудольф Клаузиус ввёл в статистическую физику понятие **энтропии** — меры уравновешенности [Дж/К].

1877 г. — Людвиг Больцман установил связь энтропии с вероятностью.

1901 г. — Макс Планк определил энтропию как

$H = k \cdot \ln(\Omega)$, где k — коэффициент Больцмана [Дж/К].

1921 г. — Роналд Фишер ввёл термин «информация» (информация, которую можно извлечь из имеющихся данных, **имеет предел**).

1928 г. — Ральф Хартли — логарифмическая мера информации для **равновероятных** событий.

1948 г. — Клод Шеннон — вычисление количества информация и энтропии.

Основное соотношение между энтропией и информацией:

$$I + \frac{\log_2 e}{k} H = const \quad \text{[бит]} \qquad \left(\frac{dI}{dt} = -\frac{\log_2 e}{k} \frac{dH}{dt} \quad \text{[бит/с]} \right).$$

Единица измерения информации

Бит — количество информации в сообщении, уменьшающем неопределённость знания в два раза.

Источник с двумя равновероятными состояниями — симметричная монета

.	?	2 возможных варианта
P	Решка	1 вариант

Неопределённость уменьшилась в 2 раза: $I(\mathtt{P}) = 1$ бит

..	Две симметричные монеты	
O.	Первая — вверх орлом	2 раза (+1 бит)
OP	Вторая — вверх решкой	2 раза (+1 бит)
	4 возможных варианта	$I(\mathtt{OP}) = 2$ бита

Другая единица — **трит:** троичный разряд; количество информации в сообщении, уменьшающем неопределённость в три раза; симметричная игральная кость D3.

1 трит $= \log_2(3) \approx 1{,}6$ бита, $\quad 1$ бит $= \log_3(2) \approx 0{,}6$ трита.

Бит используется чаще трита только из-за двоичности базы ЭВМ, а не из-за свойств бита.

Кратные единицы для бита

Три бита — **триада,** четыре бита — **тетрада** (nibble), восемь бит — **октет.**

Байт — для ЭВМ общего назначения октет, для МК и ЦСП от 6 до 16 битов (от 4 до 64).

Символ кодирования — при $(k \geqslant 6)$-битном байте **всегда символ=байт.**
Для 4-битных ЭВМ символ кодирования — часто октет: удобные символы от 6 до 9 бит.

Для наглядности примем, что на доске в байте $k = 3$ бита и символ=байт=триада
(иногда будем считать $k = 4$ и символ=байт=тетрада).

Неоднозначность двоичных и десятичных приставок для байта и бита

IEEE 1541-2002 / IEC 60027-2:2005			ГОСТ 8.417-2002		
kibibyte	KiB	1024 byte	килобайт	КБ	1024 байт
kibibit, kibit	Kib	1024 bit	Для этих величин в РФ утверждённых обозначений нет		
kilobyte	KB	1000 byte			
kilobit	Kb	1000 bit	килобит	Кб	1000 бит

Требования к мере информации $I(x)$

1. $I(x) \geqslant 0$.
2. Вероятностный подход: $I(x) = f(p_x)$.
3. Объёмный подход: $I(x)$ монотонно связана с затратами на передачу
   - два равновероятных сообщения — 0 и 1 (1 бит),
     четыре — 00, 01, 10, 11 (2 бита) и т. д.:
     $f\left(\frac{1}{2}\right) = 1, \;\; f\left(\frac{1}{4}\right) = 2, \;\; f\left(\frac{1}{8}\right) = 3, \ldots$
   - затраты на передачу независимых сообщений складываются:
     $I(x_1, \ldots, x_n) = I(x_1) + \ldots + I(x_n)$

     при этом вероятности независимых событий умножаются
     $f(p_1 \times \ldots \times p_n) = f(p_1) + \ldots + f(p_n)$.

Формула Хартли для равновероятных событий

Источник X порождает N **равновероятных** сообщений x ($\forall x \leftarrow X:\ p_X(x) = p = \frac{1}{N}$).

$$I_X(x) = I(X) = I = \log_2 N = -\log_2(p) \text{ битов}, \quad \text{или } 2^I = N;$$
$$I = \log_3 N = -\log_3(p) \text{ тритов}, \quad \text{или } 3^I = N;$$
$$I = \log_{256} N = -\log_{256}(p) \text{ октетов} = \frac{\log_2 N}{8} = \frac{I\ [\text{бит}]}{8} \text{ октетов}.$$

где $I_X(x)$ — количество информации в сообщении x;
$I(X)$ — **среднее** кол-во информации в одном сообщении источника X.

Если $N = 2$, то $I = 1$ бит. Если $N = 3$, то $I = 1$ трит.

Подбрасывание монеты

..	4 варианта	2 бита

Угадывание слов по словарю

.....	175 слов	7,5 бит
.а.и.а	122 слова	6,9 бит
р.б.т.	4 слова	2 бита

Формула Шеннона для неравновероятных событий

Количество информации I в сообщении с вероятностью $p_X(x)$:

$$I_X(x) = -\log_2 p_X(x) \text{ битов } = -\log_3 p_X(x) \text{ тритов}$$

Свойства:

1. Неотрицательность: $I_X(x) \geqslant 0, x \leftarrow\!\!\rightarrow X$.
2. Монотонность: $x_1, x_2 \leftarrow\!\!\rightarrow X, p_X(x_1) \geqslant p_X(x_2) \rightarrow I_X(x_1) \leqslant I_X(x_2)$.
3. Аддитивность: для независимых сообщений $x_1, \ldots, x_n$ $I_X(x_1, \ldots, x_n) = \sum_{i=1}^{n} I_X(x_i)$
4. Для равновероятных событий соответствует формуле Хартли.

Среднее количество информации дискретного источника $X = \{x, p_X(x)\}$:

$$I(X) = \sum_{x_i \leftarrow\!\!\rightarrow X} \Big(p_X(x_i) \cdot I_X(x_i) \Big) = - \sum_{x_i \leftarrow\!\!\rightarrow X} \Big(p_X(x_i) \cdot \log_2 p_X(x_i) \Big) \text{ битов}$$

Количество информации в тексте

Из источника символов X можно прочитать текст $\vec{x} = x_1 x_2 \dots x_k$

1. Источник без памяти: сообщения $x_1, x_2, \dots x_k$ независимы
   $p(\vec{x}) = p(x_1) \cdot p(x_2) \cdot \dots \cdot p(x_k)$
   $I(\vec{x}) = I(x_1) + I(x_2) + \dots + I(x_k)$

2. Источник с памятью:
   $p(\vec{x}) = p(x_1) \cdot p(x_2|x_1) \cdot \dots \cdot p(x_k|x_1 x_2 \dots x_{k-1})$
   $I(\vec{x}) = I(x_1) + I(x_2|x_1) + \dots + I(x_k|x_1 x_2 \dots x_{k-1})$

   Если источник марковский порядка m:
   $I(\vec{x}) = I(x_1) + \dots + I(x_i|x_{i-m} \dots x_{i-1}) + \dots + I(x_k|x_{k-m} \dots x_{k-1})$

Модель источника: X неизвестен

Оценка алфавита A_1 и вероятностей источника по сообщению: $x =$ **«молоко»**

1. A_1 — koi-8, равновероятные символы (РВ): $p = \frac{1}{256}$, $I(x) = 6 \cdot \log_2(256) = 48$ бит
2. A_1 — русский алфавит, РВ: $p = \frac{1}{33}$, $I(x) = 6 \cdot \log_2(33) \approx 30{,}3$ бита
3. A_1 — Unicode 12.1, РВ: $p = \frac{1}{137\,994}$, $I(x) \approx 6 \cdot 17{,}1 \approx 102{,}4$ бита
4. $A_1 = \{\text{к}, \text{л}, \text{м}, \text{о}\}$, РВ: $p = \frac{1}{4}$, $I(x) = 6 \cdot \log_2(4) = 12$ бит
5. $A_1 = \{\text{к}, \text{л}, \text{м}, \text{о}\}$ или koi-8, неравновероятные, стационарный источник без памяти (БП):
   о (3) + к (1) + л (1) + м (1): $p(\text{о}) = \frac{3}{6}$, $p(\text{к}) = p(\text{л}) = p(\text{м}) = \frac{1}{6}$
   $I(x) = -3 \cdot \log_2(\frac{3}{6}) - \log_2(\frac{1}{6}) - \log_2(\frac{1}{6}) - \log_2(\frac{1}{6}) = 3 \cdot \log_2(2) + 3 \cdot \log_2(6) \approx 10{,}8$ бита
6. $A_1 = \{\text{к}, \text{л}, \text{м}, \text{о}\}$ или koi-8, марковский источник первого порядка (М1) с вероятностями:

предыдущий	p(к)	p(л)	p(м)	p(о)
–	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
к, л, м	0	0	0	1
о	$\frac{1}{2}$	$\frac{1}{2}$	0	0

$$I(x) = -\log_2(\tfrac{1}{4}) - \log_2(1) - \\ -\log_2(\tfrac{1}{2}) - \log_2(1) - \\ -\log_2(\tfrac{1}{2}) - \log_2(1) = \\ = 2 + 1 + 1 = 4 \text{ бита}$$

7. $A_1 = \{\text{молоко}, \text{чай}\}$, РВ: $p = \frac{1}{2}, I(x) = 1$ бит
8. $A_1 = \{\text{молоко}\}$: $p = 1, I(x) = 0$ бит

Задачи (равновероятный источник)

1. Найти количество информации в событии «три разные симметричные монеты выпали все вверх решкой».
2. Найти количество информации в источнике «три разные симметричные монеты».

Задачи (стационарный источник без памяти)

1. Найти количество информации в событии «две из трёх неразличимых симметричных монет выпали вверх решкой, третья — орлом».
2. Найти количество информации в источнике «три неразличимые симметричные монеты».

3. Найти количество информации в событии «из урны с 3 белыми и 5 чёрными шарами извлекли чёрный шар».
4. Найти количество информации в событии «из урны с 3 белыми и 5 чёрными шарами извлекли белый шар».
5. Найти количество информации в источнике «урна с 3 белыми и 5 чёрными шарами».

Задачи (стационарный источник с памятью)

1. Источник X генерирует последовательность подстрок «хрю» и «мяу» (с равной вероятностью), не разделяя их (например, «хрюхрюхрюмяухрюмяумяухрюмяумяу...»). Из случайного места последовательности (не обязательно с начала подстроки) читается три символа подряд (сообщение x).

   Найти количество информации в событии «$x =$ рюх».

2. Источник X аналогично генерирует посл-ть «ку» и «кукареку» (например, «кукукукукарекукукукарекукукарекукукукарекукукареку...»). Из случайного места посл-ти читается два (три) символа подряд (x).

   Найти количество информации в событиях:

   - $x =$ ка;
   - $x =$ кар;
   - $x =$ ку;
   - $x =$ ук;
   - $x =$ ек.

Подсказка: основная проблема в том, что часть символов — одинаковые. Пусть они разные...
Или: пусть всего $2N >> 1$ слов, то есть N «ку» и N «кукареку»...

Задачи (построение модели источника)

Оценить алфавит и построить модели источника: а) равновероятную, б) стационарную без памяти, в) марковскую первого порядка для сообщения x, по модели оценить $I(x)$ и $I(y)$.

1. $x =$ хрюхрюхрюмяухрюмяумяухрюмяумяу (30 символов, 5 «хрю» (0) и 5 «мяу» (1) 0001011011); $y =$ рюх.
   В тексте 5 двухбуквенных сочетаний, начинающихся с «ю»: 2 «юх» и 3 «юм»
2. $x =$ кукукукукарекукукукарекукукарекукукукарекукукареку (50 символов, 5 «ку» (0) и 5 «кукареку» (1) аналогично); $y =$ кар.

Кодирование и структуры данных

Кодирование — преобразование дискретной информации

$$x \leftrightarrow X, x \in A_1^+ \quad \rightarrow \quad code(x) \in A_2^+$$

смена алфавита, **сжатие, защита от шума,** шифрование.

Декодирование — обратное преобразование $code(x) \rightarrow x$

x — сообщение, исходный текст, исходная строка, блок;

X — источник сообщений;

A_1 — первичный алфавит (до преобразования);

A_2 — вторичный (алфавит конечного представления).

Обычно A_1 — байты, исходные тексты x — бинарные файлы.

Характеристики кодов

1. Первичный алфавит A_1
2. Оптимальность (неизбыточность)
3. Избыточность (в том числе помехоустойчивость)

} модель источника!

4. Вторичный алфавит A_2 ($A_2 = \{0, 1\}$ — двоичный код)
5. Однозначная декодируемость [должна быть!]
6. Разделяемость — код $code(x)$ любой последовательности $x = \overline{a_1 \dots a_n}$ единственным образом разделим на кодовые слова $c_i = code(a_i), a_i \in A_1$:
   1. коды фиксированной ширины — $a, b, c \to$ 00, 01, 10;
   2. коды с разделителем — 1, 11, 111 (0 как разделитель символов);
   3. префиксные коды (дерево) — 0, 10, 11;
   4. прочие — например, $11, 1110111, 11100111$.

Теорема Шеннона для сжатия

Первая теорема Шеннона (для сжатия): $|code(X)| \geqslant I(X)$

$\mathbb{N}$B: **усреднение по источнику X!**

При отсутствии помех средняя длина кода может быть сколь угодно близкой к средней информации сообщения.

Следствия:

1. не существует архиватора, который любой файл сжимает до 8 байт;
2. не существует архиватора, который любой блок из 9 байт сжимает до 8 байт.
3. не существует и такого архиватора, который любой блок из $N + 1$ бит сжимает ровно до N бит, ни при каком N.

Кодирование с $|code(X)| \to I(X)$ и $|code(x)| \to I(x)$ — **оптимальное.**

Оптимальное кодирование источника X

Пусть X порождает последовательность из 2^N возможных символов.

1. Равновероятный источник ($I(X) = N$) — кодирование отдельных символов кодами фиксированной ширины N бит.
2. Стационарный источник без памяти, порождающий символы с разными постоянными вероятностями ($I(X) < N$) — кодирование **отдельных символов** кодами переменной ширины: энтропийное кодирование (коды Хаффмана, методы семейства арифметического кодирования) без учёта контекста.
3. Стационарный источник с памятью, порождающий символы с вероятностями, зависящими от контекста ($I(X) < N$) — кодирование **сочетаний символов:** энтропийное кодирование (Хф и А) с учётом контекста, словарные методы семейства LZ77 (словарь=текст) и семейства LZ78 (отдельный словарь в виде дерева/таблицы).

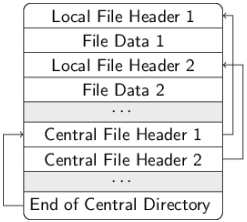
Если изначально каждый символ записан кодом фиксированной ширины (N бит) $\Rightarrow$ сжатие для 2 и 3.

Формат файла

1. Сигнатура (обычно первые 2-4 байта для общепринятых форматов)
   - быстрое распознавание типа файла (свой/чужой).
2. Метаданные (заголовок)
   - версия формата;
   - исходная длина файла;
   - смещение начала данных, их размер и формат;
   - тип сжатия, параметры для распаковки (обычно чем нестандартнее модель источника, тем объёмнее);
   - тип защиты от помех, параметры для восстановления;
   - зарезервированные поля для выравнивания;
   - контрольная сумма заголовка;
   - контрольная сумма файла и т. д.
3. Данные
   - могут включать вложенные заголовки (контейнеров) с сигнатурами.

Формат zip — несколько файловых записей

1. Каждый файл (элемент) архива имеет локальный заголовок (Local File Header); сжимается и хранится отдельно.
2. Центральный каталог — список центральных записей (Central File Header), каждая содержит заголовок файла, в том числе:
   - смещение локального заголовка;
   - длина имени файла (с относительным путём) и собственно имя.
3. End of central directory (EOCD) фиксированного размера, содержащая, в т. ч.:
   - количество записей центрального каталога;
   - размер центрального каталога;
   - смещение центрального каталога.



- Декодирование zip-файла начинается с конца.
- Каждый заголовок (и EOCD) включает в том числе сигнатуру в начале,
- но в начале всего архива могут быть доп. данные (самораспаковывающиеся архивы).

Натуральный двоичный код

Целые неотрицательные числа: от 0 до $2^N - 1$.

Для $N = 4$ — целые от 0 до $2^4 - 1 = 16 - 1 = 15$:

0	1	2	3	4	5	6	7
0000	0001	0010	0011	0100	0101	0110	0111
8	9	A (10)	B (11)	C (12)	D (13)	E (14)	F (15)
1000	1001	1010	1011	1100	1101	1110	1111

Циклическая арифметика по модулю 2^N : $2^N \equiv 0$

то есть $(2^N - 1) + 1 = 0$, $max + 1 = min$.

Беззнаковый сумматор — сложение и вычитание «в столбик».

Взвешенный код: $x = 1 \cdot \texttt{bit[0]} + 2 \cdot \texttt{bit[1]} + \ldots + 2^{N-1} \cdot \texttt{bit[N-1]} =$

$= \alpha_0 \cdot \texttt{bit[0]} + \ldots + \alpha_{N-1} \cdot \texttt{bit[N-1]}$.

Дополнительный код

Целые знаковые числа: от (-2^{N-1}) до $+(2^{N-1}-1)$.

$0, +1, +2, \ldots$ — как беззнаковые $0, 1, 2, \ldots$; используется беззнаковый сумматор $(2^N \equiv 0)$:

$$
\begin{array}{r|l|l}
0 & = 000\ldots000 & \\
+1 & = 000\ldots001 & -1 = 0-1 \equiv 2^N-1 = 111\ldots111 \\
+2 & = 000\ldots010 & -2 = 0-2 \equiv 2^N-2 = 111\ldots110 \\
 & \ldots & \ldots \\
+(2^{N-1}-1) & = 011\ldots111 & -(2^{N-1}-1) \equiv 2^N-(2^{N-1}-1) = 2^{N-1}+1 = 100\ldots001 \\
 & & -2^{N-1} \equiv 2^N-2^{N-1} = 2^{N-1} = 100\ldots000
\end{array}
$$

0	+1	+2	+3	+4	+5	+6	+7	
0000	0001	0010	0011	0100	0101	0110	0111	
	−1	−2	−3	−4	−5	−6	−7	−8
	1111	1110	1101	1100	1011	1010	1001	1000

$(-x) = 0 - x = (-1 - x) + 1 = (\sim x) + 1;$ $\qquad max + 1 = min.$

Циклическая арифметика; сложение и вычитание те же, что и для беззнаковых.

Код со смещением

Целые числа (возможно — знаковые) $x \in [a, b]$

записываем $(x - a) \in [0, b - a]$ натуральным двоичным кодом.

Значения a, b и разрядность кода N взаимозависимы: $b - a + 1 = 2^N$ (количество состояний x).

Обычно первичны $x_{\min} = a$ и разрядность N; вычисляется $x_{\max} = b$:

$$0 \leqslant x - a \leqslant 2^N - 1 \quad \Longrightarrow \quad a \leqslant x \leqslant 2^N + a - 1 \quad \Longrightarrow \quad b = 2^N + a - 1$$

Код с плавающей запятой (IEEE 754): знак + смещ. порядок + мантисса

0	0000…0000	0000…0000	$+0$	1	0000…0000	0000…0000	-0
0	0111…1111	0000…0000	$+1$	1	0111…1111	0000…0000	-1
0	1000…0000	0000…0000	$+2$	1	1000…0000	0000…0000	-2
0	1111…1110	1111…1111	ближайшее к $+\infty$ число	1	1111…1110	1111…1111	ближайшее к $-\infty$ число
0	1111…1111	0000…0000	$+\infty$	1	1111…1111	0000…0000	$-\infty$
0	1111…1111	????…????, не все нули	нечисло (nan)	1	1111…1111	????…????, не все нули	нечисло (nan)

$float$, $32 = 1 + 8 + 23$ бита $2^{-126} \leqslant |x| \leqslant 2^{127} \cdot (2 - 2^{-23})$, менее 2^{32} чисел
«одинарная точность»

$double$, $64 = 1 + 11 + 52$ бита $2^{-1022} \leqslant |x| \leqslant 2^{1023} \cdot (2 - 2^{-52})$, менее 2^{64} чисел
«двойная точность»

Нециклическая неассоциативная арифметика: $x + (y + z) \neq (x + y) + z$.

Единичный код

Избыточный невзвешенный
рефлексный (при переходе между кодовыми комбинациями изменяется только один бит)
нециклический ($max + 1 \neq min$) двоичный код

Для N битов — целые 0 до N:

0	1	2	3	4
0000	0001	0011	0111	1111
	0010	0101	1011	
	0100	1001	1101	
	1000	0110	1110	
		1010		
		1100		

ASCII и Unicode

ASCII — 128 символов и семибитная (~однобайтовая) кодировка

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

Unicode — 137 994 символ (в версии 12.1), первые 128 символов совпадают с ASCII,
и набор кодировок: **UTF-8,** UTF-16 (UTF-16BE, UTF-16LE) и **UTF-32** (UTF-32BE, UTF-32LE)

UTF-8 (актуальный стандарт — от 1 до 4 байт, первый — от 1 до 6 байт)

1 байт	0aaa aaaa	1 бит служебный, 7 на номер Unicode
2 байта	110u uuuu 10uu uuuu	$5 + 6 = 11$ бит на номер Unicode
3 байта	1110 uuuu 10uu uuuu 10uu uuuu	$4 + 2 \cdot 6 = 16$
4 байта	1111 0uuu 10uu uuuu 10uu uuuu 10uu uuuu	$3 + 3 \cdot 6 = 21$
5 байт	1111 10uu 10uu uuuu 10uu uuuu 10uu uuuu 10uu uuuu	
6 байт	1111 110u 10uu uuuu 10uu uuuu 10uu uuuu 10uu uuuu 10uu uuuu	

'я': номер в Unicode 044F = 0100 0100 1111, код UTF-8 11010001 10001111 = D1 8F, код UTF-32 00 00 04 4F

Строки

Строка — цепочка символов (в кодировке ASCII, UTF-8 и т. п.) переменной длины:

1. C-строки — цепочка символов + завершающий нулевой символ:
   - в буфере переменного размера;
   - в буфере фиксированного размера (ФС ext2/ext3 — имя файла не длиннее 255 байтов).
2. Pascal-строки — длина + цепочка символов (zip и т. п.).

Коды Грея и Джонсона

Код Грея — неизбыточный невзвешенный рефлексный циклический двоичный код

0	1	2	3	4	5	6	7
0000	0001	0011	0010	0110	0111	0101	0100
8	9	A	B	C	D	E	F
1100	1101	1111	1110	1010	1011	1001	1000

Код Джонсона — избыточный невзвешенный рефлексный циклический двоичный код

0	1	2	3	4	5	6	7
0000	0001	0011	0111	1111	1110	1100	1000

Код Морзе (Фридрих Герке)

Азбука Морзе

А • —	К — • —	Ф •• — •	1 •— — — —	. • • • • • • •
Б — •••	Л •— ••	Х • • • •	2 •• — — —	, • — • — • —
В • — —	М — —	Ц — • — •	3 ••• — —	; — • — • — •
Г — — •	Н — •	Ч — — — •	4 •••• —	: — — — •••
Д — ••	О — — —	Ш — — — —	5 • • • • •	? •• — — ••
Е •	П •— — •	Щ — — • —	6 — ••••	! — — •• — —
Ж •••—	Р • — •	Ь,Ъ — •• —	7 — — •••	- — •••• —
З — — ••	С •••	Ы — •— —	8 — — — ••	« • — •• — •
И ••	Т —	Э ••— ••	9 — — — — •	(—•— —•—
Й • — — —	У ••—	Ю ••— —	0 — — — — —	/ — ••— •
		Я • — • —		

Код Бодо (Дональд Мюррей)

Международный телеграфный код №2 (ITA2) + 00000 (русские буквы) = МТК-2

Русский шрифт		**Е**	Перевод строки ≡	Пробел	Возврат каретки ≺	**Т**	**А**	**И**	**Н**	**О**	**С**	**Р**	**Х**	**Д**	**Л**	**З**	**У**	**Ц**	**М**	**Ф**	**Й**	**Г**	**П**	**Ы**	**Б**	**В**		**К**	**Ж**	**Ь**	**Я**		
Цифры		**3**				**5**	**-**	**8**	**,**	**9**	**'**	**Ч** Ч4	**Щ**	кто там?	**)**	**+**	**7**	**:**	**.**	**Э**	**Ю** (зв)	**Ш**	**0**	**Ъ** 5 6	**?**	**2**	Цифры	**(**	**=**	**/**	**1**	Буквы лат.	Буквы рус.
Латинский шрифт		**E**				**T**	**A**	**I**	**N**	**O**	**S**	**R**	**H**	**D**	**L**	**Z**	**U**	**C**	**M**	**F**	**J**	**G**	**P**	**Y**	**B**	**W**		**K**	**V**	**X**	**Q**		
Ведущие отверстия	1	●					●				●			●		●	●			●	●			●	●	●	●	●		●	●	●		
	2		●				●	●				●			●		●	●			●	●	●			●	●	●	●			●	●		
		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○		
	3			●				●	●		●		●				●	●	●	●			●	●				●	●	●	●	●		
	4				●				●	●		●		●				●	●	●	●	●			●		●	●	●	●		●		
	5					●				●			●		●	●				●			●	●	●	●	●	●		●	●	●	●		

фиксированной ширины 5, режимы; цифра 4 и русская Ч — считаются одним; строчных нет

Спасибо за внимание!

МИЭТ
www.miet.ru

Александра Игоревна Кононова: +7-977-977-97-29 (WhatsApp),
gitlab.com/illinc/raspisanie

ОТИК
https://gitlab.com/illinc/otik