

Кодирование информации

- § 11. Кодирование текстов
- § 12. Кодирование рисунков: растровый метод
- § 13. Кодирование рисунков: другие методы
- § 14. Кодирование звука и видео
- § 15. Передача информации
- § 16. Сжатие данных

Кодирование информации

§ 11. Кодирование текстов

Равномерное кодирование

1) выбрали алфавит:

$\{А, Б, В, \dots, Я, \dots\}$

всего M символов (мощность)



Сколько битов
нужно?

2) длина кода i битов, так что: $2^i \geq M$

3) составили кодовую таблицу:

А	Б	В	...	Я	...
0	1	2	...	32	...

В ДВОИЧНОМ КОДЕ

000000	000001	000010	...	100000	...
---------------	---------------	---------------	-----	---------------	-----

Шрифты

Файл `vasya.txt` («только текст»):

01000001	01000010	01000011
----------	----------	----------

65

66

67

КОДЫ
СИМВОЛОВ



Какие это символы?

Кодовая таблица:

...	A	B	C	...
...	65	66	67	...



Где взять изображение?

в файле его нет!

Шрифтовой файл (`.ttf`, `.otf`):

65 → **A**

66 → **B**

67 → **C**

Times New Roman



Кодировка (коддовая таблица) ASCII

ASCII = American Standard Code for Information Interchange

Всего **128** символов, коды с **0** до **127**



Сколько бит нужно?

7-битная кодировка

48 – 57 цифры **0..9**

65 – 90 заглавные латинские буквы **A-Z**

97 – 122 строчные латинские буквы **a-z**

знаки препинания: . , : ; ! ?

специальные знаки: + - * / () { } []



Международный стандарт!

Однобайтные кодировки

Добавим 1 бит до 8 бит на символ.



Сколько символов можно закодировать?

Всего **256** символов, коды с **0** до **255**

0	1	127	128	254	255
коды ASCII				расширение (национальный алфавит)	

Кодовая страница (расширенная таблица ASCII)

для русского языка:

Windows-1251 – для системы *Windows* (Интернет)

KOI8-R – для системы *UNIX* (Интернет)

CP-866 – альтернативная кодировка (для системы *MS DOS*)

MacCyrillic – для компьютеров фирмы *Apple*

Однobaйтные кодировки



- небольшой размер файла
- удобно обрабатывать программистам



- можно использовать только 256 СИМВОЛОВ

- неизвестно какая кодировка

Windows-1251

KOI8-R

Здравствуй, мир!
ъДТБЧУФЧХК, НЙТ!

гДПЮБЯРБСИ, ЛХП!
Здравствуй, мир!

Кодировки UNICODE

нужно
больше!

Идея: все символы в одну таблицу!

16 битов на символ $\rightarrow 2^{16} = 65536$ символов

Стандарт UNICODE: место для 1 112 064 символов

Windows:

UTF-16 2 или 4 байта на символ

заполнены
~ 136 тысяч (2017)

Linux и др.:

UTF-8 от 1 до 4 байт на символ

более 80% сайтов



Символы ASCII имеют те же коды!



- можно использовать много символов



- увеличивается объём файла
- сложнее обрабатывать, если переменное число байт на символ

Информационный объём текста

При равномерном кодировании:

$$I = L \cdot i$$

Количество информации ← I ← Место для 1 символа

L ← Количество символов

i ← Место для 1 символа

Задача. Определите информационный объём сообщения

ПРИВЕТ, МИР!

при использовании 16-битной кодировки.

- 1) $L = 12$ символов
- 2) $i = 16$ бит = **2 байта**
- 3) $I = 12 \cdot 16 = 192$ бита
 $I = 12 \cdot 2 = \mathbf{24}$ байта



Считаем пробелы
и знаки препинания!

Информационный объём текста

Задача. Рассказ, набранный на компьютере, содержит 12 страниц, на каждой странице 48 строк, в каждой строке 64 символа. Определите информационный объём рассказа в Кбайтах в кодировке, в которой каждый символ кодируется 16 битами.

$$I = L \cdot i$$

$$3 \cdot 4 = 3 \cdot 2^2$$

$$3 \cdot 16 =$$

$$3 \cdot 2^4$$

$$2^6$$

$$1) L = 12 \cdot 48 \cdot 64 = 9 \cdot 2^{12} \text{ СИМВОЛОВ}$$

$$2) i = 16 \text{ бит} = 2 \text{ байта}$$

$$3) I = 9 \cdot 2^{12} \cdot 2 = 9 \cdot 2^{13}$$

байтов

$$= \frac{9 \cdot 2^{13}}{2^{10}} = 72 \text{ Кбайт}$$



1 Кбайт = 2^{10} байтов!

Информационный объём текста

Задача. Информационный объём статьи 96 Кбайт.

Сколько страниц займет статья, если на одной странице электронного документа помещается 32 строки по 64 символа, а каждый символ занимает 8 бит памяти?.

 2^5 2^6

$$I = L \cdot i$$

$$1) L = x \cdot 32 \cdot 64 = x \cdot 2^{11}$$

$$2) i = 8 \text{ бит} = 1 \text{ байт}$$

$$3) I = x \cdot 2^{11} \cdot 1 = x \cdot 2^{11} \text{ байт} = 96 \text{ Кбайт}$$

$$4) x \cdot 2^{11} \text{ байт} = 96 \cdot 2^{10}$$

байт

$$x \cdot 2 = 96$$

$$x = 48 \text{ страниц}$$

$$1 \text{ Кбайт} = 2^{10} \text{ байт!}$$

Информационный объём текста

Задача. Некоторый файл перекодировали из 8-битной кодировки в 16-битную, в результате его информационный объём увеличился на 40 Кбайт. Чему был равен информационный объём файла до перекодирования?

1) информационный объём увеличился в 2 раза

$$I_2 = 2 I_1$$

2) по условию

$$I_2 = I_1 + 40$$

3) тогда

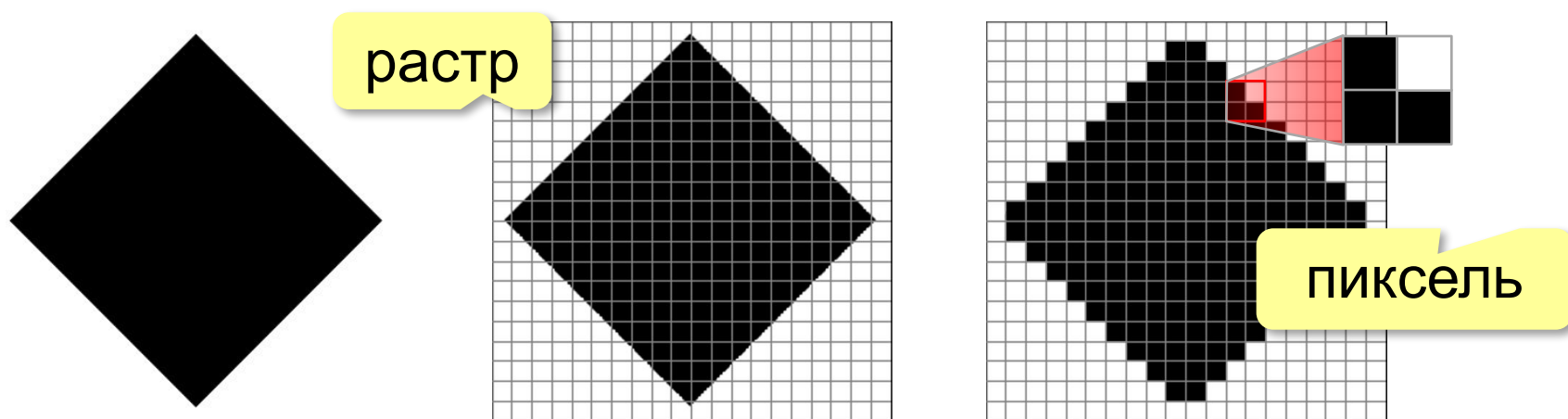
$$2I_1 = I_1 + 40$$

4) $I = 40$ Кбайт

Кодирование информации

§ 12. Кодирование рисунков: растровый метод

Растровое кодирование



дискретизация



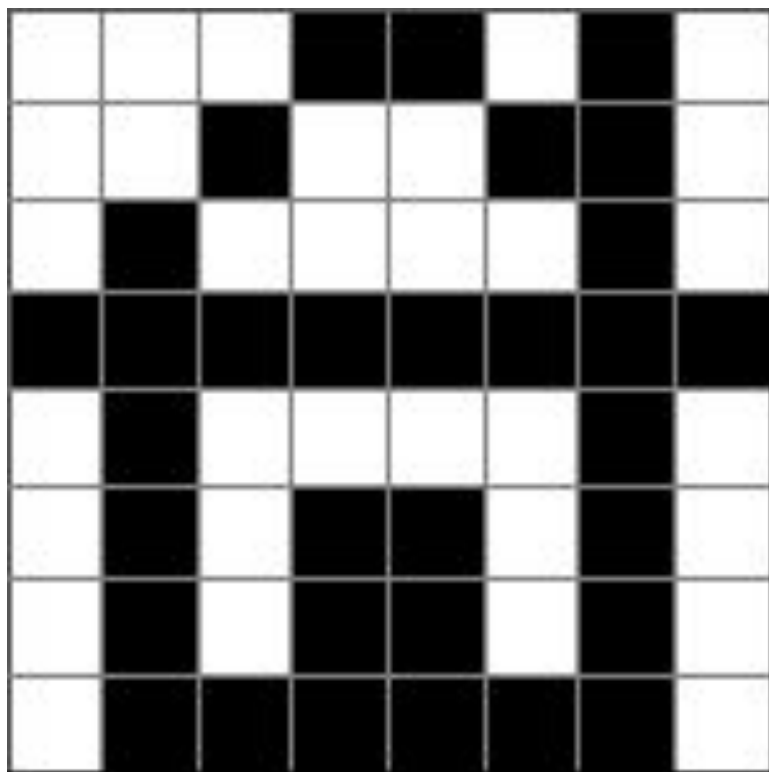
Рисунок искажается!



Пиксель – это наименьший элемент рисунка, для которого можно задать свой цвет.

Растровое изображение – это изображение, которое кодируется как множество пикселей.

Растровое кодирование



1	1	1	0	0	1	0	1	E5
1	1	0	1	1	0	0	1	D9
1	0	1	1	1	1	0	1	BD
0	0	0	0	0	0	0	0	00
1	0	1	1	1	1	0	1	BD
1	0	1	0	0	1	0	1	A5
1	0	1	0	0	1	0	1	A5
1	0	0	0	0	0	0	1	81

E5D9BD00BDA5A581

Разрешение

Разрешение – это количество пикселей, приходящихся на дюйм размера изображения.

ppi = *pixels per inch*, пикселей на дюйм

1 дюйм = 2,54 см



300 ppi

печать

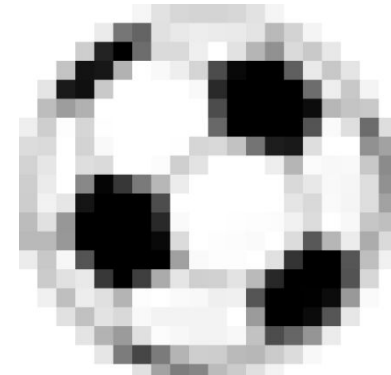


96 ppi

экран



48 ppi



24 ppi

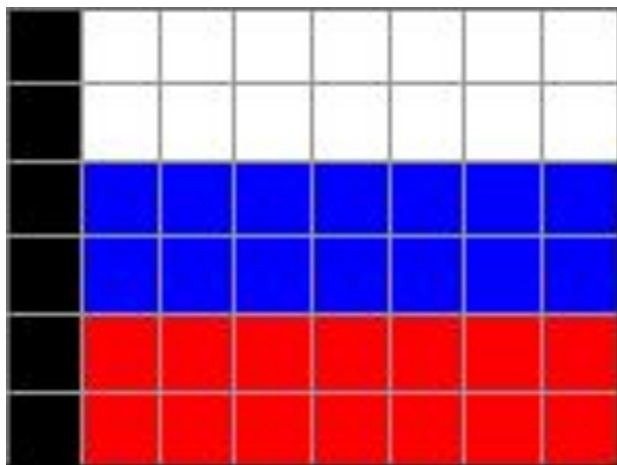
Разрешение

Задача. Какой размер в пикселях должен иметь закодированный рисунок с разрешением **300 ppi**, чтобы с него можно было сделать отпечаток размером **10×15 см**?

$$\text{высота} \quad \frac{10 \text{ см} \times 300 \text{ пикселей}}{2,54 \text{ см}} \approx \mathbf{1181 \text{ пиксель}}$$

$$\text{ширина} \quad \frac{15 \text{ см} \times 300 \text{ пикселей}}{2,54 \text{ см}} \approx \mathbf{1771 \text{ пиксель}}$$

Кодирование цвета



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



Как выводить на монитор цвет с кодом 00?



Как закодировать цвет в виде чисел?

Цветовая модель RGB

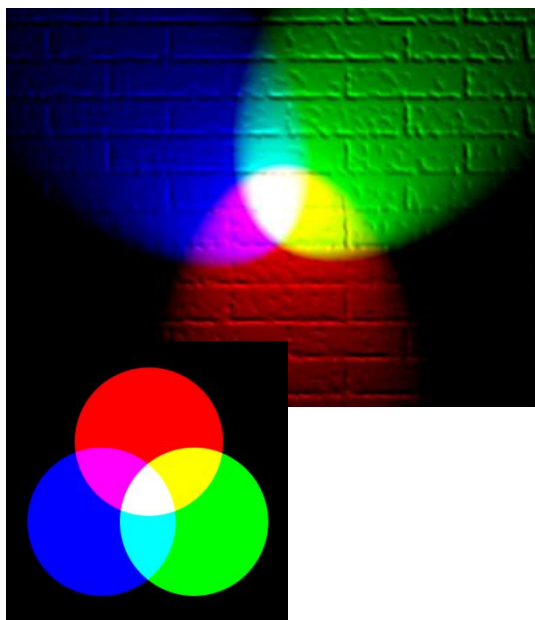
Д. Максвелл, 1860

цвет = (**R**, **G**, **B**)

red *green* *blue*

красный зеленый синий

0..255 0..255 0..255



■ (0, 0, 0)

□ (255, 255, 255)

■ (255, 0, 0)

■ (255, 150, 150)

■ (150, 150, 150)

■ (0, 255, 0)

■ (255, 255, 0)

■ (0, 0, 255)

■ (100, 0, 0)

■ (20, 20, 20)



Сколько разных цветов можно кодировать?

$256 \cdot 256 \cdot 256 = \mathbf{16\ 777\ 216}$ (*True Color*, «истинный цвет»)



RGB – цветовая модель для устройств, излучающих свет (мониторов)!

Глубина цвета

Глубина цвета — это количество битов, используемое для кодирования цвета пикселя.



Сколько памяти нужно для хранения цвета 1 пикселя в режиме *True Color*?

R (0..255) 256 = 2^8 оттенков 8 битов = 1 байт

R G B: 24 бита = 3 байта

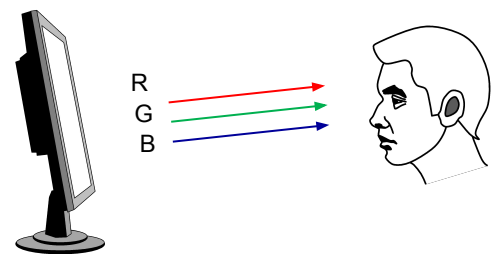
True Color
(истинный цвет)

Задача. Определите размер файла, в котором закодирован растровый рисунок размером **20×30 пикселей** в режиме истинного цвета (*True Color*).

$20 \cdot 30 \cdot 3 \text{ байта} = \mathbf{1800}$
байт

Кодирование цвета при печати (СМУК)

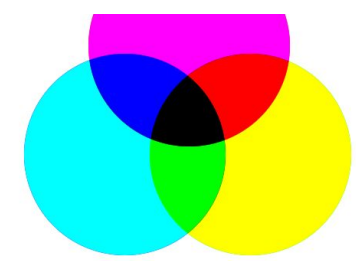
монитор



печатный документ

Белый – красный = голубой
Белый – зелёный = пурпурный
Белый – синий = желтый

C = Cyan
M = Magenta
Y = Yellow



Модель CMY

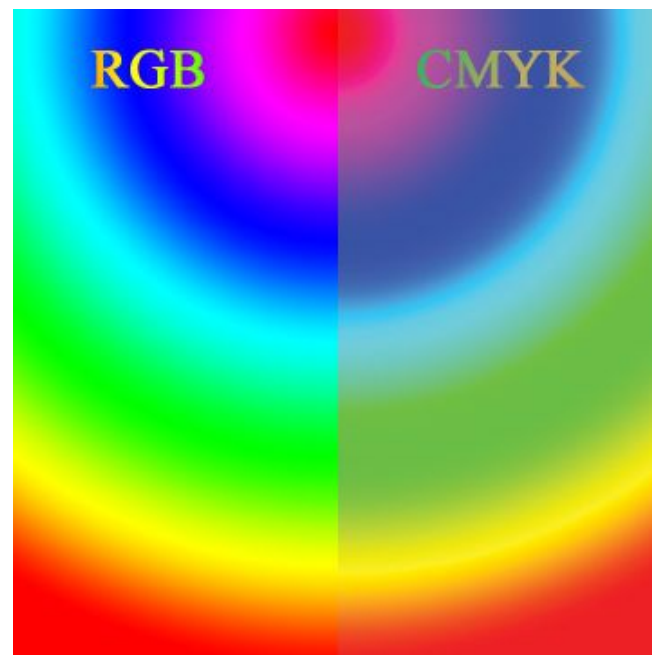
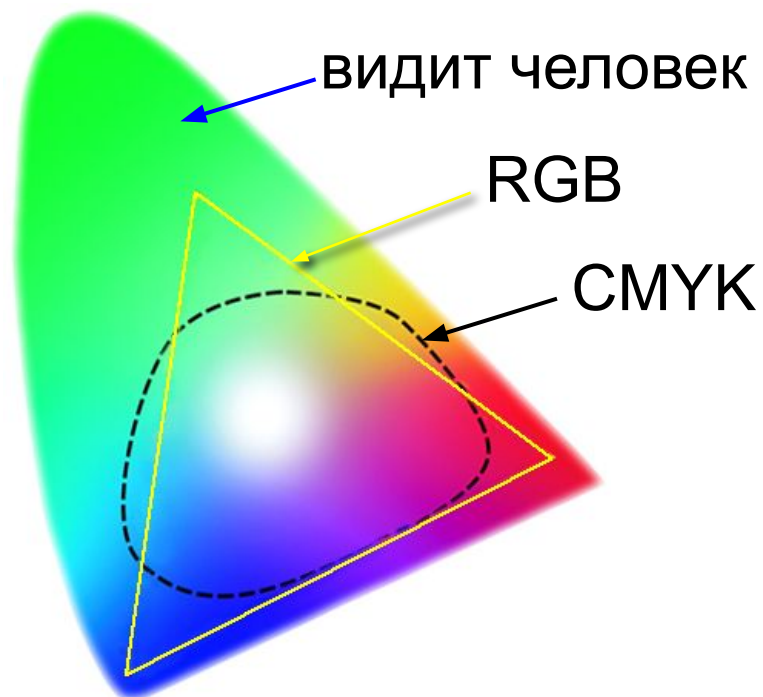
C	M	Y	
0	0	0	
255	255	0	
255	0	255	
0	255	255	
255	255	255	

Модель СМУК: + **Key color**



- меньший расход краски и лучшее качество для чёрного и серого цветов

RGB и CMYK



- не все цвета, которые показывает монитор (RGB), можно напечатать (CMYK)
- при переводе кода цвета из RGB в CMYK цвет искажается

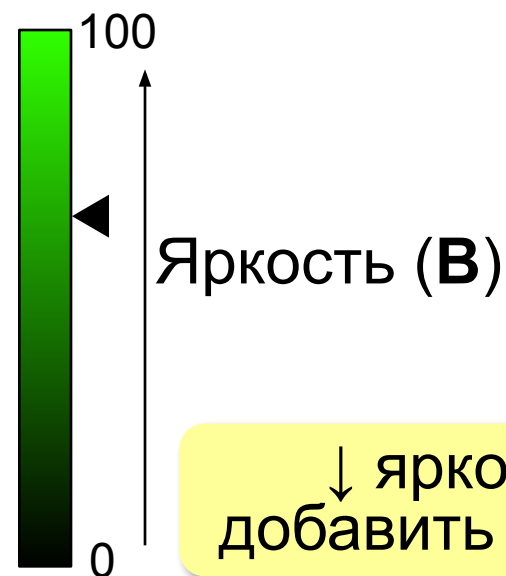
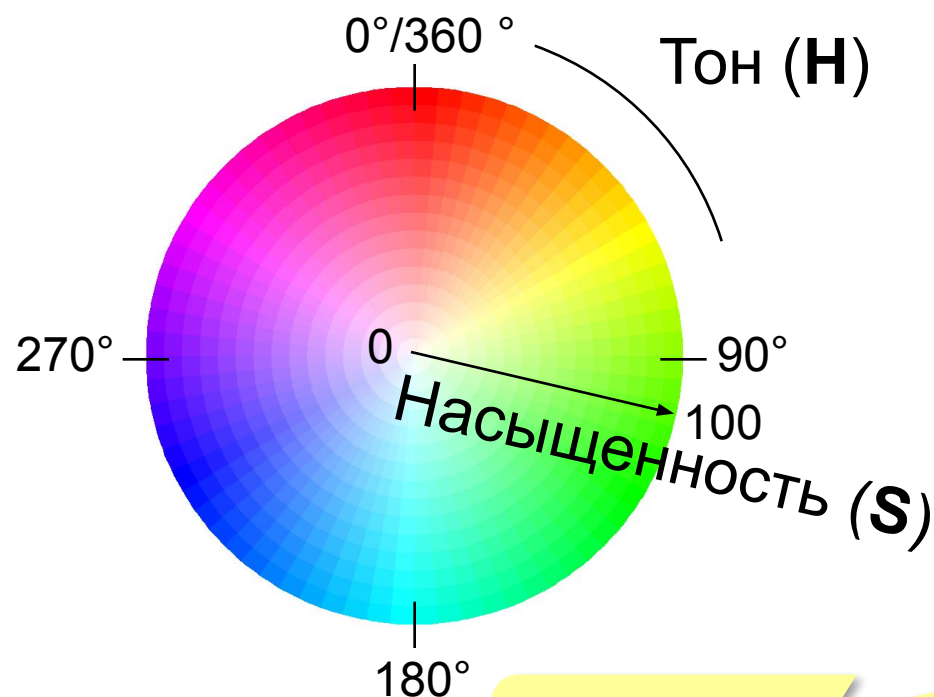
 **RGB(0,255,0)**
 **CMYK(65,0,100,0)**
→ **RGB(104,175,35)**

Цветовая модель HSB (HSV)

HSB = *Hue* (тон, оттенок)

Saturation (насыщенность)

Brightness (яркость) или *Value* (величина)



↓ яркость —
добавить чёрного

↓ насыщенность —
добавить белого

Кодирование с палитрой



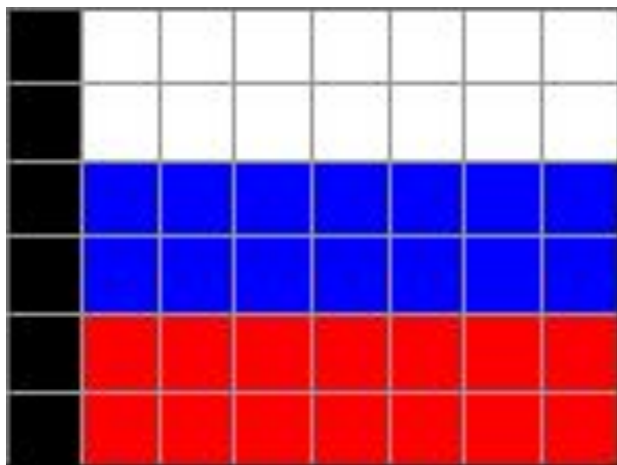
Как уменьшить размер файла?

- уменьшить разрешение
- уменьшить глубину цвета

снижается
качество

Цветовая палитра – это таблица, в которой каждому цвету, заданному в виде составляющих в модели RGB, сопоставляется числовой код.

Кодирование с палитрой



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

Палитра:

0	0	0	0	0	255	255	0	0	255	255	255
цвет 00 ₂			цвет 01 ₂			цвет 10 ₂			цвет 11 ₂		



Какая глубина цвета?

2 бита на пиксель



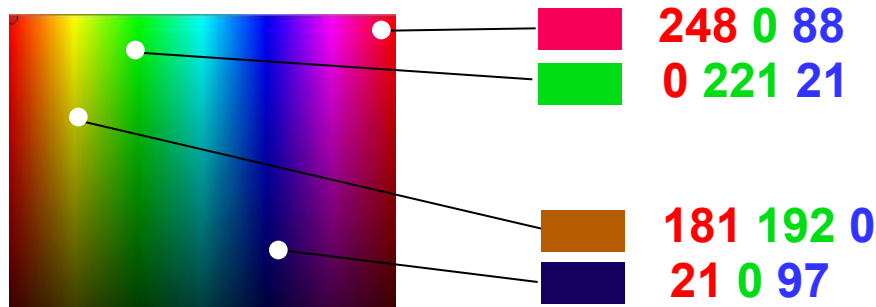
Сколько занимает палитра?

3 · 4 = 12 байтов

Кодирование с палитрой

Шаг 1. Выбрать количество цветов: 2, 4, ... 256.

Шаг 2. Выбрать 256 цветов из палитры:



Шаг 3. Составить палитру (каждому цвету – номер 0..255)
палитра хранится в начале файла

0	1	45	254	255
248 0 88	0 221 21	148 35 115	181 192 0	21 0 97

Шаг 4. Код пикселя = номеру его цвета в палитре

1	254	45	14	...	12	23
---	-----	----	----	-----	----	----

Кодирование с палитрой

Задача. Определите размер файла, в котором закодирован растровый рисунок размером **20×30 пикселей** с 16 цветной палитрой. Место для хранения палитры не учитывать.

- 1) количество пикселей $20 \cdot 30 = 600$
- 2) 16 цветов \rightarrow 4 бита на пиксель ($16 = 2^4$)
- 3) $600 \cdot 4 = 2400$ бита = **300 байт**

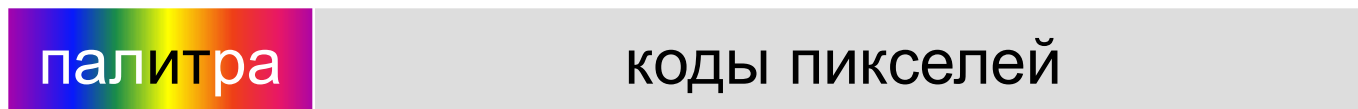


Сколько занимает палитра?

$$3 \cdot 16 = 48 \text{ байтов}$$

Кодирование с палитрой















Файл с палитрой:





3 байта на цвет

Количество цветов	Размер палитры (байтов)	Глубина цвета (битов на пиксель)
2	6	1
4	12	2
16	48	4
256	768	8

Растровые рисунки: форматы файлов

Формат	True Color	Палитра	Прозрачность	Анимация
BMP				
JPG				
GIF				
PNG				

Растровое кодирование: итоги

-  универсальный метод (можно закодировать любое изображение)
- единственный метод для кодирования и обработки размытых изображений, не имеющих чётких границ (фотографий)
-  есть **потеря информации** (почему?)
- при изменении размеров цвет и форма объектов на рисунке **искажается**
- **размер файла** не зависит от сложности рисунка (а от чего зависит?)

Кодирование информации

§ 13. Кодирование рисунков: другие методы

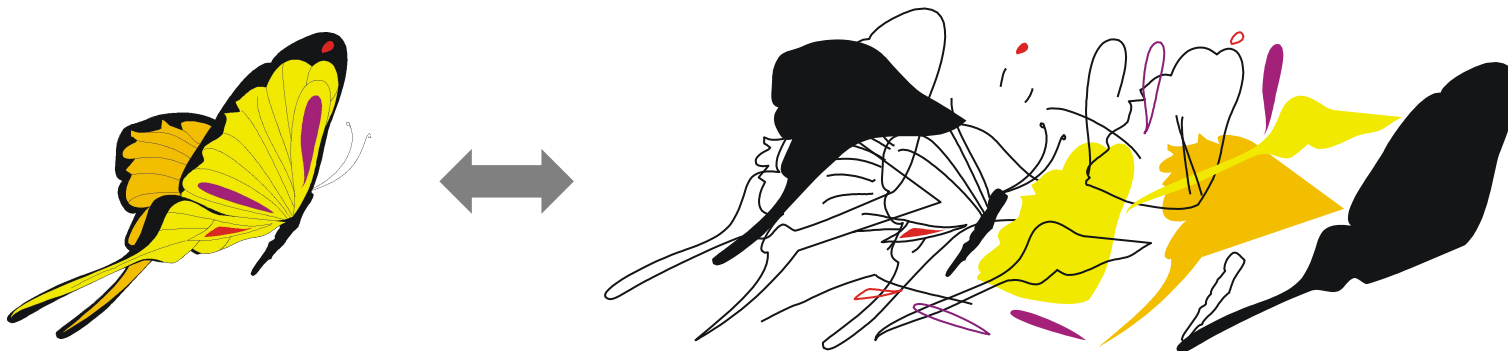
Векторное кодирование

Рисунки из геометрических фигур:

- отрезки, ломаные, прямоугольники
- окружности, эллипсы, дуги
- сглаженные линии (кривые Безье)

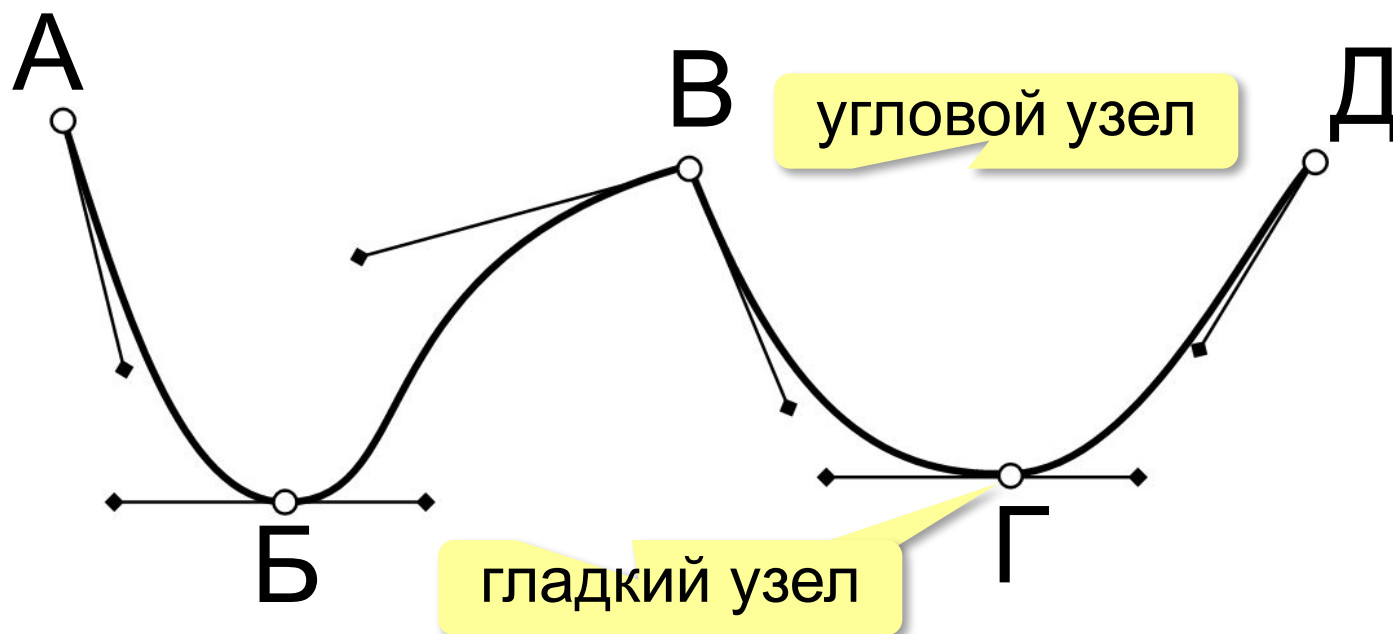
Для каждой фигуры в памяти хранятся:

- размеры и координаты на рисунке
- цвет и стиль границы
- цвет и стиль заливки (для замкнутых фигур)



Векторное кодирование

Кривые Безье:



Хранятся координаты узлов и концов «рычагов»
(3 точки для каждого узла, кривые 3-го порядка).

Векторное кодирование (итоги)



- лучший способ для **чертежей, схем, карт**
- при кодировании **нет потери информации**
- при изменении размера **нет искажений**



растровый
рисунок



векторный
рисунок

- меньше **размер файла**, зависит от сложности рисунка



- не используют для **фотографий** и **размытых изображений**

Векторное кодирование: форматы файлов

- **WMF, EMF** (*Windows Metafile*)
- **ODG** (формат *OpenOffice Draw*)
- **CDR** (программа *CorelDraw*)
- **AI** (программа *Adobe Illustrator*)
- **EPS** (для подготовки печатных изданий)
- **SVG** (*Scalable Vector Graphics*,
масштабируемые
векторные изображения)

для веб-
страниц

Векторные рисунки: SVG

<svg>

прямоугольник

размеры

<rect width="135" height="30"

x="0" y="10"

координаты

stroke-width="1" stroke="rgb(0,0,0)"

контур

fill="rgb(255,255,255)"/>

заливка

<rect width="135" height="30" x="0" y="40"

stroke-width="1" stroke="rgb(0,0,0)"

fill="rgb(0,0,255)"/>

<rect width="135" height="30" x="0" y="70"

stroke-width="1" stroke="rgb(0,0,0)"

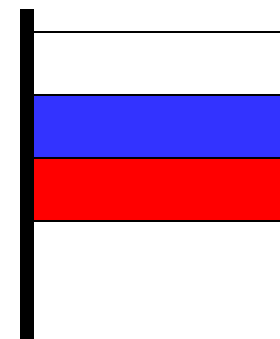
fill="rgb(255,0,0)"/>

<line x1="0" y1="0"

x2="0" y2="150"

stroke-width="15" stroke="rgb(0,0,0)"/>

</svg>

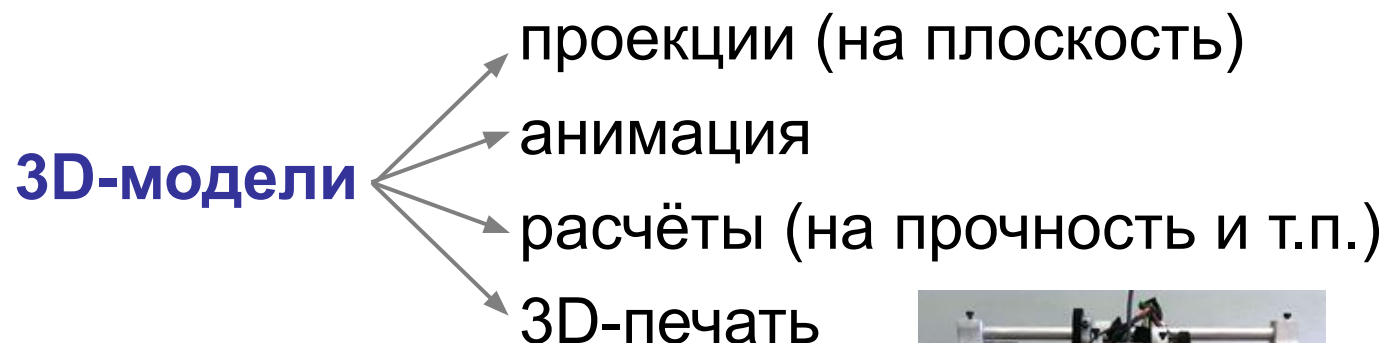


Это программа для браузера!

3D-графика

Трёхмерная графика (3D-графика) – это раздел компьютерной графики, который занимается созданием моделей и изображений *трёхмерных* объектов.

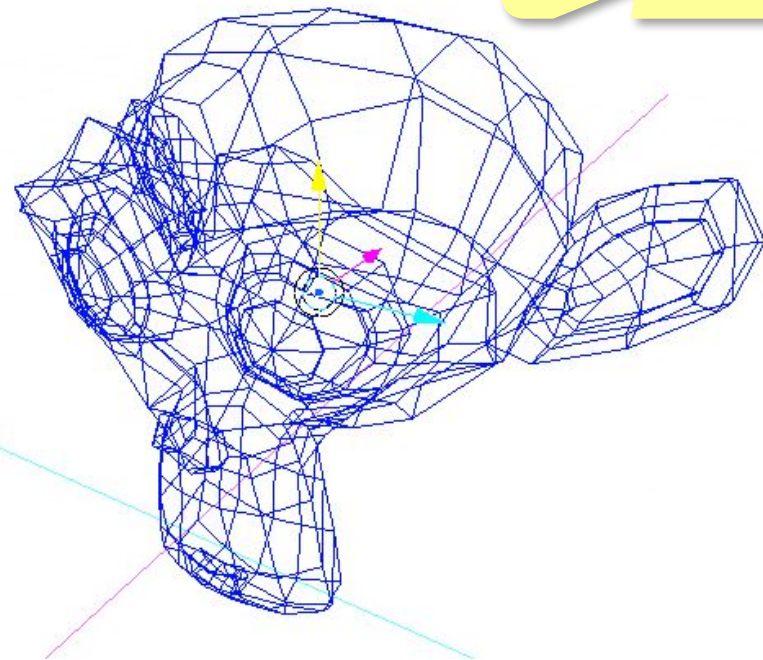
3D-модели: каждая точка имеет 3 координаты



Построение каркаса (рёбер)

узлы
(вершины)

рёбра

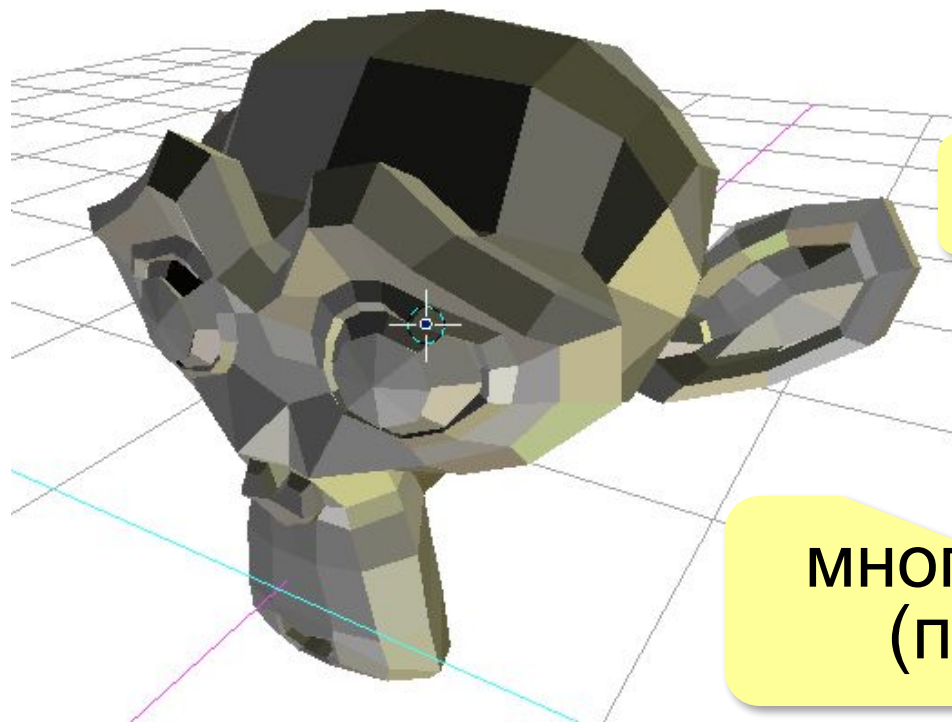


Хранятся координаты
точек (x, y, z) !



~~Раск~~ривая или векторная?

Поверхность

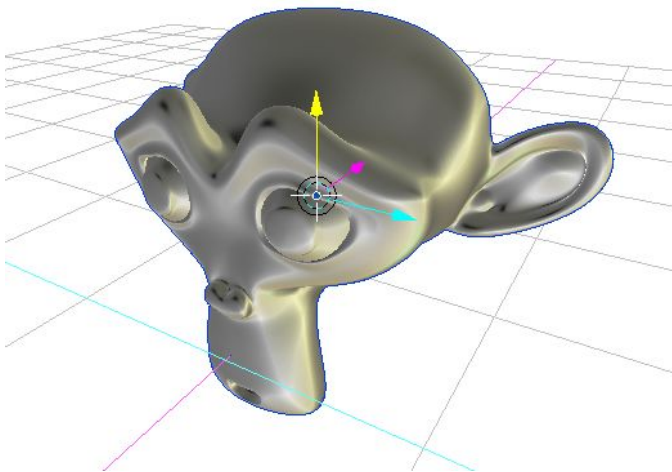


треугольники

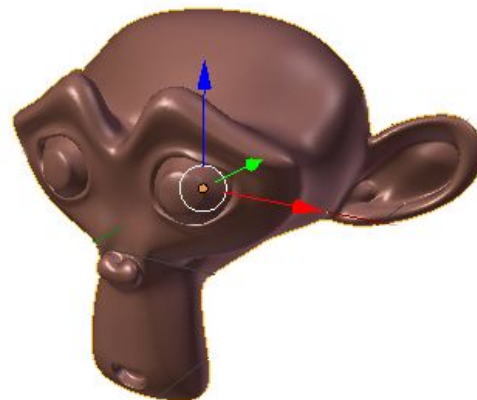
многоугольники
(полигоны)

Завершение модели

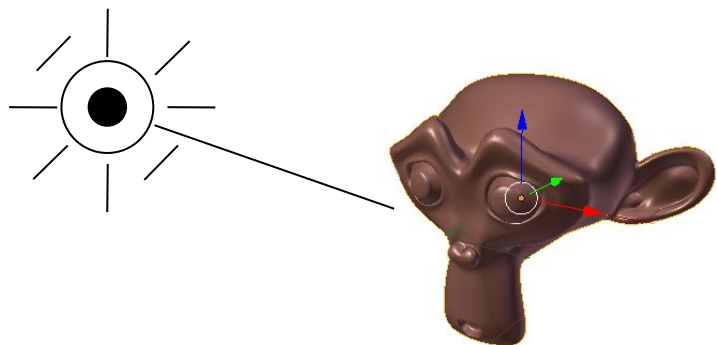
сглаживание



материал

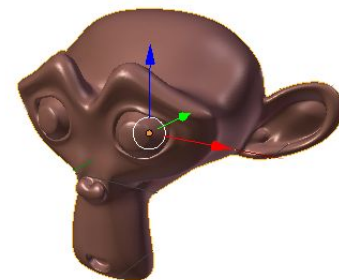
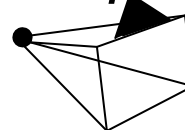


установка света



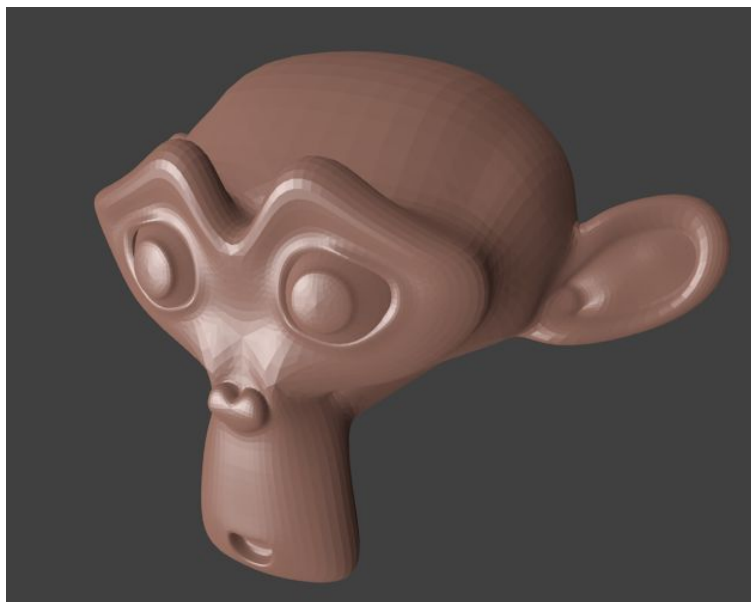
установка камеры

камера



Результат

рендеринг



3D-печать

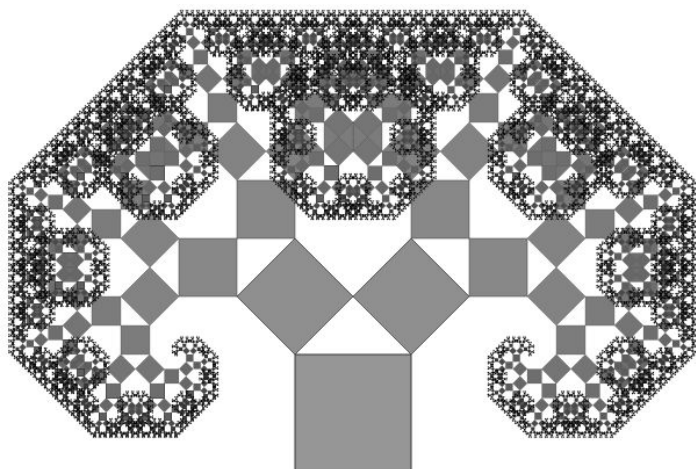


Рендеринг (визуализация) — построение двухмерного изображения по 3D-модели.

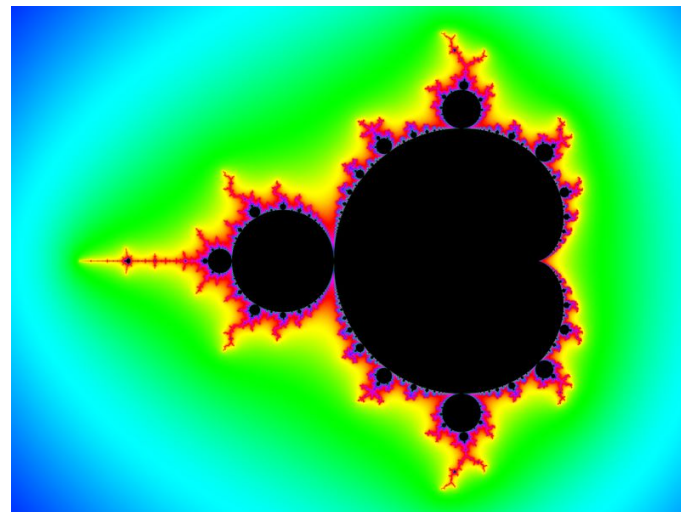
Фрактальная графика

Фрактал — это фигура, обладающая *самоподобием*: основная фигура состоит из нескольких таких же, только меньшего размера.

Пифагорово дерево



Множество Мандельброта

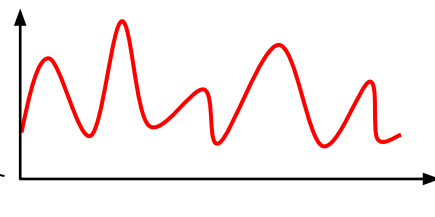
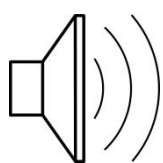


Задаётся математической формулой + алгоритмом построения.

Кодирование информации

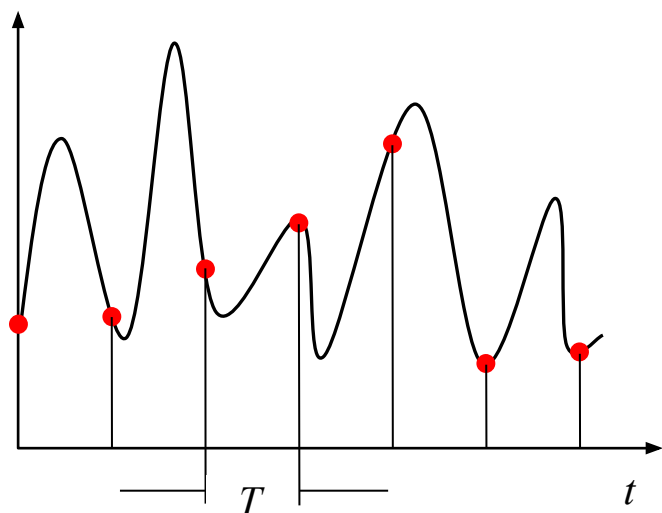
§ 14. Кодирование звука и видео

Оцифровка звука



аналоговый
сигнал

Оцифровка – это преобразование аналогового сигнала в цифровой код (дискретизация).



Человек слышит

16 Гц ... 20 кГц

T – интервал дискретизации (с)
 $f = \frac{1}{T}$ – частота дискретизации
(Гц, кГц)

8 кГц – минимальная частота для
распознавания речи

11 кГц, 22 кГц,

44,1 кГц – качество CD-дисков

48 кГц – фильмы на DVD

96 кГц, 192 кГц

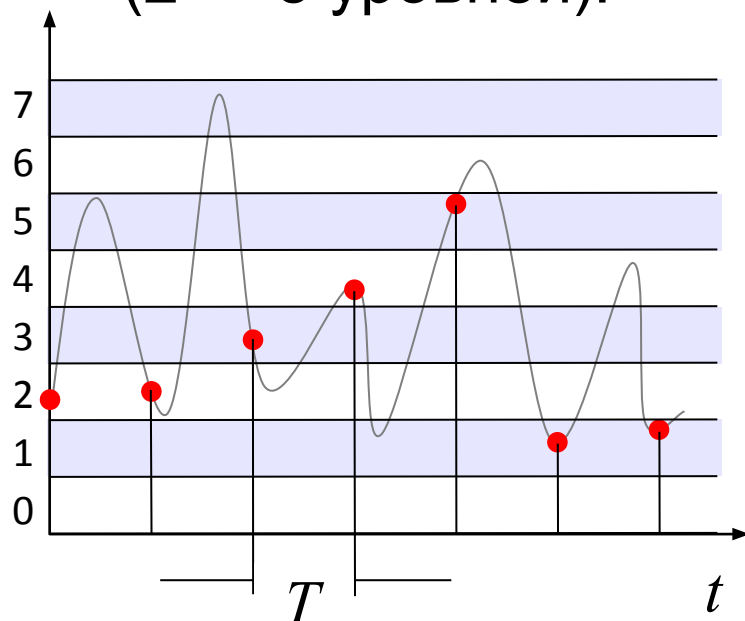
Глубина кодирования



Результаты измерения записываются как целое число!

АЦП = **А**налого-**Ц**ифровой

Преобразователь
3-битное кодирование
($2^3 = 8$ уровней):



в звуковой
карте

8 битов = 256 уровней

16 битов = 65536 уровней

24 бита = $2^{24} = 16\,777\,216$ уровней

Глубина кодирования — это
число битов для хранения
одного результата измерений.



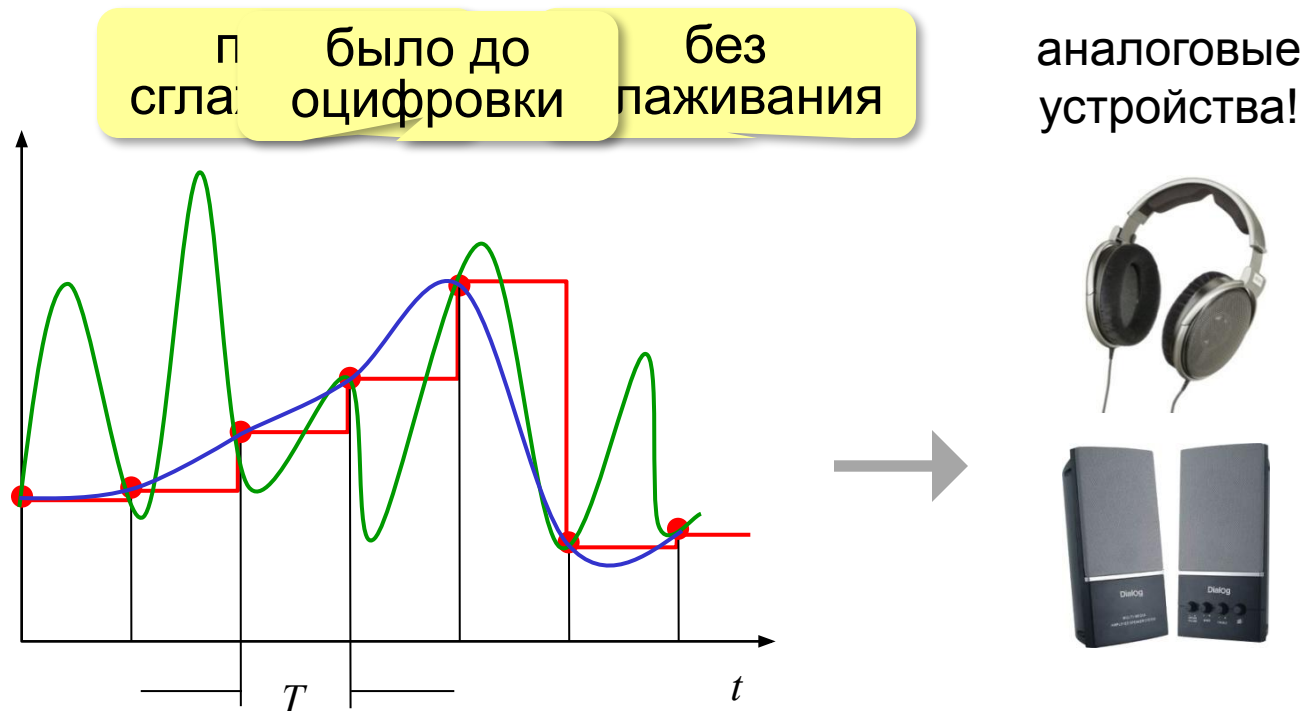
Лучше больше или меньше?

Вывод цифрового звука

в звуковой
карте

Как восстановить сигнал?

ЦАП = Цифро-Аналоговый Преобразователь



? Как улучшить качество?

уменьшать T

? Что при этом ухудшится?

↑ размер файла

Оцифровка звука

Задача. Определите информационный объем данных, полученных при оцифровке звука длительностью **1 минута** с частотой **44 кГц** с помощью **16-битной** звуковой карты. Запись выполнена в режиме «стерео».

За 1 сек *каждый канал* записывает **44000** значений,
каждое занимает **16 битов = 2 байта**
всего **$44000 \cdot 2$ байта = 88000 байтов**


С учётом «стерео»
всего **$88000 \cdot 2 = 176000$ байтов**

За 1 минуту
 $176000 \cdot 60 = 10560000$ байтов
 ≈ 10313 Кбайт ≈ 10 Мбайт

Оцифровка – итог

 можно закодировать **любой звук** (в т.ч. голос, свист, шорох, ...)

 • есть **потеря информации**
• большой **объем файлов**

 Какие свойства оцифрованного звука определяют качество звучания?

Форматы файлов:

WAV (*Waveform audio format*), часто без сжатия (размер!)

MP3 (*MPEG-1 Audio Layer 3*, сжатие с учётом восприятия человеком)

AAC (*Advanced Audio Coding*, 48 каналов, сжатие)

WMA (*Windows Media Audio*, потоковый звук, сжатие)

OGG (*Ogg Vorbis*, открытый формат, сжатие)

Инструментальное кодирование

MIDI (*Musical Instrument Digital Interface* — цифровой интерфейс музыкальных инструментов).

в файле **.mid**:

- нота (высота, длительность)
- музыкальный инструмент
- параметры звука (громкость, тембр)
- до 1024 каналов

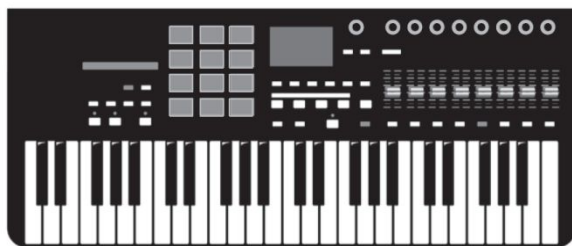
128 мелодических и
47 ударных

программа для
звуковой карты!

в памяти звуковой карты:

- образцы звуков (волновые таблицы)

MIDI-клавиатура:



- **нет потери информации** при кодировании инструментальной музыки
- **небольшой размер файлов**



невозможно закодировать нестандартный звук, голос

Кодирование видео



Видео = изображения + звук Синхронность!

изображения:

- ≥ 25 кадров

- **PAL/SEC**

за 1 с

за 1 м

- **HDTV**: 12

- **4K**: 4096

- исходный

- сжатие (к

DivX, Xvid,

звук:

- 48 кГц, 16

- сжатие (к



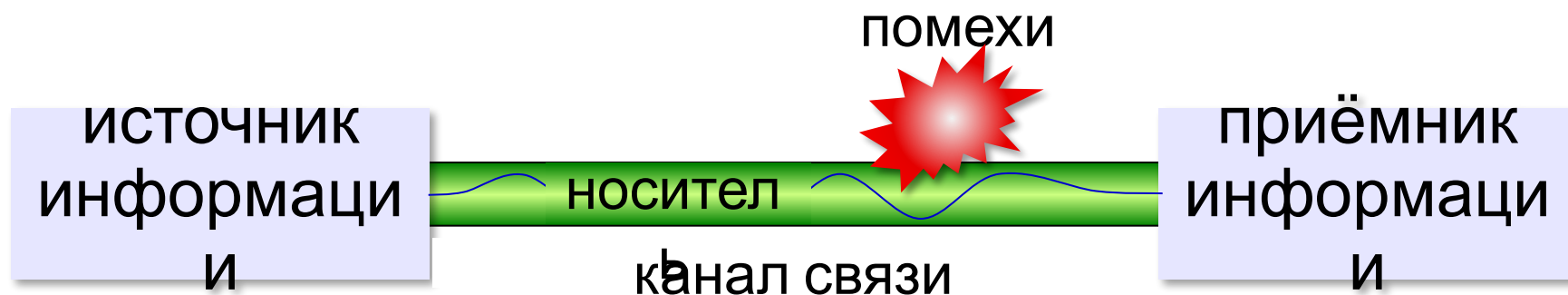
Форматы видеофайлов

- AVI** – *Audio Video Interleave* – чередующиеся звук и видео; могут использоваться разные алгоритмы сжатия
- MPEG** – *Motion Picture Expert Group*
- MP4** – несколько потоков видео + субтитры
- WMV** – *Windows Media Video*, формат фирмы *Microsoft*
- MOV** – *Quick Time Movie*, формат фирмы *Apple*
- WebM** – открытый формат, поддерживается браузерами

Кодирование информации

§ 15. Передача информации

Как происходит передача данных?



Канал связи — это среда и технические устройства, с помощью которых передаётся информация.

почта, воздух, электрические кабели, радиоканал

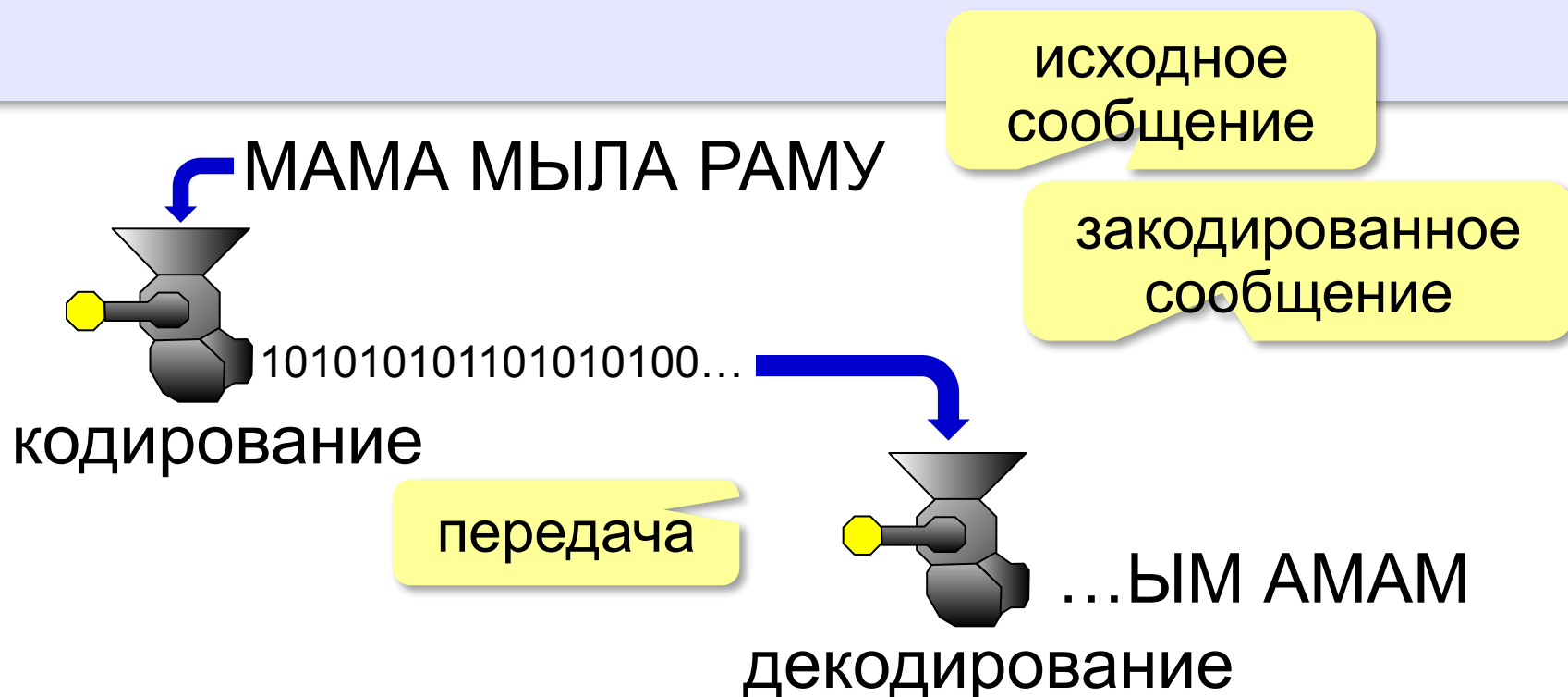
Носитель — это объект, который может некоторое время сохранять информацию.

бумага, звуковые волны, радиоволны, ток

Сигнал и сообщение

Сигнал — это изменение свойств носителя, которое используется для передачи информации.

Сообщение — это последовательность сигналов.



Скорость передачи данных

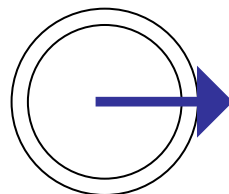


км/час
м/с



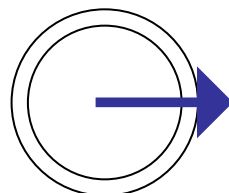
В каких единицах?

лимонад



л/мин
м³/с

10101001



бит/с кбит/с
байт/с Мбит/с
Мбайт/с

Средняя скорость передачи данных:

$$v = I / t,$$

где I — количество переданных данных

t — время передачи.

Измерение скорости

1 **к**бит/с = **1 000** бит/с

1 Мбит/с = **1 000 000** бит/с

1 Гбит/с = **1 000 000 000** бит/с

Сравните:

1 **К**байт = 1024 байта

1 Мбайт = 1024 Кбайта

1 Гбайт = 1024 Мбайта

Файл размером 1000 бит был передан по каналу связи за 40 с. Найдите среднюю скорость передачи данных в битах в секунду. **25 бит/с**

Файл размером 1000 **байт** был передан по каналу связи за 40 с. Найдите среднюю скорость передачи данных в битах в секунду. **200 бит/с**

Пропускная способность канала связи — это предельно возможная скорость передачи.

Вычисление объёма переданных данных

Скорость передачи данных по линии связи равна 200 бит/с. Сколько бит будет передано за 5 секунд?

$$I = v \cdot t$$

Ответ: 1000 бит

Скорость передачи данных по линии связи равна 200 бит/с. Сколько **байт** будет передано за 5 **минут**?

Ответ: 7500 байт

Вычисление времени передачи

Скорость передачи данных по линии связи равна 1000 бит/с. Сколько секунд потребуется на передачу файла размером 4000 бит?

$$t = I / v \quad \text{Ответ: } 4 \text{ с}$$

Скорость передачи данных по линии связи равна 2^{12} бит/с. Сколько секунд потребуется на передачу файла размером 2 Кбайт?

Ответ: 4 с

Кодирование информации

§ 16. Сжатие данных

Зачем сжимать данные?

- сэкономить место в долговременной памяти (флэшки, жесткие диски)
- уменьшить время передачи по сети



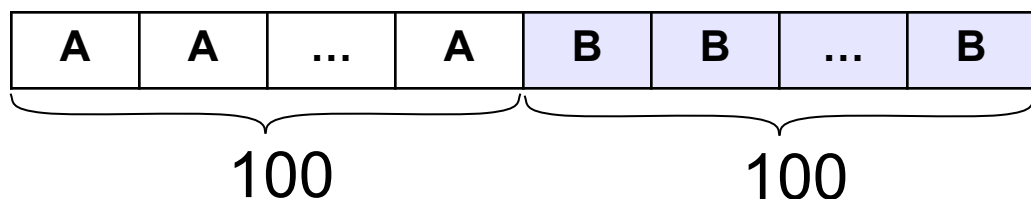
Цифровой звук и видео почти всегда сжаты!

Кодек – это программа для сжатия/распаковки цифрового звука или видео.

Пример алгоритма сжатия

Алгоритм RLE (англ. *Run Length Encoding*, кодирование цепочек одинаковых символов, используется для рисунков *.bmp).

Файл qq.txt



200 байт

Файл qq.rle (сжатый)

100	65	100	66
-----	----	-----	----

4 байта

Код буквы А

Код буквы В

$I_{сж}$

I_0

$$k_{сж} = \frac{I_0}{I_{сж}} = 50$$

Коэффициент сжатия – это отношение размера исходного файла к размеру сжатого файла.

Принцип сжатия



Почему удалось сжать в 100 раз?

Сжатие возможно, если в данных есть **избыточность**:

- повторяющиеся символы
- повторяющиеся цепочки символов
- другие закономерности

Сжатие «устраняет» **избыточность**.

Влга впдт в Кспске мре.

Хорошо сжимаются

- тексты
- рисунки
- несжатый звук
- несжатое видео

Плохо сжимаются

- программы
- случайные данные
- сжатые данные

Сжатие без потерь

Сжатие без потерь — это такое уменьшение объёма данных, при котором можно восстановить их исходный вид без искажений.

- Тексты, программы, числовые данные и т.п.
- Рисунки: **GIF**, **PNG**

Сжатие с потерями

Сжатие с
объёма
файл мо

- Рисунки
- Звук: **MP3**

ение
анный

WMV



100%
12 Кбайт



4 Кбайт 5 Кбайт



0%
2 Кбайт

Программы-архиваторы

Архивация — это создание файла-архива, который объединяет группу файлов.

Зачем?

- резервная копия данных
- объединить много файлов в один архив
- передать данные по сети
- зашифровать с паролем

Обычно **архивация + сжатие**.

Форматы архивов

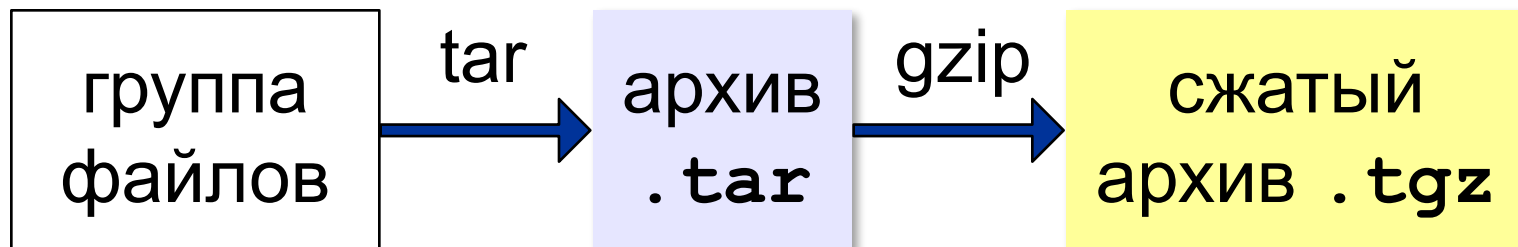


Сжатие без потерь!

Лучший для
Интернета

Форматы (расширения):

- **ZIP** – самый известный формат
- **RAR** – алгоритм Е. Рошала
- **7Z** – алгоритм программы 7Zip
- **TGZ** – алгоритм программы gzip



Программы-архиваторы

Windows – распаковка ZIP

бесплатно!



- свободная программа (с исходными кодами)
- кроссплатформенная
- упаковывает в форматах ZIP и 7Z
- распаковывает всё



WinRAR (www.rarlab.com)

- условно-бесплатная программа
- только для *Windows*
- упаковывает в форматах ZIP и RAR
- распаковывает всё

Программы-архиваторы

Linux



Ark



FileRoller

macOS



Stuffit Expander



BetterZip

Android



RAR



WinZip



ZArchiver

iOS



iZip Pro



WinZip

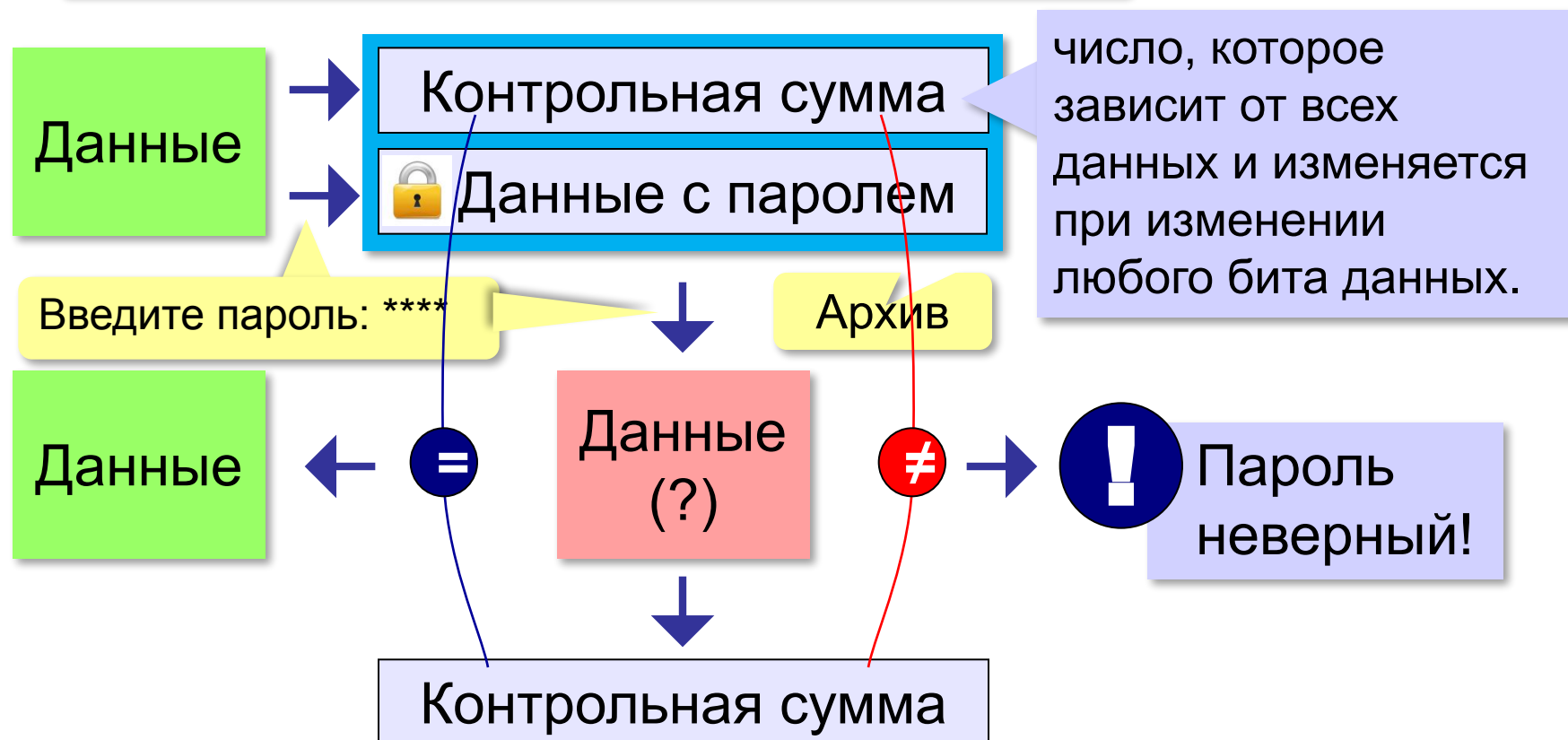


iFiles

Упаковка файлов с паролем

Как можно вскрыть пароль?

! Пароли в архиве не хранятся!



Упаковка с паролем

Как можно вскрыть пароль?

- догадаться (личные данные автора)
- подбор по словарю
- полный перебор вариантов...

Требования к паролю:

- не личные данные автора
- не осмысленное слово
- длина не менее 6 символов
- разные символы (заглавные и строчные буквы, цифры, другие знаки...)

Время на перебор паролей

Перебор 1000 вариантов в секунду.

	4 симв.	6 симв.	8 симв.	10 симв.
только цифры	10 с	17 мин	28 часов	117 дней
латинские строчные буквы	8 мин	87 часов	7 лет	4476 лет
латинские строчные буквы и цифры	28 мин	25 дней	89 лет	115936 лет
латинские строчные и заглавные буквы + цифры	4 часа	2 года	6923 лет	26 млн лет

Упаковка с паролем

Плохие пароли:

- телефон (9112345678)
- фамилия (ivanov)
- дата рождения (12052001)
- слово, которое есть в словаре (password)
- удобно набирать на клавиатуре (qwerty)
- одни цифры (12345)
- только строчные английские буквы

Хорошие пароли:

rH3cyJf198H

67B/C9QD7dJeY

55RuNUChMW666

Самораспаковывающиеся архивы (SFX)

SFX-архив — это файл (с расширением **.exe**), который содержит сжатые данные и программу для их распаковки.

около 15 Кбайт



- для распаковки не нужен архиватор
- может распаковать неквалифицированный пользователь



- увеличение размера файла
- опасность заражения вирусами

Конец фильма

ПОЛЯКОВ Константин Юрьевич

д.т.н., учитель информатики

ГБОУ СОШ № 163, г. Санкт-Петербург

kpolyakov@mail.ru

ЕРЕМИН Евгений Александрович

к.ф.-м.н., доцент кафедры мультимедийной

дидактики и ИТО ПГГПУ, г. Пермь

eremin@pspu.ac.ru

Источники иллюстраций

1. <http://s1.iconbird.com>
2. <https://sandstorm.deviantart.com>
3. <http://http://compression.ru>
4. <http://ru.wikipedia.org>
5. <https://www.khanacademy.org>
6. <https://www.kns.ru>
7. <http://nix.ru>
8. <http://http://www.computer-services.ru>
9. <http://http://www.masterna4as.com>
10. <http://blendercontest.com>
11. <http://http://geeky-gadgets.com>
12. авторские материалы