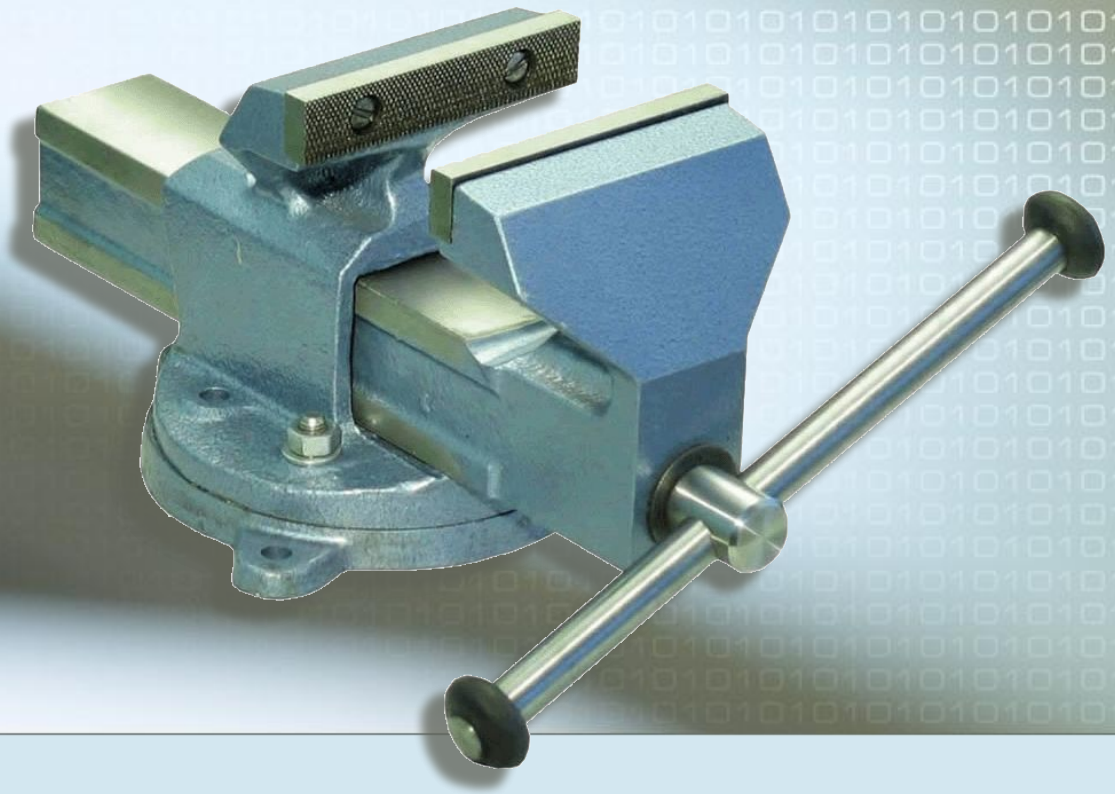
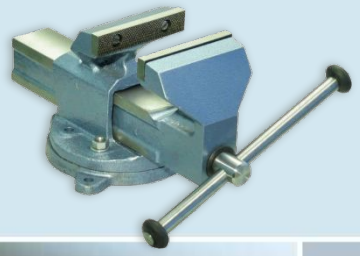


СЖАТИЕ

информации

Работа Ярцевой О.В.

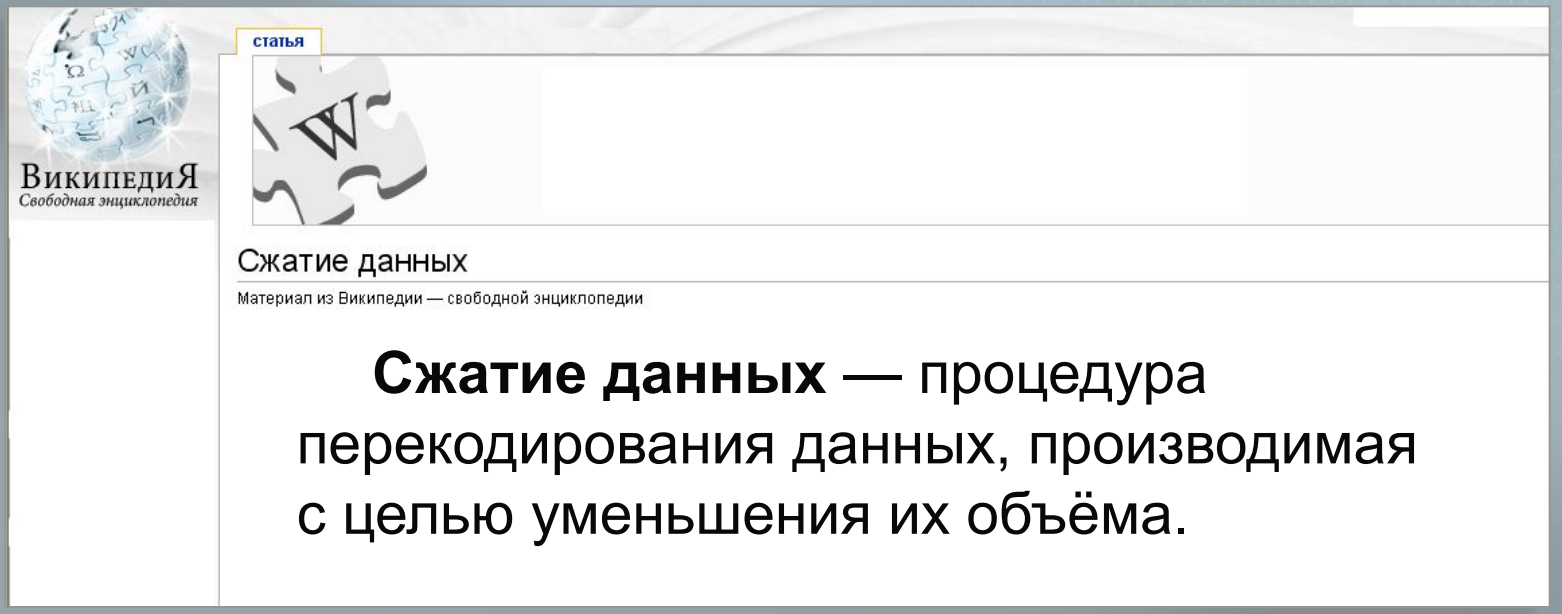


Сжатие текстовой информации

123	Использование цифр вместо слов
a || b, x ∈ [-2, 8]	Математические символы
д.ф.-м.н., проф. ЯрГУ	Общепринятые сокращения
imo, plz, ruok	Молодежный язык SMS
	Стенография

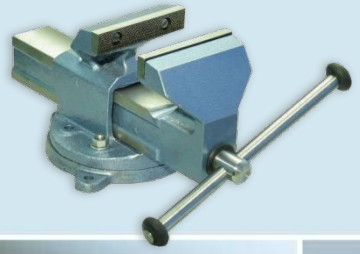
Основные понятия

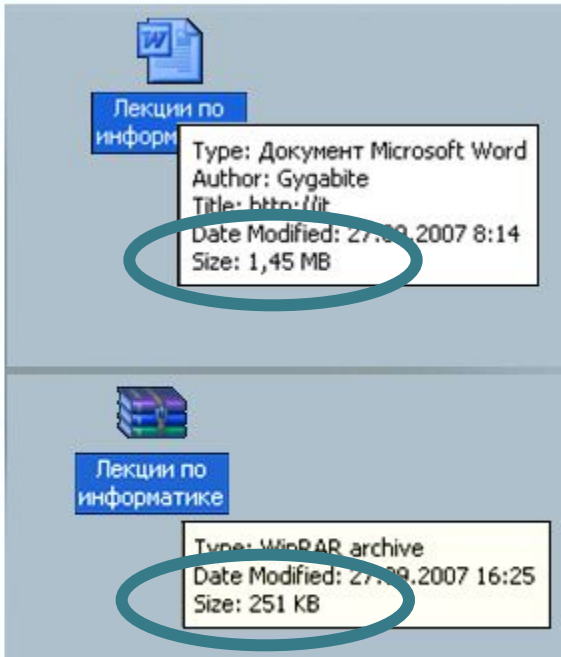
Кодирование информации является **избыточным**, если количество бит в полученном коде больше, чем это необходимо для однозначного декодирования исходной информации.



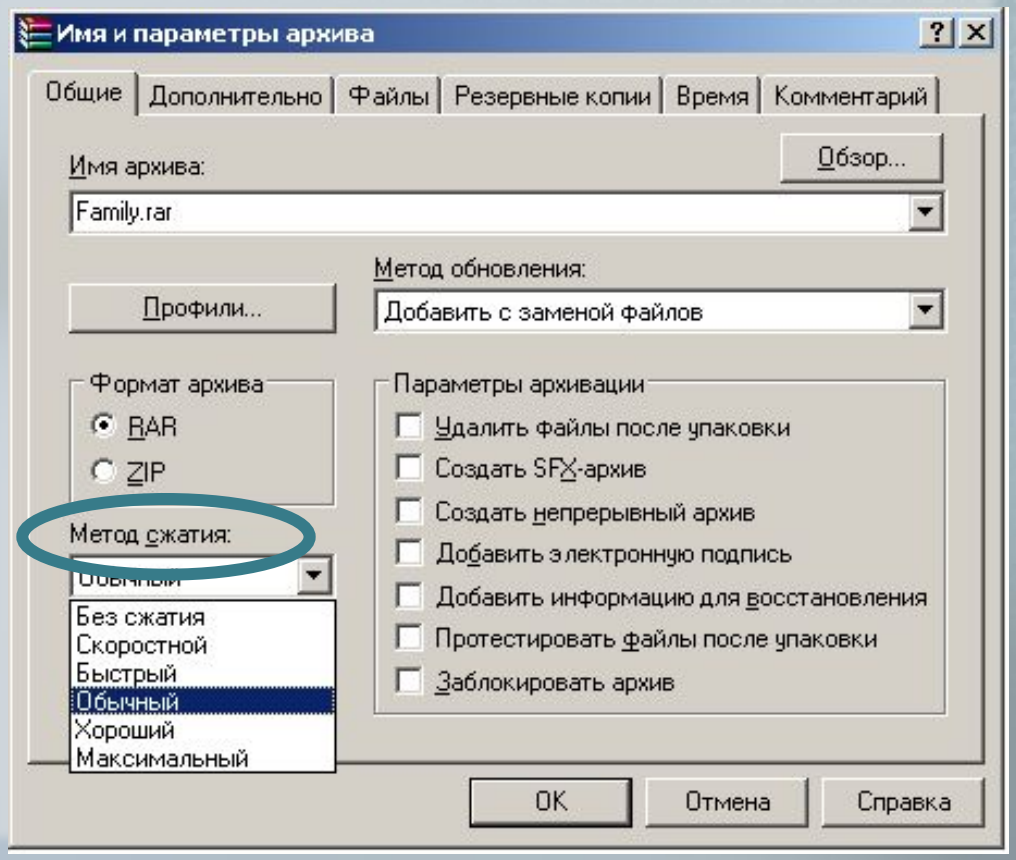
Декомпрессия - это способ восстановления сжатых данных в исходные.

Архиваторы





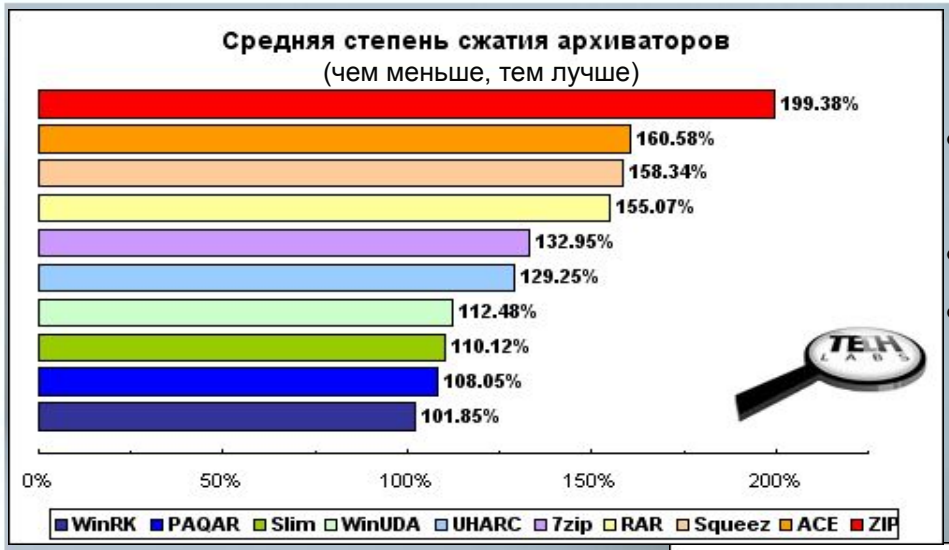
Под архиватором понимается программа-архиватор, формат архива и метод сжатия в комплексе.



Коэффициент сжатия:

$$\frac{1{,}45*1024}{251} \approx 5{,}92$$

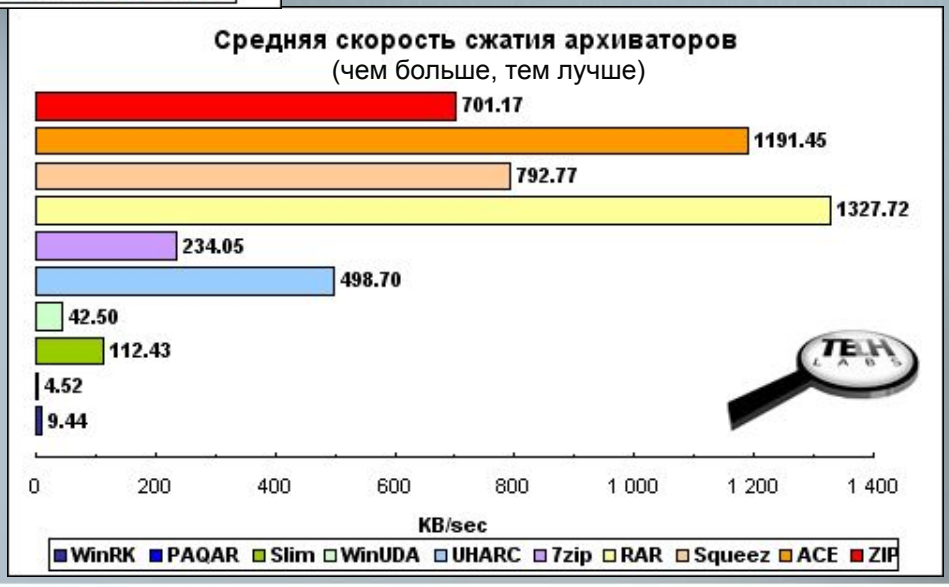
Сравнительные характеристики

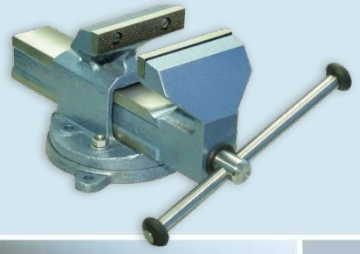


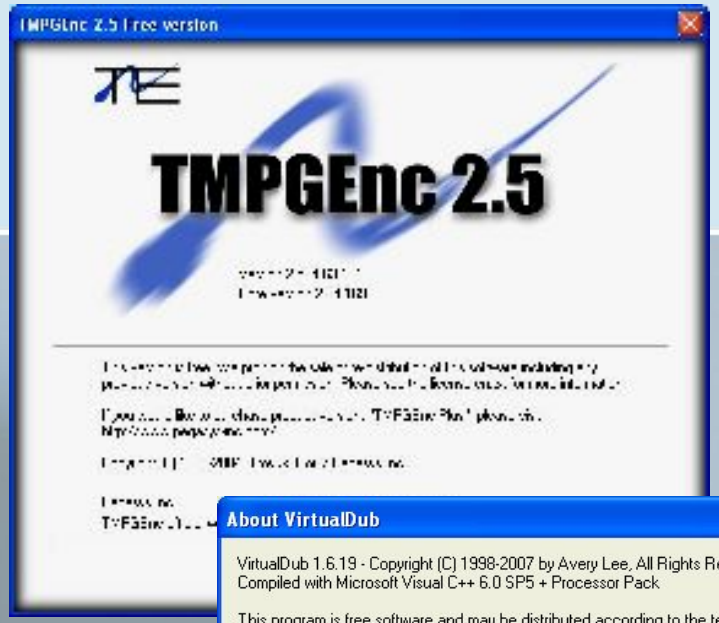
Степень сжатия зависит от

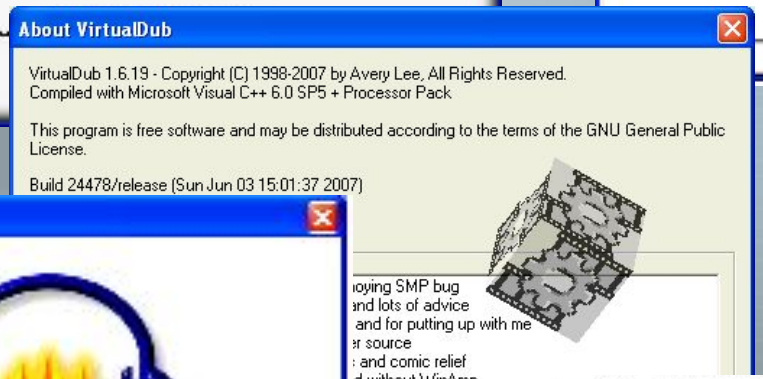
- используемого архиватора;
- метода сжатия;
- типа исходного файла.

Архиваторы **ACE**, **RAR** и **Squeez** имеют близкие результаты с небольшим преимуществом по степени сжатия у RAR, и при высокой скорости сжатия у Squeez.

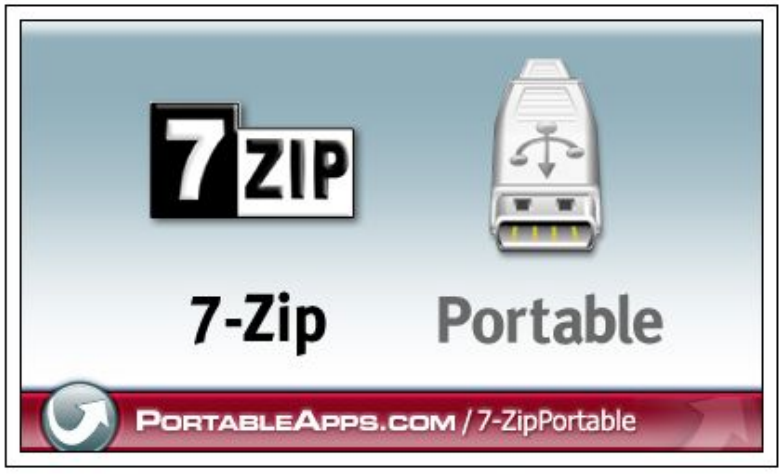


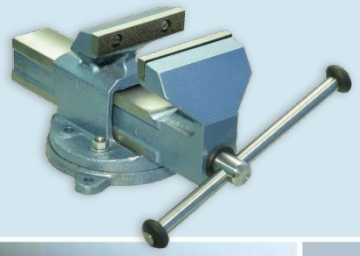






Бесплатные программы для обработки аудио- и видеоинформации





Виды сжатий

Сжатие

с потерями

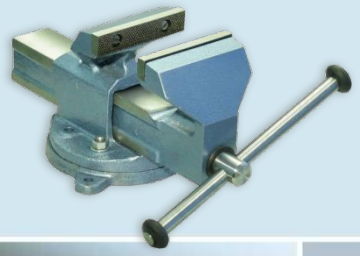
Восстановление возможно с искажениями, малозаметными для человеческого глаза или уха.

Форматы файлов: jpg, mpeg, adpcm .

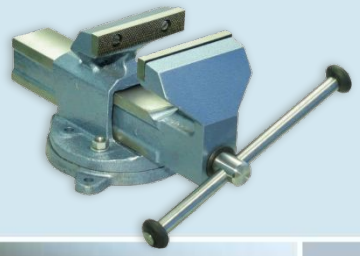
без потерь

Возможно восстановление исходных данных без искажений.

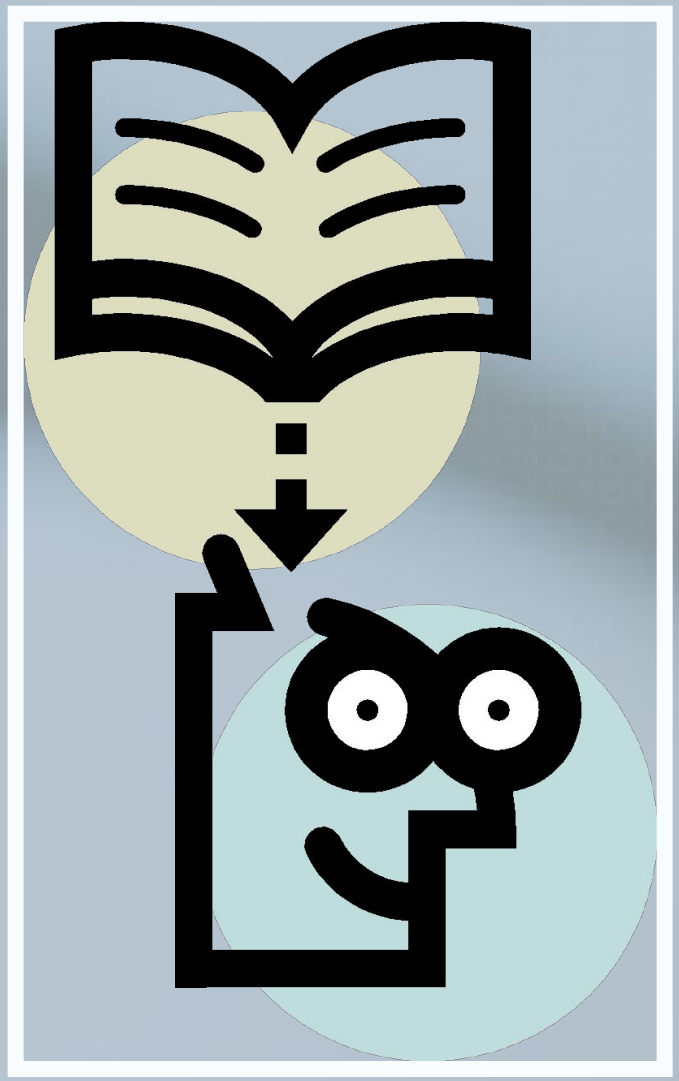
Форматы файлов: gif, tif, png, pcx, avi, zip, rar, arj.



Сжатие с потерей качества



Сжатие без потерь
Алгоритм **RLE**



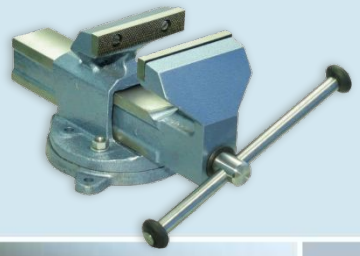
от англ. **R**un **L**ength **E**ncoding

В файле записывается, сколько раз повторяются одинаковые байты.

Схематично:

"RRRRRGGGBBBBBBRRRB
BRRRRRRR"

"5R3G6B3R2B7R".

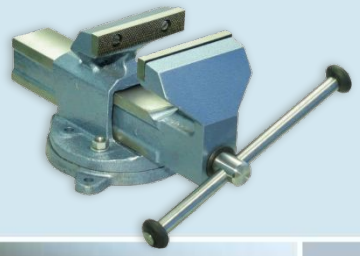


ASCII-коды

В восьмиразрядной таблице символьной кодировки ASCII каждый символ кодируется восемью битами и, следовательно, занимает в памяти 1 байт.

Знак, клавиша	*Двоичный код*	*10-ый код*
пробел	**00100000**	32
A (лат.)	**01000001**	65
B (лат.)	**01000010**	66
Z	**01011010**	90
0	**00110000**	48
1	**00110001**	49
9	**00111001**	57
Клавиша Esc	**00011011**	27
Клавиша Enter	**00001101**	13

Некоторые ASCII-коды



Сжатие без потерь
Метод упаковки

Пример №1

0	1	2	3	4	5	6	7	8	9
0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

1010	1011	1100	1101	1110	1111

Двукратное сжатие.
Формат BCD –
Binary **C**oded **D**ecimal

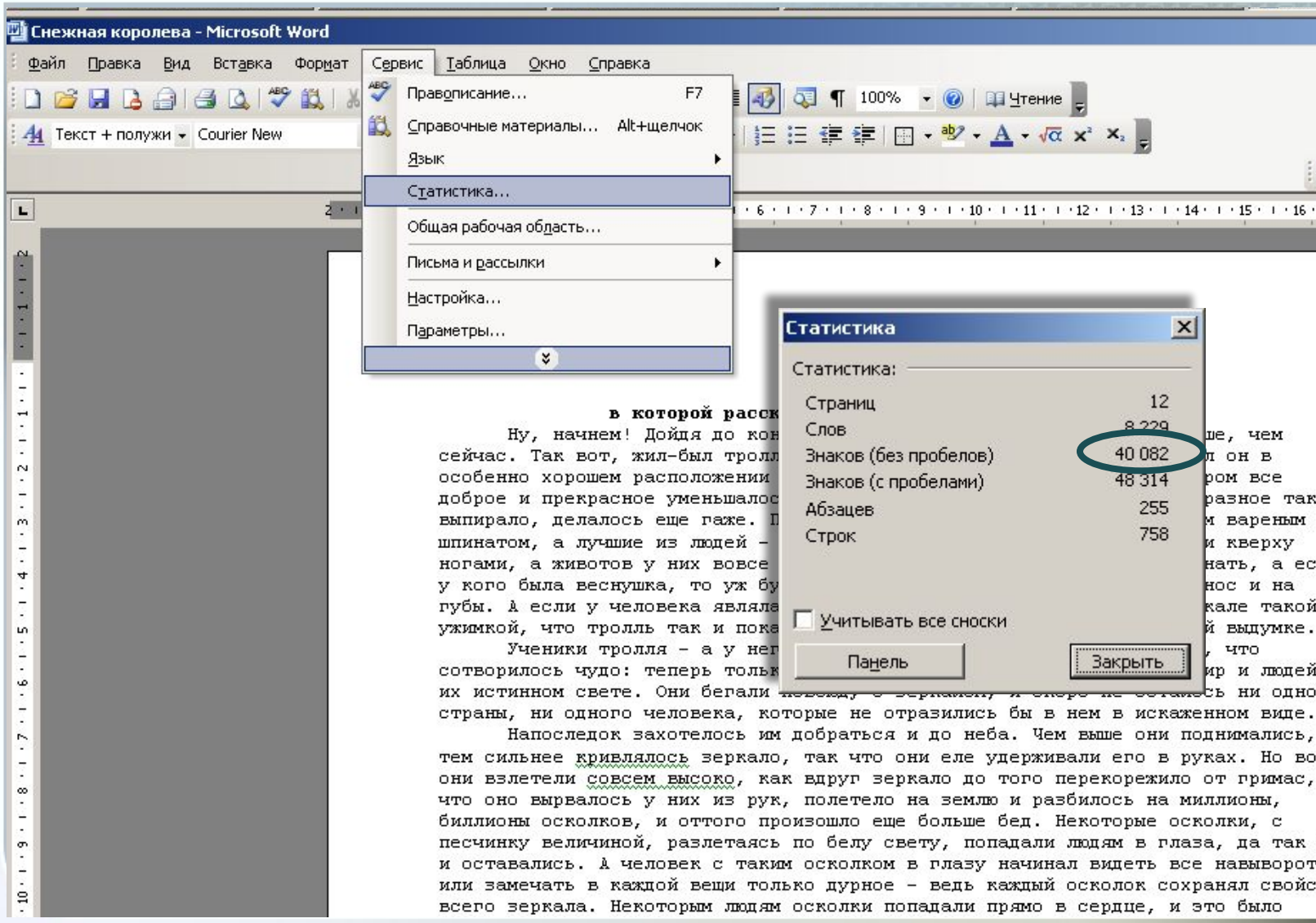
Пример №2

TO BE OR NOT TO BE?

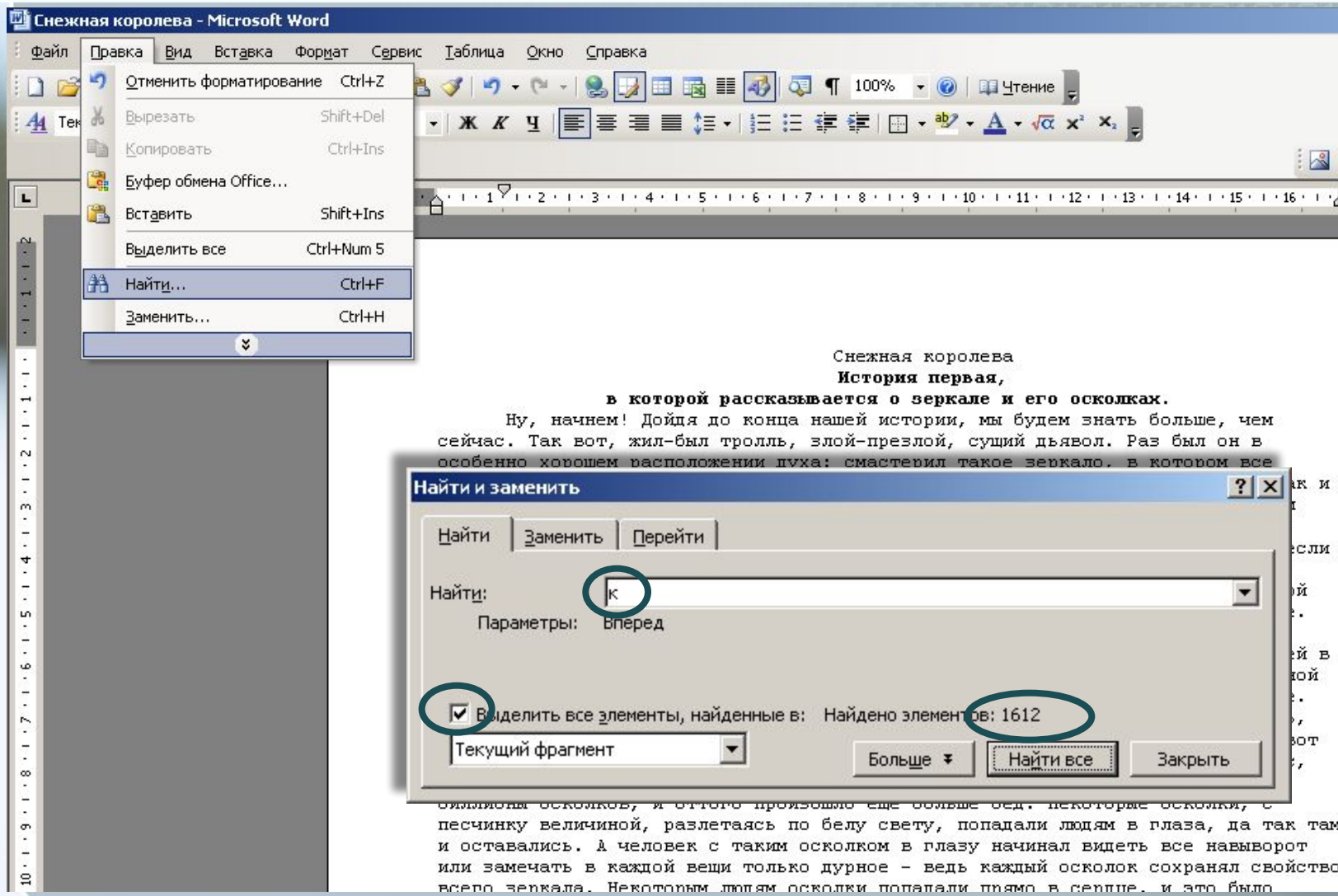
T	O	B	E	R	N	пробел	?
000	001	010	011	100	101	110	111

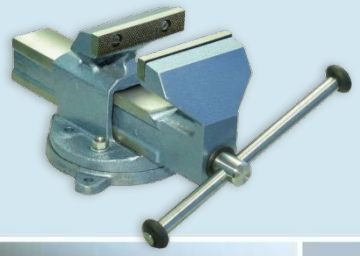
19 символов в предложении: 3*19=57 бит=8 байт
Коэффициент сжатия: 19/8≈ 2,4

Практическая работа



Практическая работа

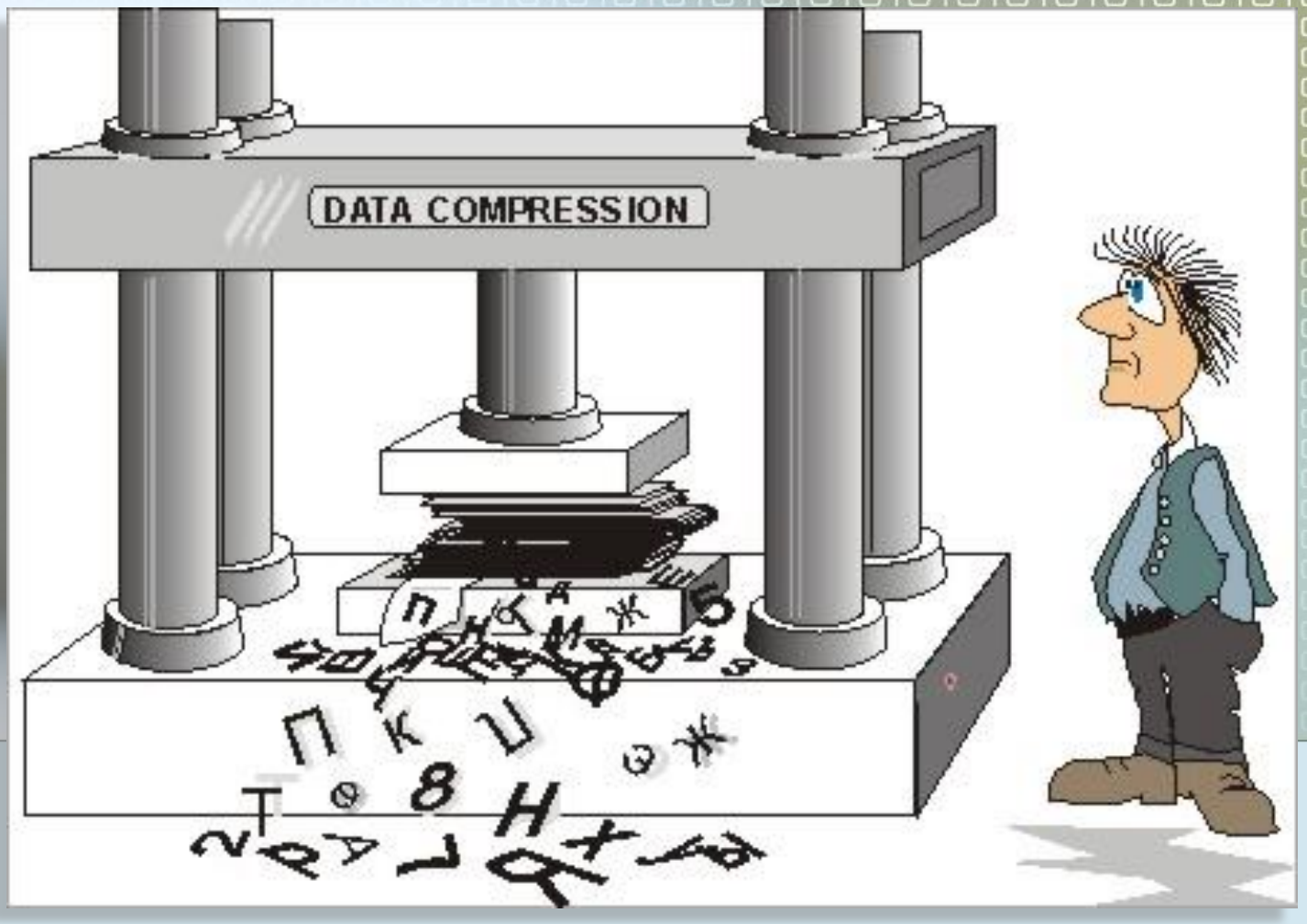




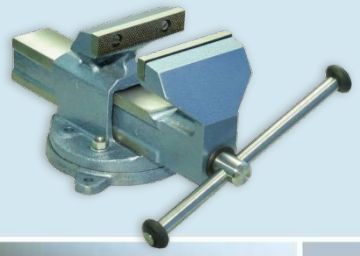
Практическая работа

“Частотный анализ букв русского языка”

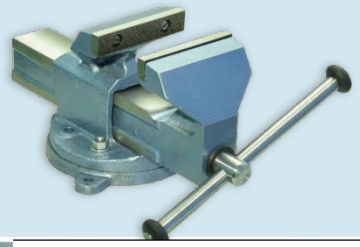
1. Открыть с помощью Microsoft Office Word документ Skazka.doc.
2. Подсчитать количество каждой из букв русского алфавита в тексте и заполнить таблицу Alphabet.xls в Microsoft Office Excel. Вычислить частоту встречаемости букв.
3. Упорядочить данные таблицы по убыванию частот.
4. Построить график распределения частот букв.



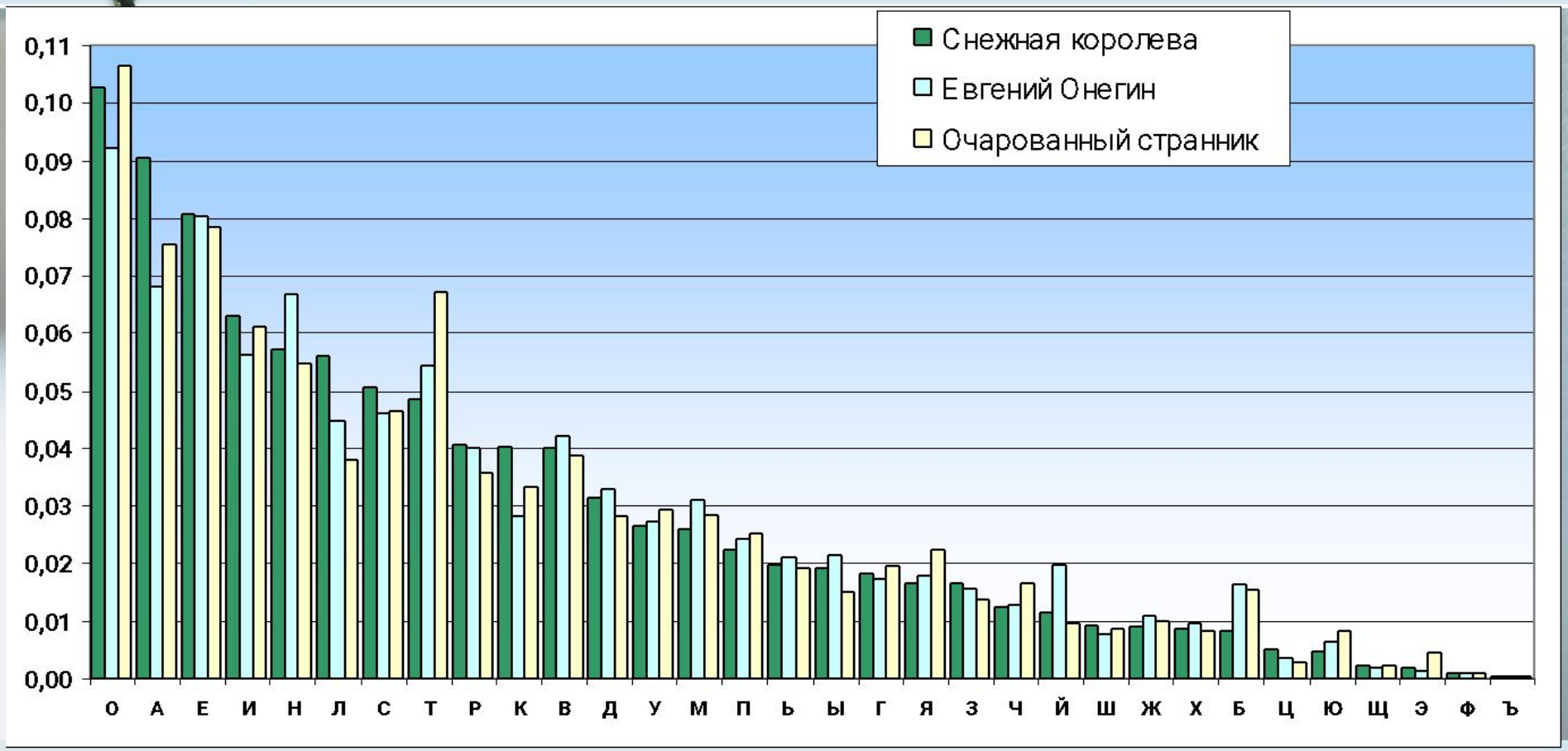
АЛГОРИТМ Хаффмана



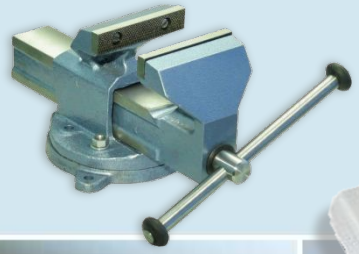
Частотный анализ



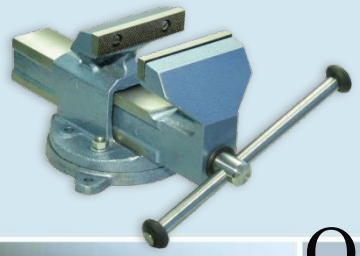
Сравнительный частотный анализ



«Анна Каренина»	оеанитслврккдмупяьыгбчзжйшхэющцфъ	280 тыс. слов
Солженицын А.И.	оеаинтсвлрлдмпуьяыгбзчйхжшюцщэфъ	86 тыс. слов
Новости	оеаинтсрвлкдмпуяьыгзбчйжхюшцщэфъё	25 тыс. слов



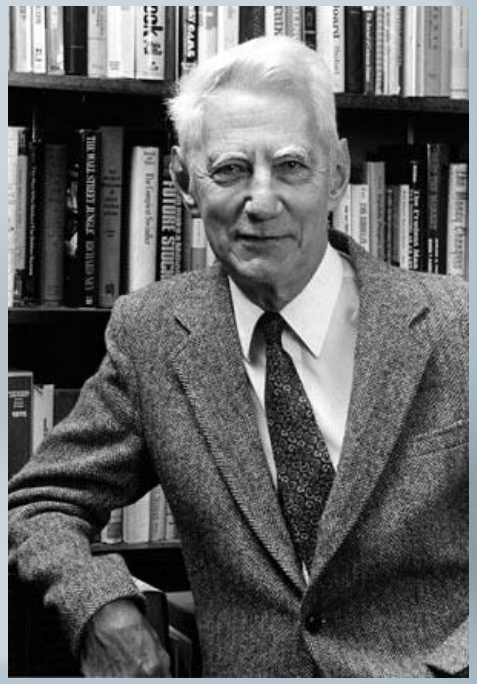
В тексте, написанном
на русском языке,
в каждой тысяче
символов в среднем
будет 90 букв "о",
72 - "е" и только 2 -"ф".
Больше же всего
окажется пробелов:
174.



Алгоритм Шеннона-Фано и Хаффмана

Оба этих алгоритма используют коды переменной длины: часто встречающийся символ кодируется двоичным кодом меньшей длины, редко встречающийся — кодом большей длины. Коды Шеннона-Фано и Хаффмана — префиксные, то есть никакое кодовое слово не является началом любого другого. Это свойство позволяет однозначно декодировать любую последовательность кодовых слов. В отличие от алгоритма Шеннона-Фано, алгоритм Хаффмана обеспечивает минимальную избыточность, то есть минимальную длину кодовой последовательности при побайтном кодировании.

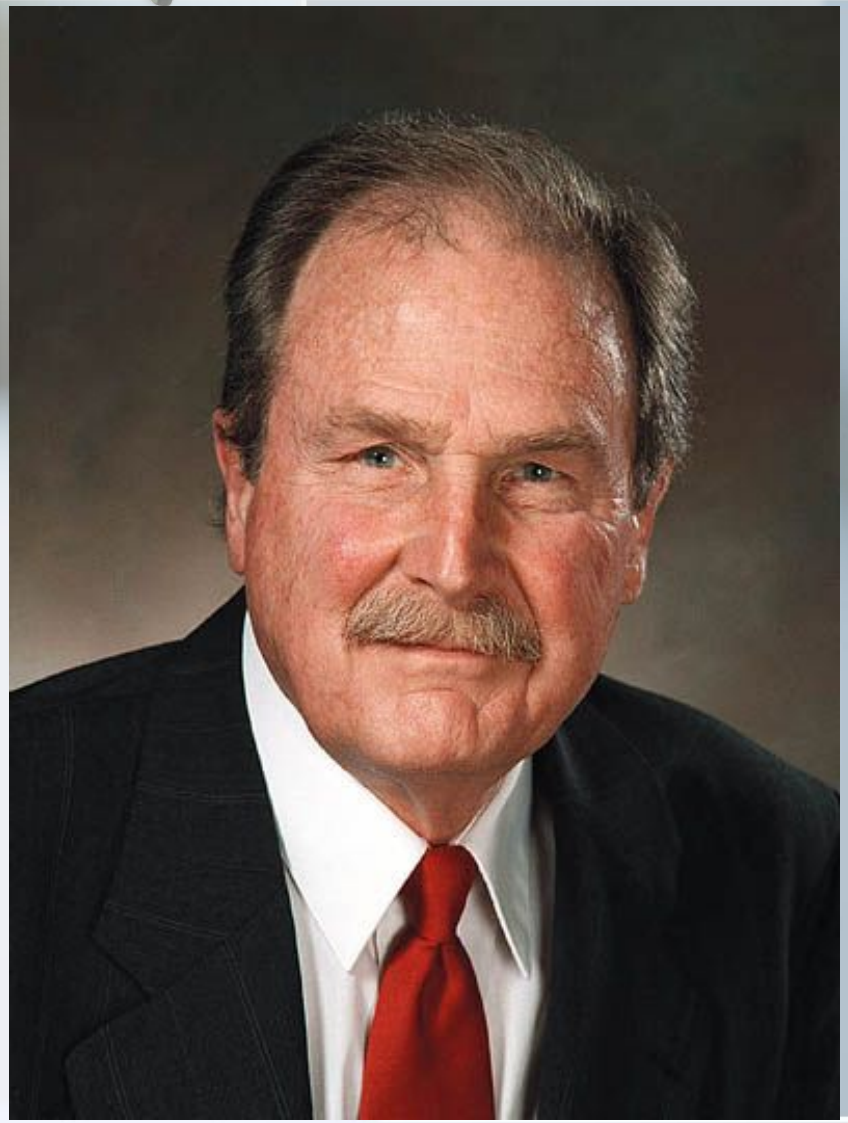
Алгоритм Шеннона-Фано



К.Шеннон и Р.Фано сформулировали алгоритм сжатия, который использует коды переменной длины.

	%		%		%
A	8.1	K	0.4	U	2.4
B	1.4	L	3.4	V	0.9
C	2.7	M	2.5	W	1.5
D	3.9	N	7.2	X	0.2
E	13.0	O	7.9	Y	1.9
F	2.9	P	2.0	Z	0.1
G	2.0	Q	0.2		
H	5.2	R	6.9		
I	6.5	S	6.1		
J	0.2	T	10.5		

Дэвид Хаффман



David Huffman

(1925-1999)

В 18 лет Дэвид получил степень бакалавра электротехники в университете штата Огайо. Основную концепцию кодирования данных Хаффман разработал во время Второй мировой войны, когда служил на эсминце офицером-связистом. Изначально алгоритм предназначался для кодирования радиосообщений.

За свою деятельность он получил множество наград за исключительный вклад в теорию информации.

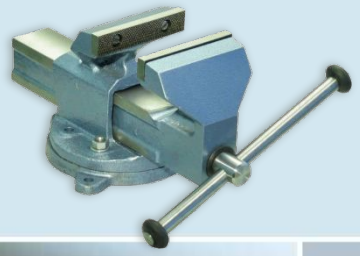
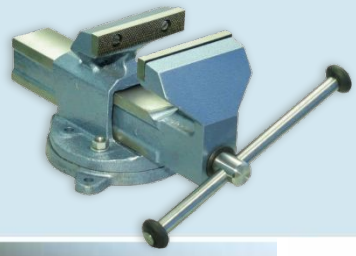


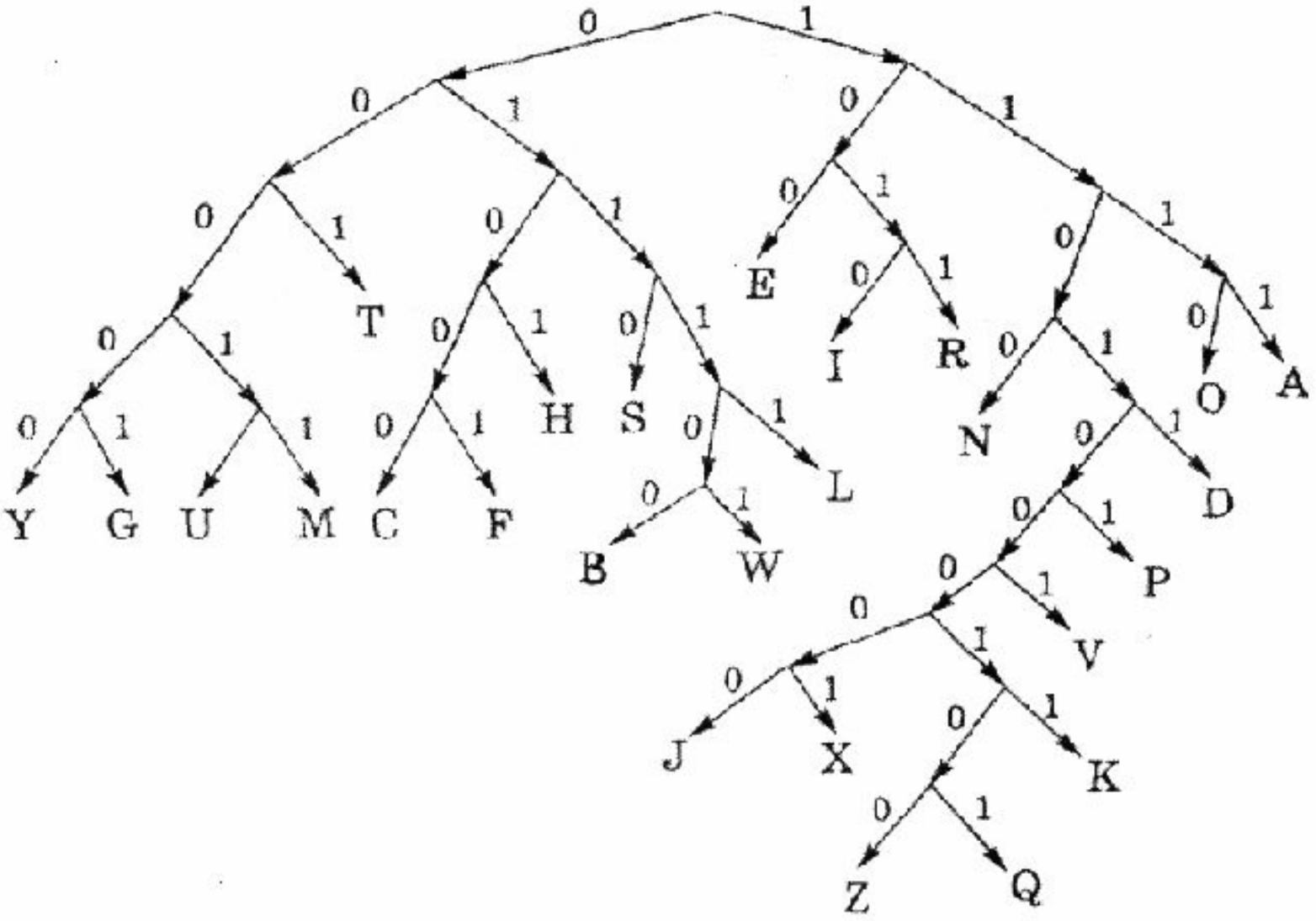
Таблица кодов Хаффмана

Буква	Код Хаффмана	Буква	Код Хаффмана	Буква	Код Хаффмана
E	100	**D**	11011	**W**	011101
T	001	**L**	01111	**B**	011100
A	1111	**F**	01001	**V**	1101001
O	1110	**C**	01000	**K**	110100011
N	1100	**M**	00011	**X**	110100001
R	1011	**U**	00010	**J**	110100000
I	1010	**G**	00001	**Q**	1101000101
S	0110	**Y**	00000	**Z**	1101000100
H	0101	**P**	110101		

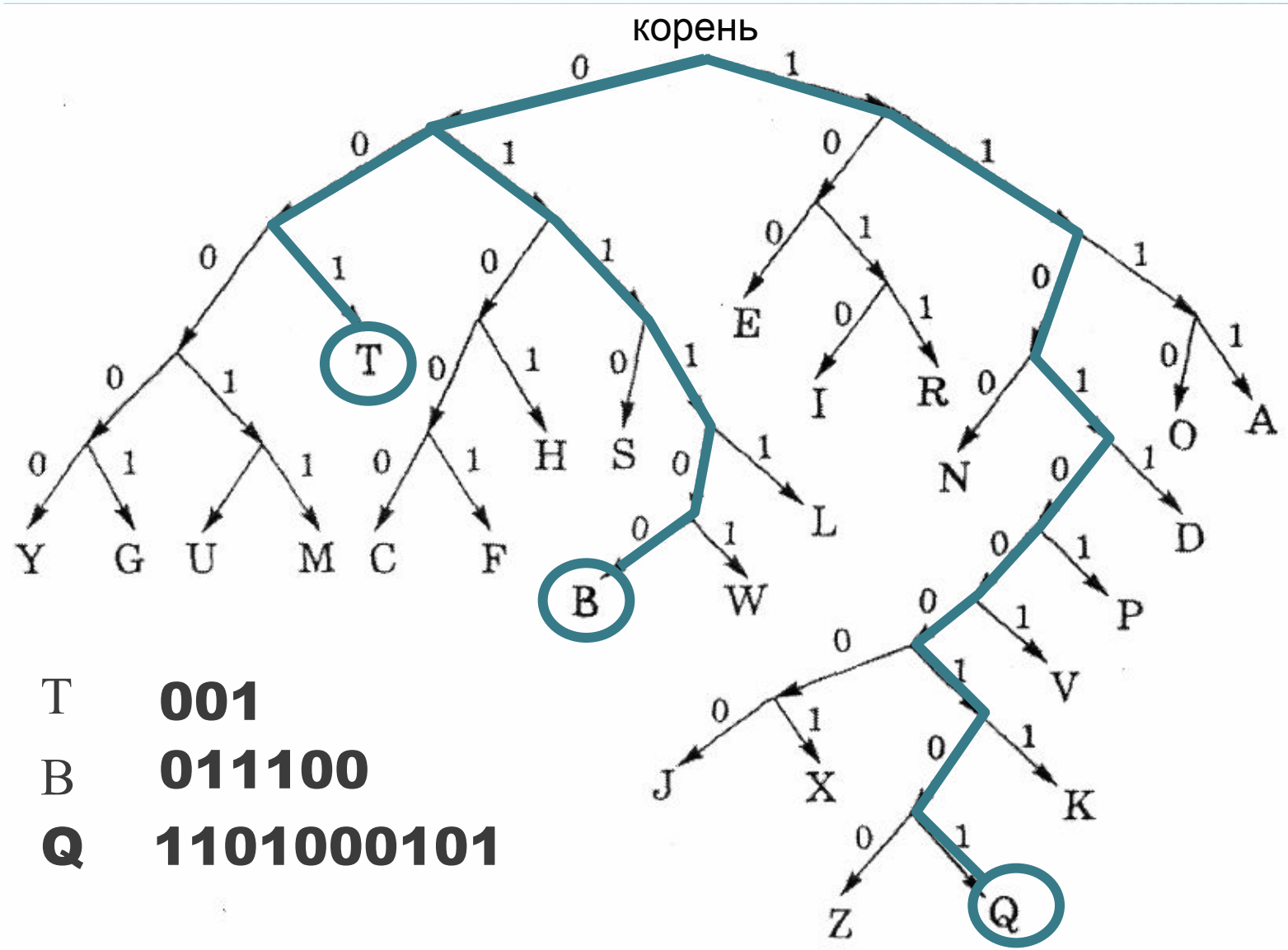


Двоичное дерево

Двоичным (бинарным) называется дерево, из каждой вершины которого выходят две ветви.

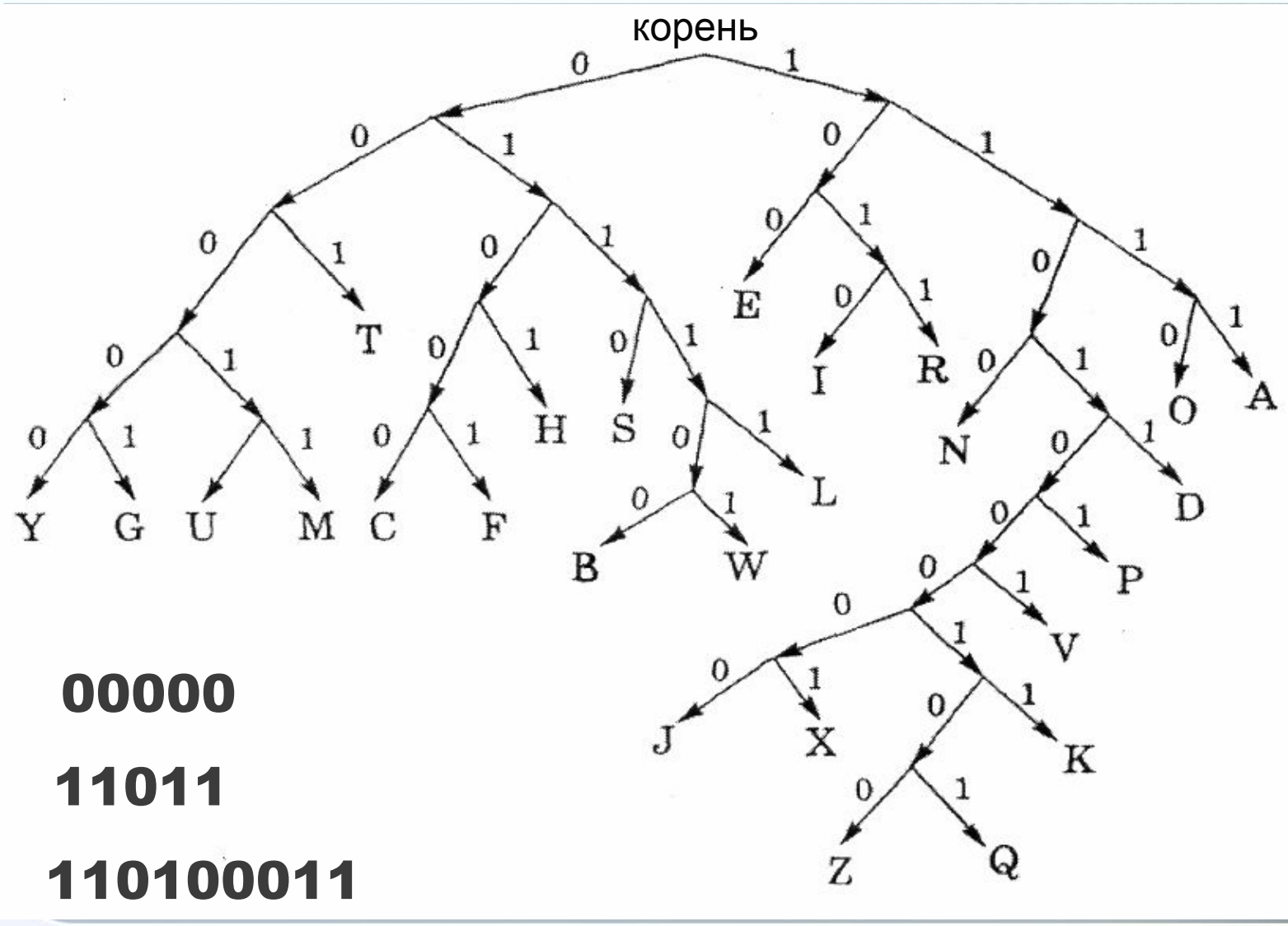


Дерево Хаффмана



T	**001**
B	**011100**
Q	**1101000101**

Дерево Хаффмана

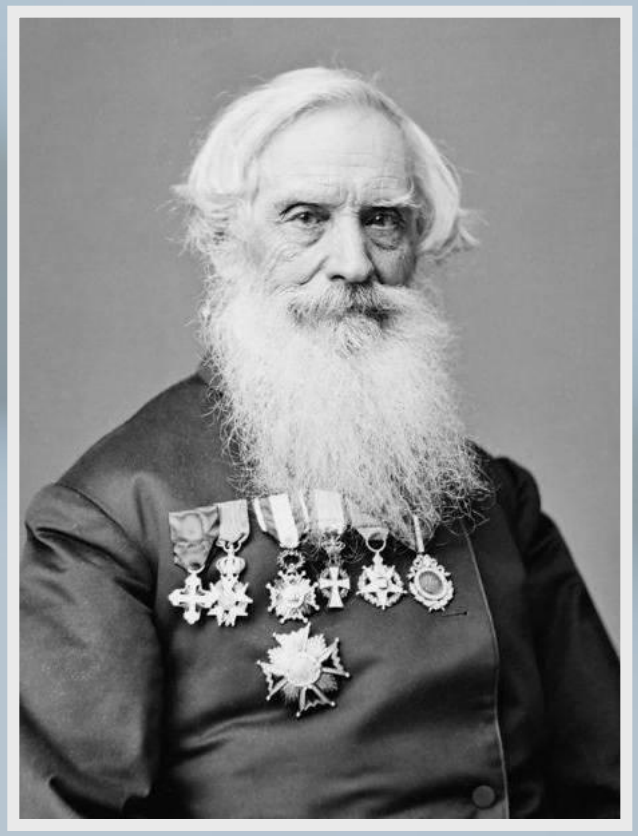


00000

11011

110100011

Азбука Морзе



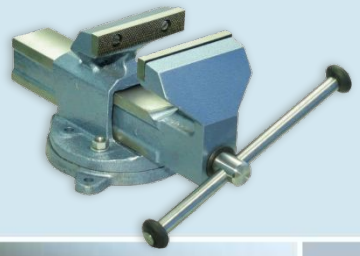
Сэмюэль Морзе (1791-1872) - американский изобретатель и художник

INTERNATIONAL MORSE CODE

1. A dash is equal to three dots.
2. The space between parts of the same letter is equal to one dot.
3. The space between two letters is equal to three dots.
4. The space between two words is equal to five dots.

A	• —	U	• • —
B	— • • •	V	• • • —
C	— • — •	W	• — —
D	— • •	X	— • • —
E	•	Y	— • — —
F	• • — •	Z	— — • •
G	— — •		
H	• • • •		
I	• •		
J	• — — —		
K	— • —		
L	• — • •		
M	— —		
N	— •		
O	— — —		
P	• — — •		
Q	— — • —		
R	• — •		
S	• • •		
T	—		

1	• — — — —
2	• • — — —
3	• • • — —
4	• • • • —
5	• • • • •
6	— • • • •
7	— — • • •
8	— — — • •
9	— — — — •
0	— — — — —

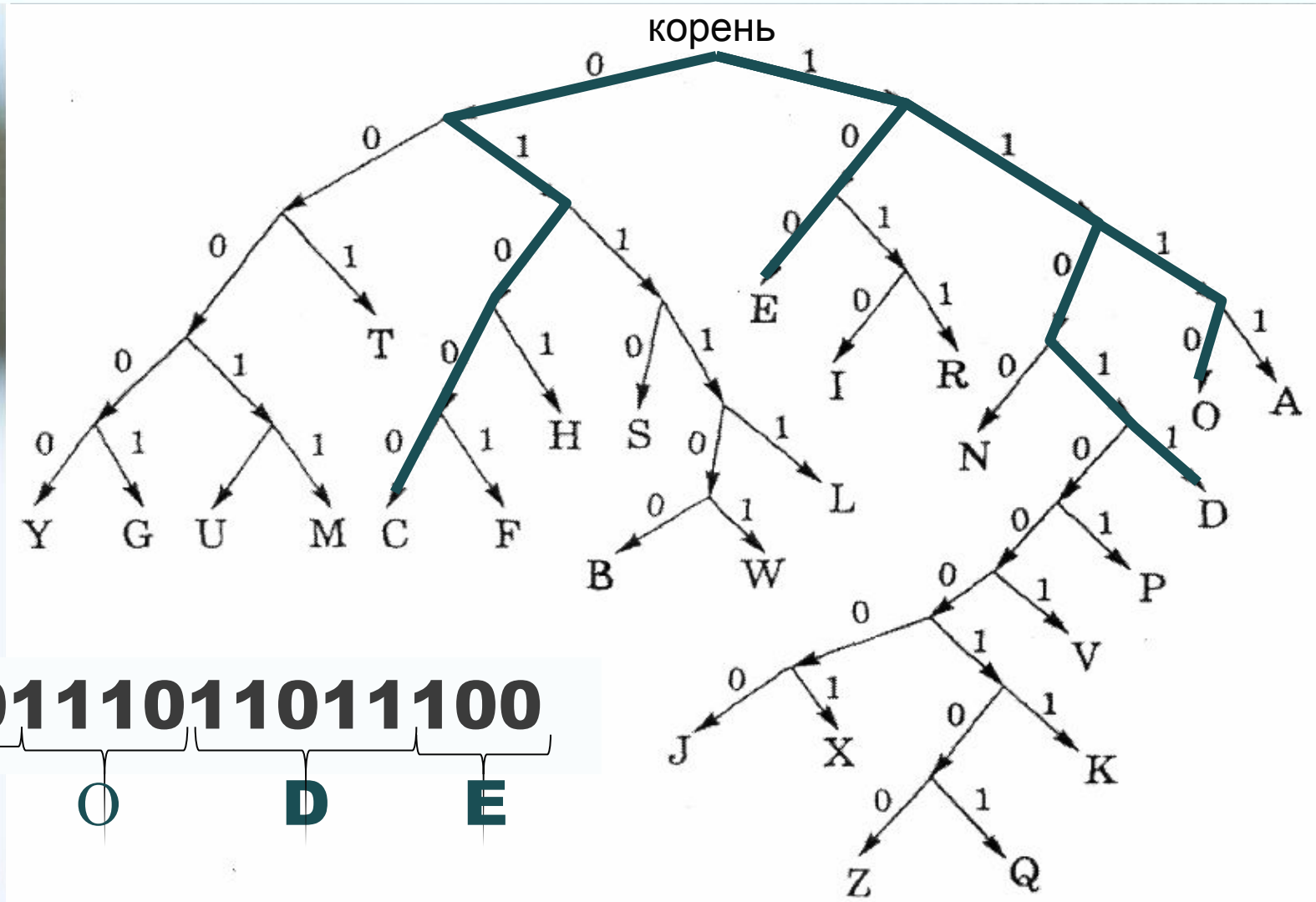


Азбука Морзе

А	●▬	П	●▬▬●	Ь	▬●●▬
Б	▬●●●	Р	●▬●	Ы	▬●▬
В	●▬▬	С	●●●	Й	●▬▬▬
Г	▬▬●	Т	▬		
Д	▬●●	У	●●▬●	1	●▬▬▬▬
Е	●	Ф	●●▬●	2	●●▬▬▬
Ж	●●●▬	Х	●●●●	3	●●●▬▬
З	▬▬●●	Ц	▬●▬●	4	●●●●▬
И	●●	Ч	▬▬▬●	5	●●●●●
К	▬●▬●	Ш	▬▬▬▬	6	▬●●●●
Л	●▬●●	Щ	▬▬●▬	7	▬▬●●●
М	▬▬	Э	●●▬●●	8	▬▬▬●●
Н	▬●	Ю	●●▬▬	9	▬▬▬▬●
О	▬▬▬	Я	●▬●▬	0	▬▬▬▬▬

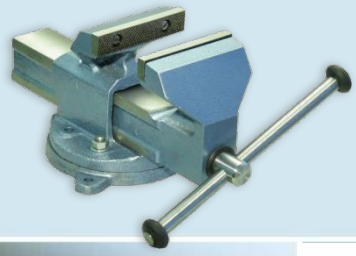
с т е к ж а р е ф т а е

Дерево Хаффмана

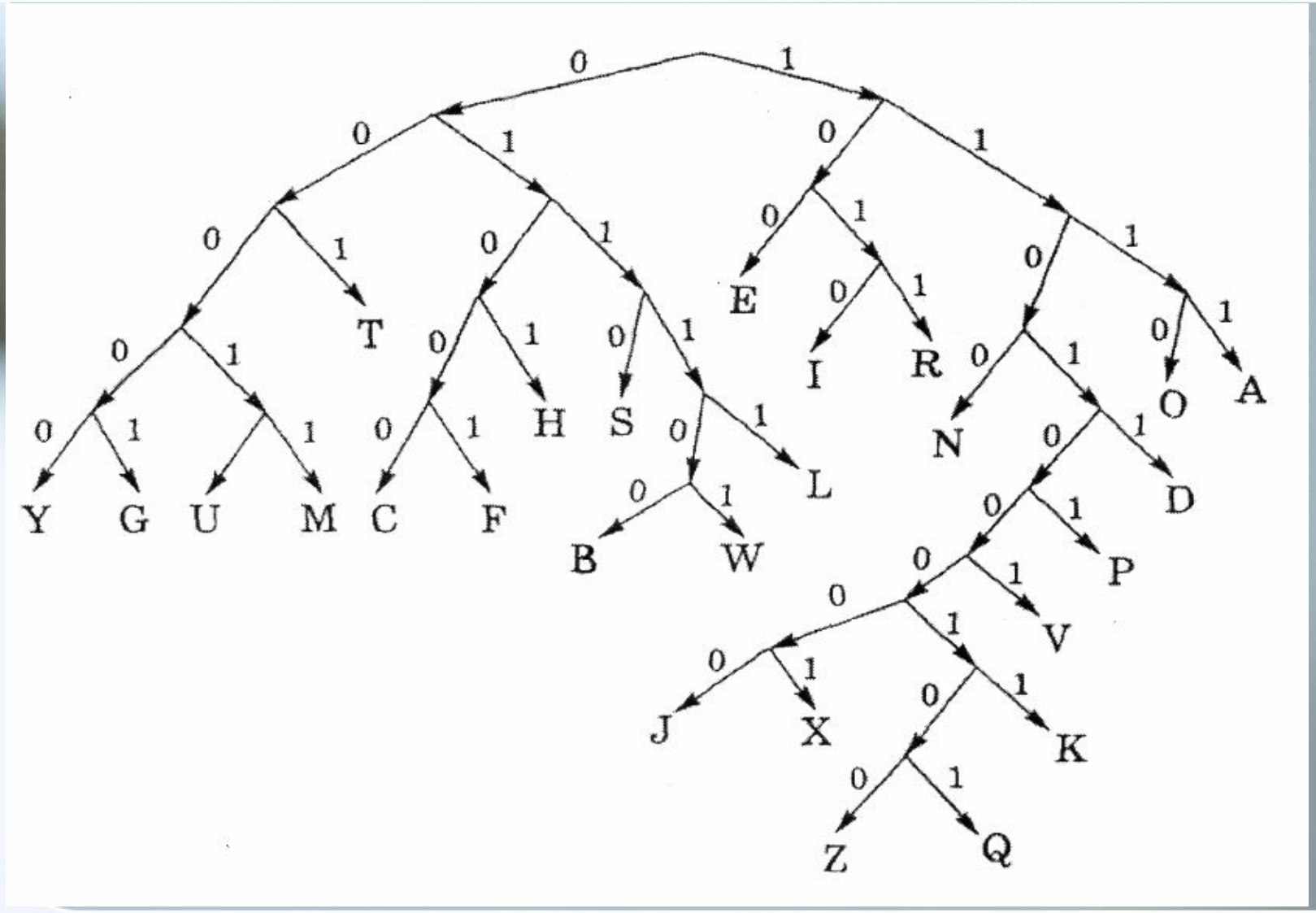


01000111011011100

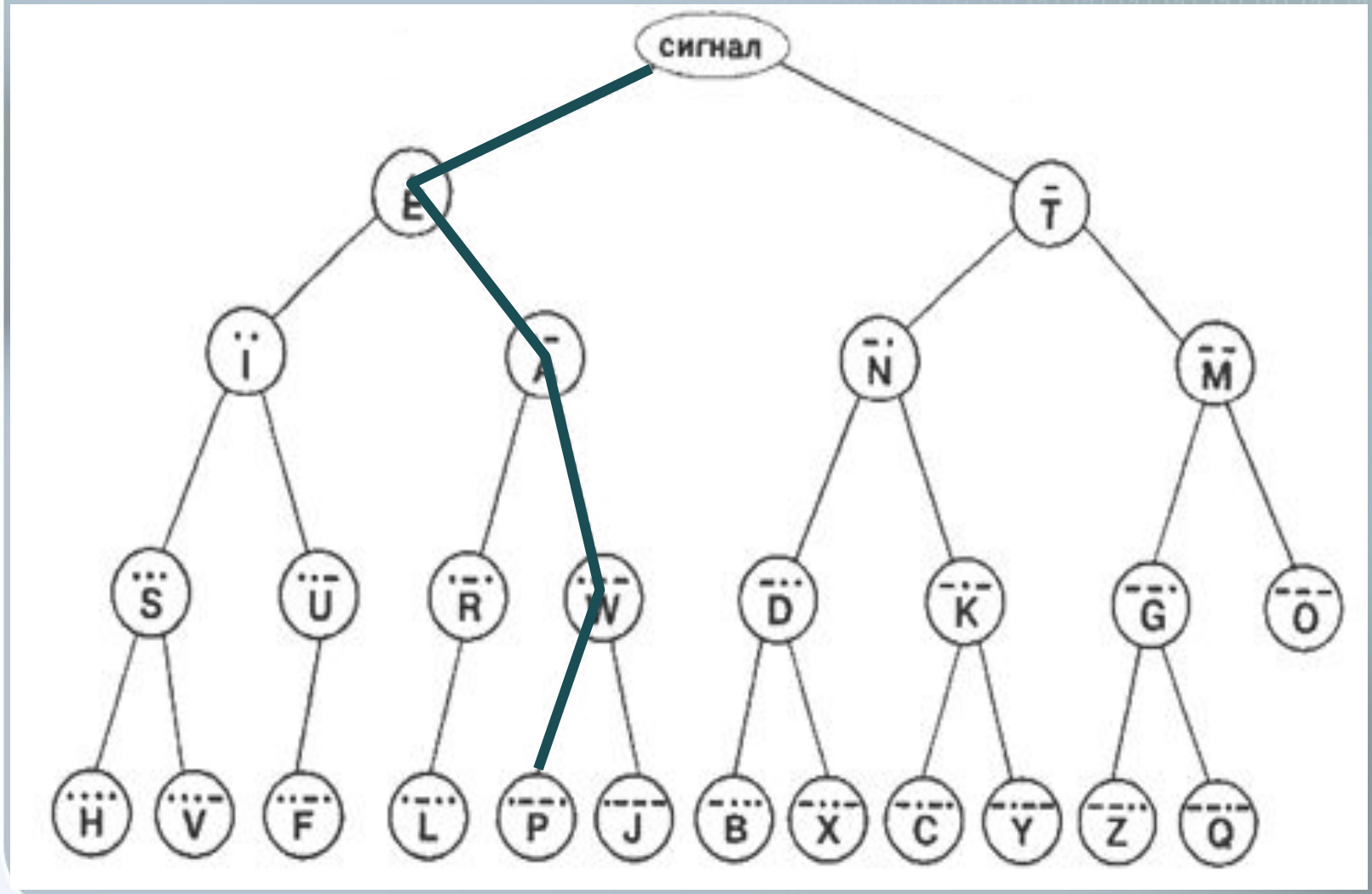
C O D E

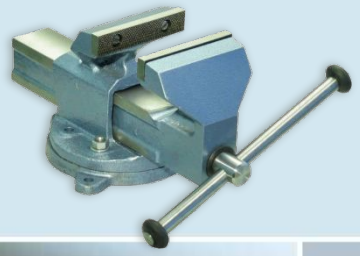


Код Хаффмана обладает свойством **префиксности**, то есть код никакого символа не является началом кода какого-либо другого символа.



Дерево азбуки Морзе





Алгоритм Хаффмана

Алгоритм Хаффмана двухпроходный:

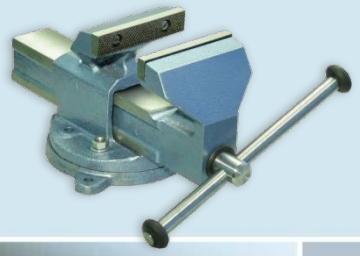
на **первом** проходе строится частотный словарь и генерируются коды;

на **втором** проходе происходит непосредственно кодирование.

НА ДВОРЕ ТРАВА, НА ТРАВЕ ДРОВА

1. Подсчитать частоту встречаемости (вес) каждого символа.

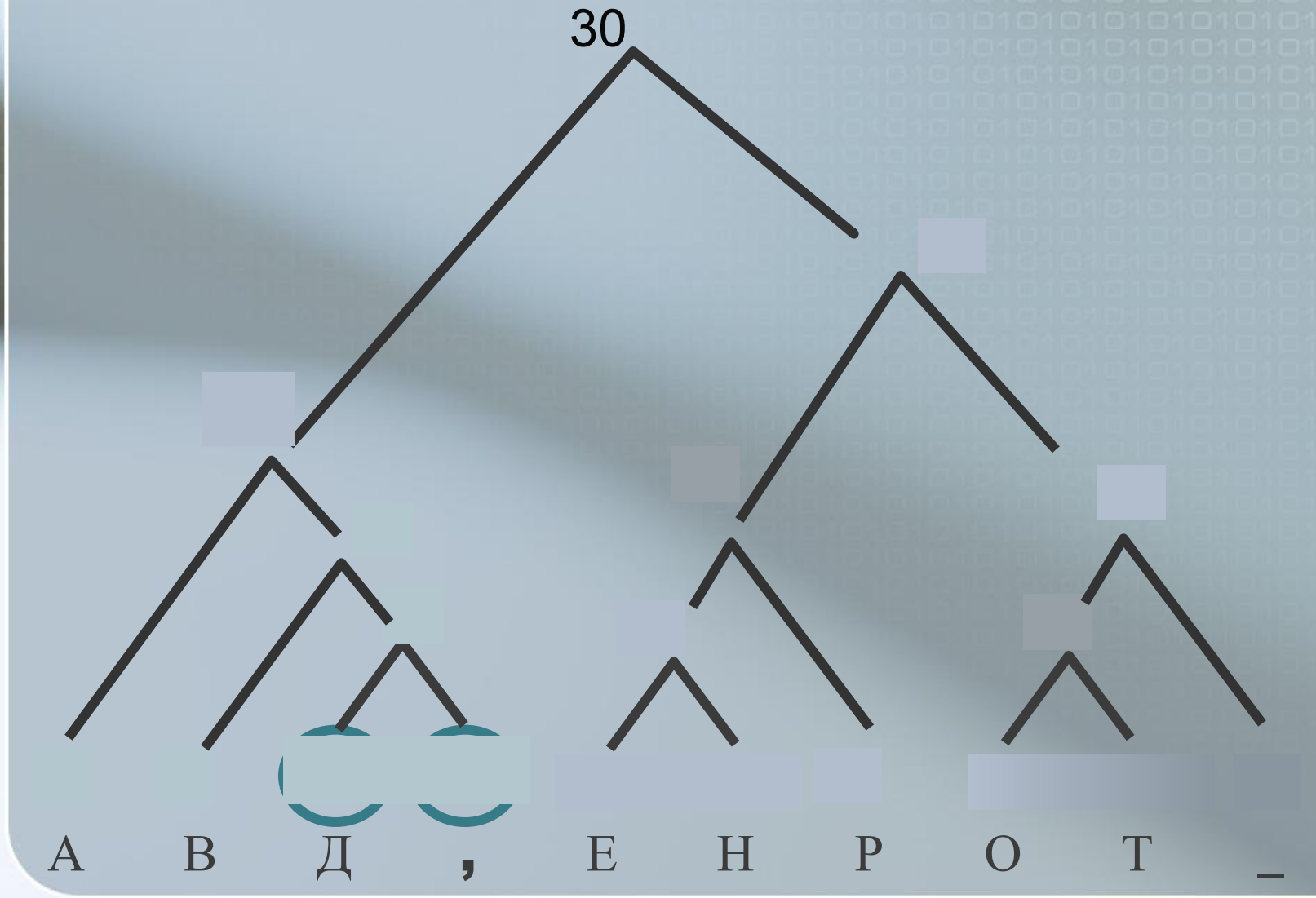
символ	А	В	Д	,	Е	Н	Р	О	Т	_
вес	6	4	2	1	2	2	4	2	2	5

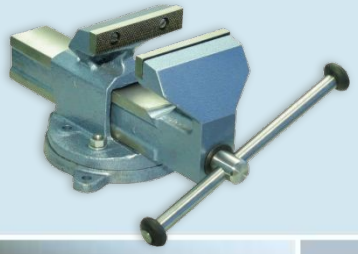


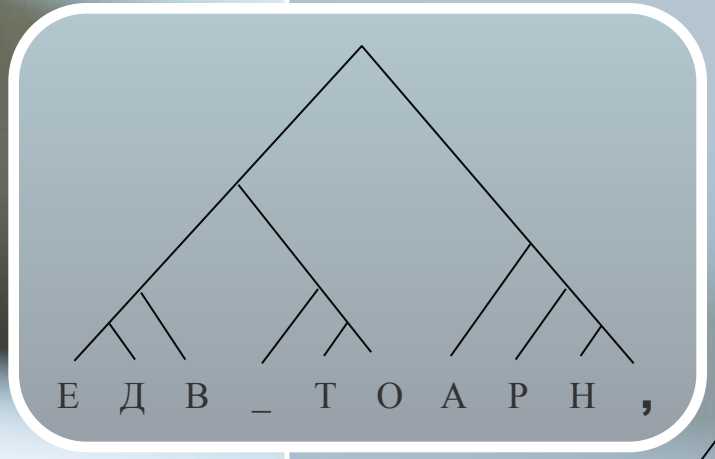
Алгоритм построения дерева Хаффмана

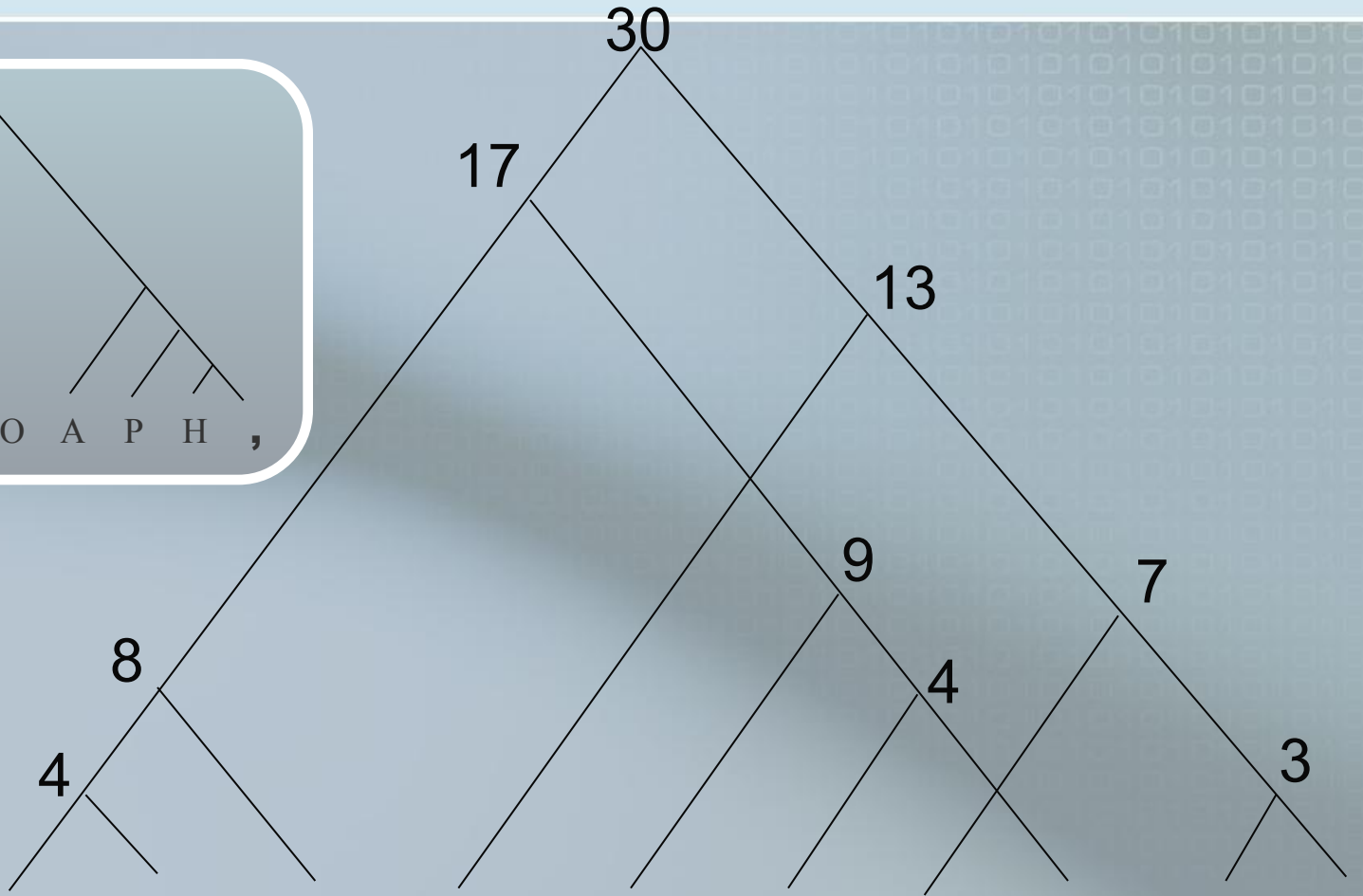
1. Среди символов выбрать два с наименьшими весами (если таких пар несколько, выбирается любая из них).
2. Свести ветки дерева от этих двух символов в одну точку с весом, равным сумме двух весов, при этом веса самих элементов стираются.
3. Повторять пункты 1 и 2 до тех пор, пока не останется одна вершина с весом, равным сумме весов исходных символов.
4. Пометить одну ветку нулём, а другую - единицей (по собственному усмотрению).

Построение дерева Хаффмана



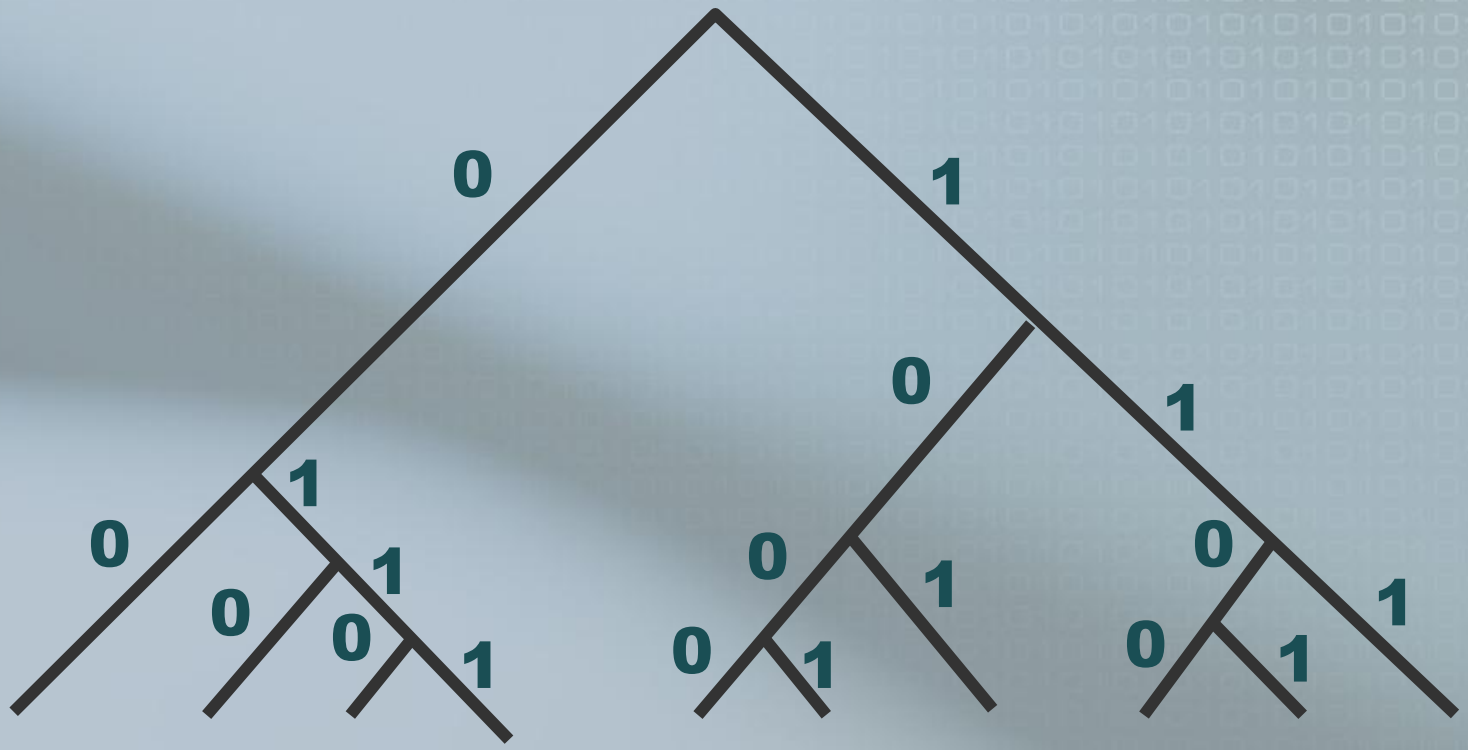




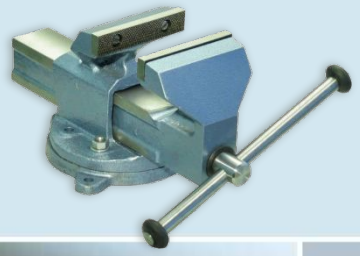


Е	Д	В	А	_	Т	Р	О	Н	,
2	2	4	6	5	2	4	2	2	1

Построение дерева Хаффмана



А	В	Д	,	Е	Н	Р	О	Т	_
00	**010**	**0110**	**0111**	**1000**	**1001**	**101**	**1100**	**1101**	**111**
6	4	2	1	2	2	4	2	2	5



Кодирование текста

А	В	Д	,	Е	Н	Р	О	Т	_
00	010	0110	0111	1000	1001	101	1100	1101	111

НА ДВОРЕ ТРАВА, НА ТРАВЕ ДРОВА

10010011101100101100101100011111101

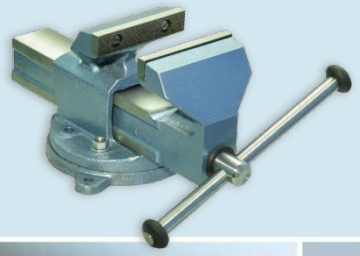
Н А _ Д В О Р Е _ Т

10100010000111111100100111110101101

Р А В А , _ Н А _ Т Р

00010100011101101011100001000

А В Е _ Д Р О В А



Кодирование текста

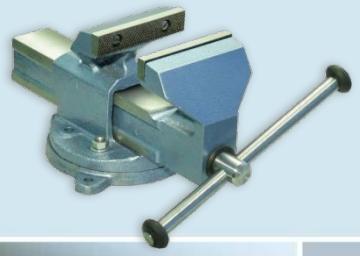
А	В	Д	,	Е	Н	Р	О	Т	_
00	010	0110	0111	1000	1001	101	1100	1101	111

НА ДВОРЕ ТРАВА, НА ТРАВЕ ДРОВА

1001001111011001011001011000011111

0110100010000111111100100111111101

1010001010000111011010111000101000



Коэффициент сжатия

Коэффициентом сжатия называется отношение объема исходного сообщения к объему сжатого.

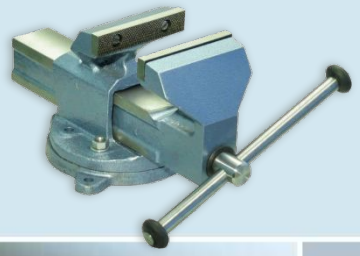
символ	А	В	Д	,	У	Н	Р	О	Т	_
код	**00**	**010**	**0110**	**0111**	**1000**	**1001**	**101**	**1100**	**1101**	**111**
вес	**6**	**4**	**2**	**1**	**2**	**2**	**4**	**2**	**2**	**5**

Объем **сжатого** сообщения:

6*2+**4***3+**2***4+**1***4+**2***4+**2***4+**4***3+**2***4+**2***4+**5***3=95 бит=12 байт.

Объем **исходного** сообщения в ASCII равен 30 байт.

Коэффициент сжатия составляет 30 / 12 = **2,5**



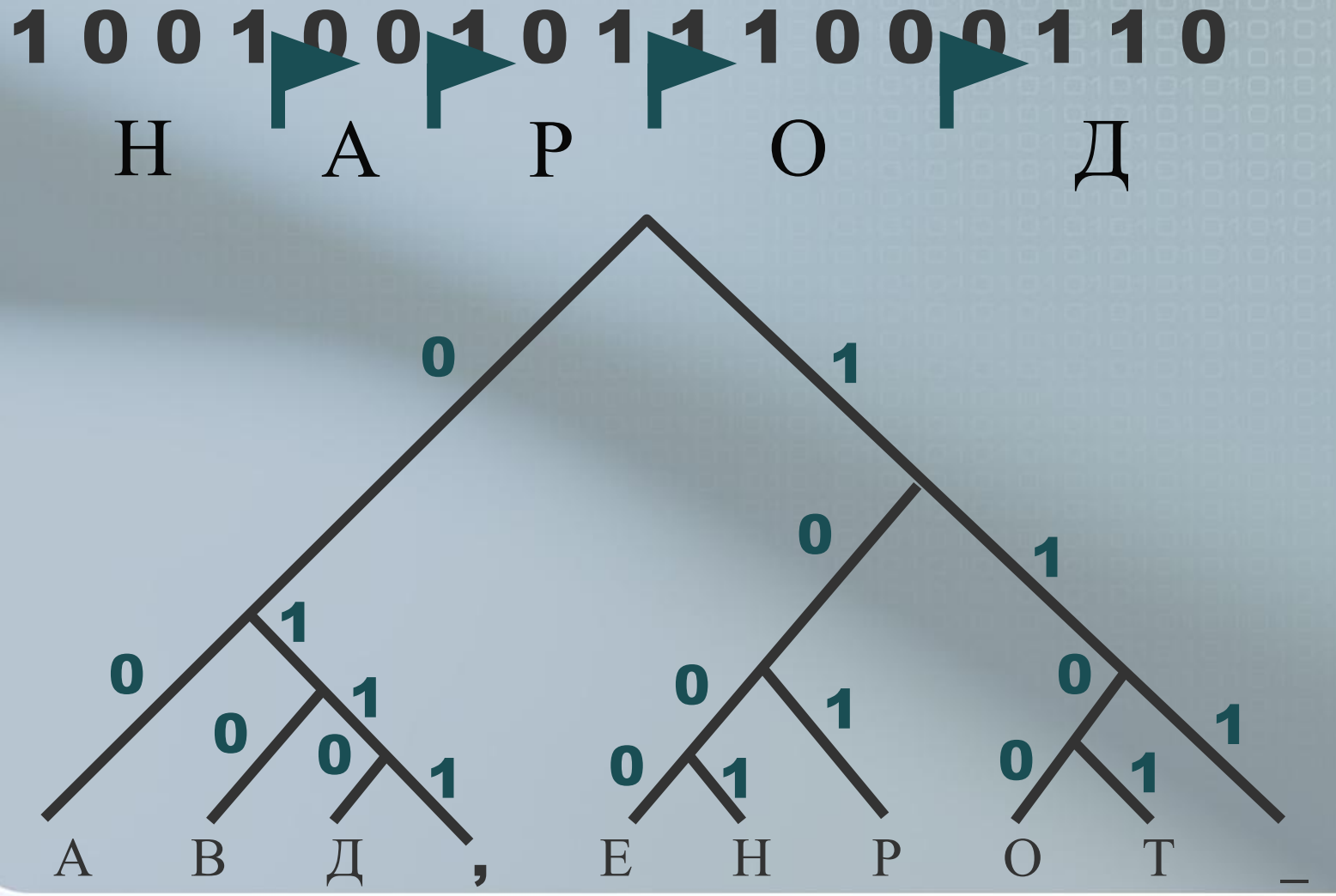
Декодирование

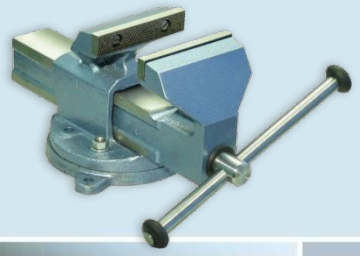
Восстановить исходный текст:

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

символ	А	В	Д	,	У	Н	Р	О	Т	_
код	00	010	0110	0111	1000	1001	101	1100	1101	111

Декодирование





Самостоятельная работа

Постройте код Хаффмана для предложения:

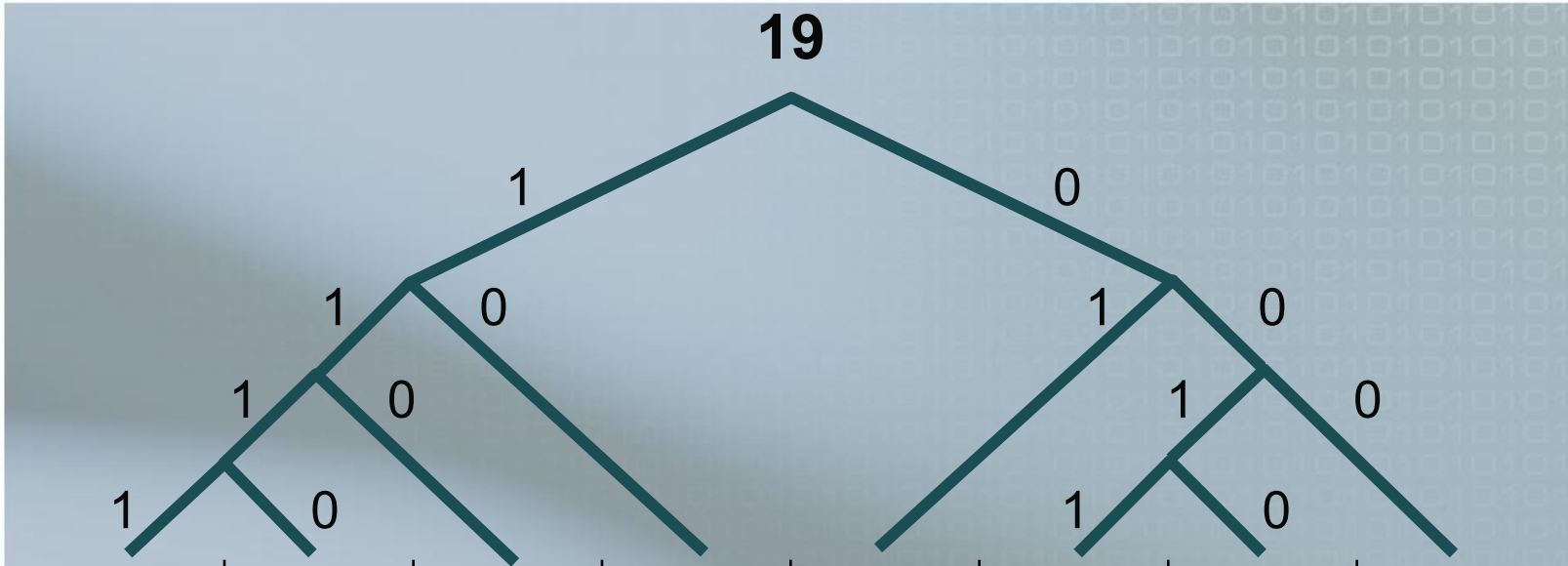
TO BE OR NOT TO BE?

Определите коэффициент сжатия для данной фразы, если каждый символ кодируется в ASCII.

Сравните результат с тем, что был получен при сжатии фразы методом упаковки, сделайте выводы.

Проверьте правильность выполнения задания: закодируйте предложение, используя полученный код, а затем попробуйте его восстановить.

Одно из решений

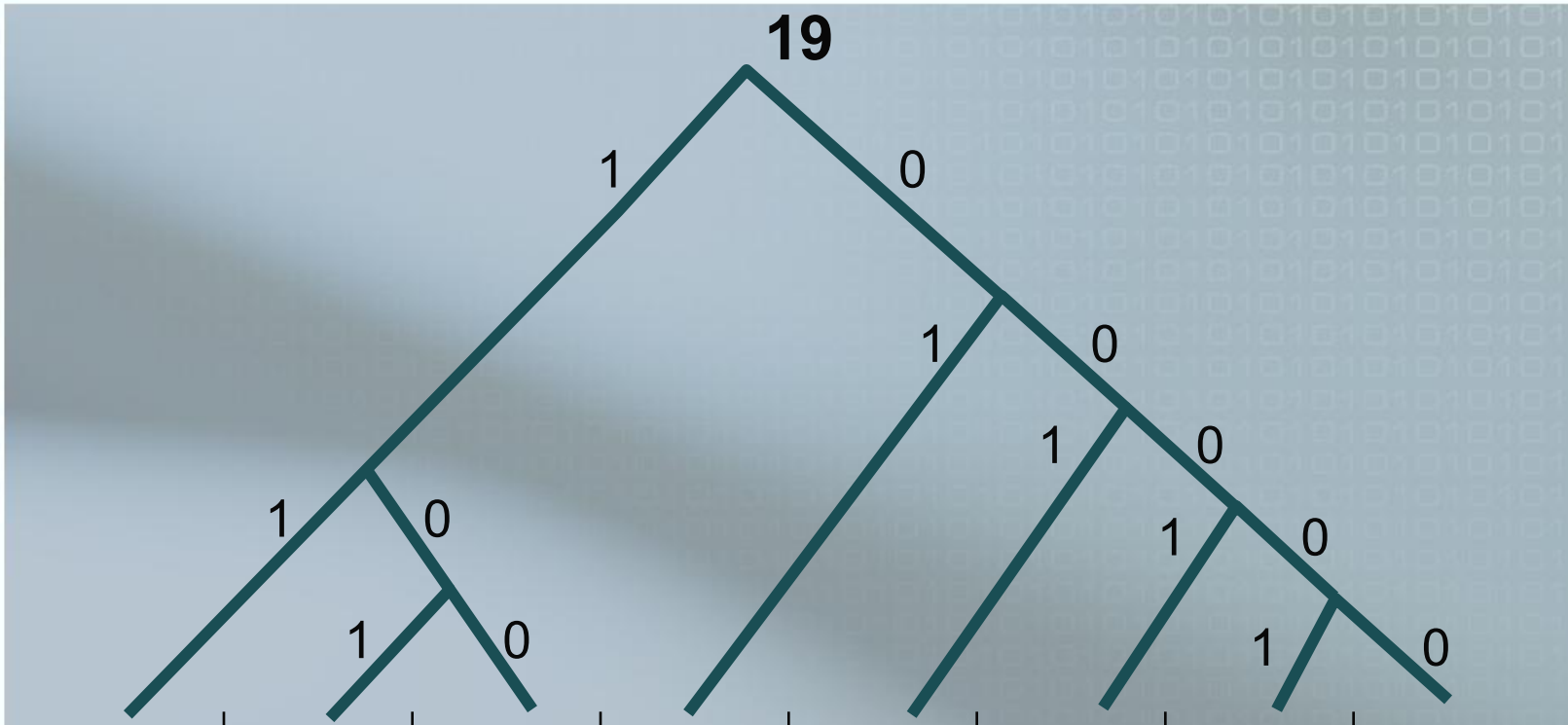


2	1	3	5	4	1	1	2
B	R	T	пробел	O	N	?	E
1111	1110	110	10	01	0011	0010	000

2*4+1*4+3*3+5*2+4*2+1*4+1*4+2*3=53 бита=7 байт

Коэффициент сжатия: 19/7 ≈ **2,7**

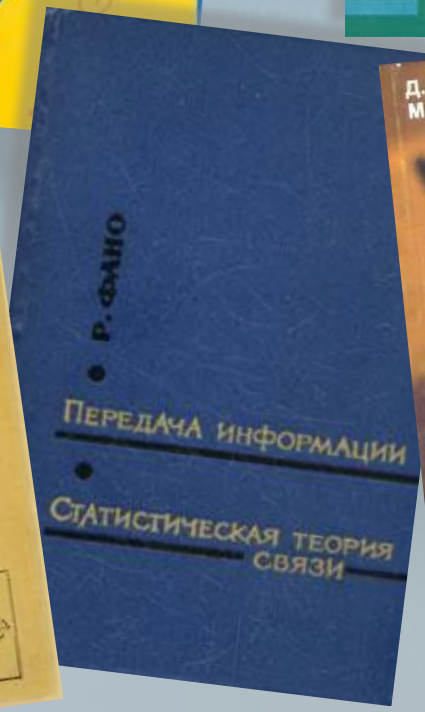
Другое решение

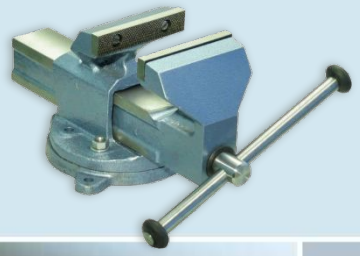


4	2	2	5	3	1	1	1
O	B	E	пробел	T	R	N	?
11	101	100	01	001	0001	00001	00000

4*2+2*3+2*3+5*2+3*3+1*4+1*5+1*5=53 бита

Книги по теме





задание

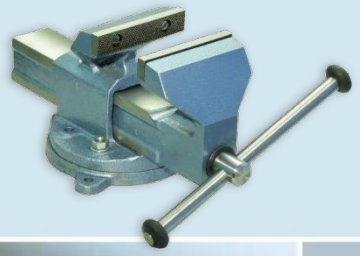
Задание №1

Постройте код Хаффмана для фраз:

1) **Человек как музыкальный инструмент, как настроишь, так и живет.**

2) **Музыка показывает человеку те возможности величия, которые есть в его душе.**

Определите коэффициент сжатия для данной фразы, если каждый символ кодируется в ASCII.



Домашнее задание

Задание №2

На языке Си++ напишите программу, реализующую алгоритм RLE для текстовых данных.

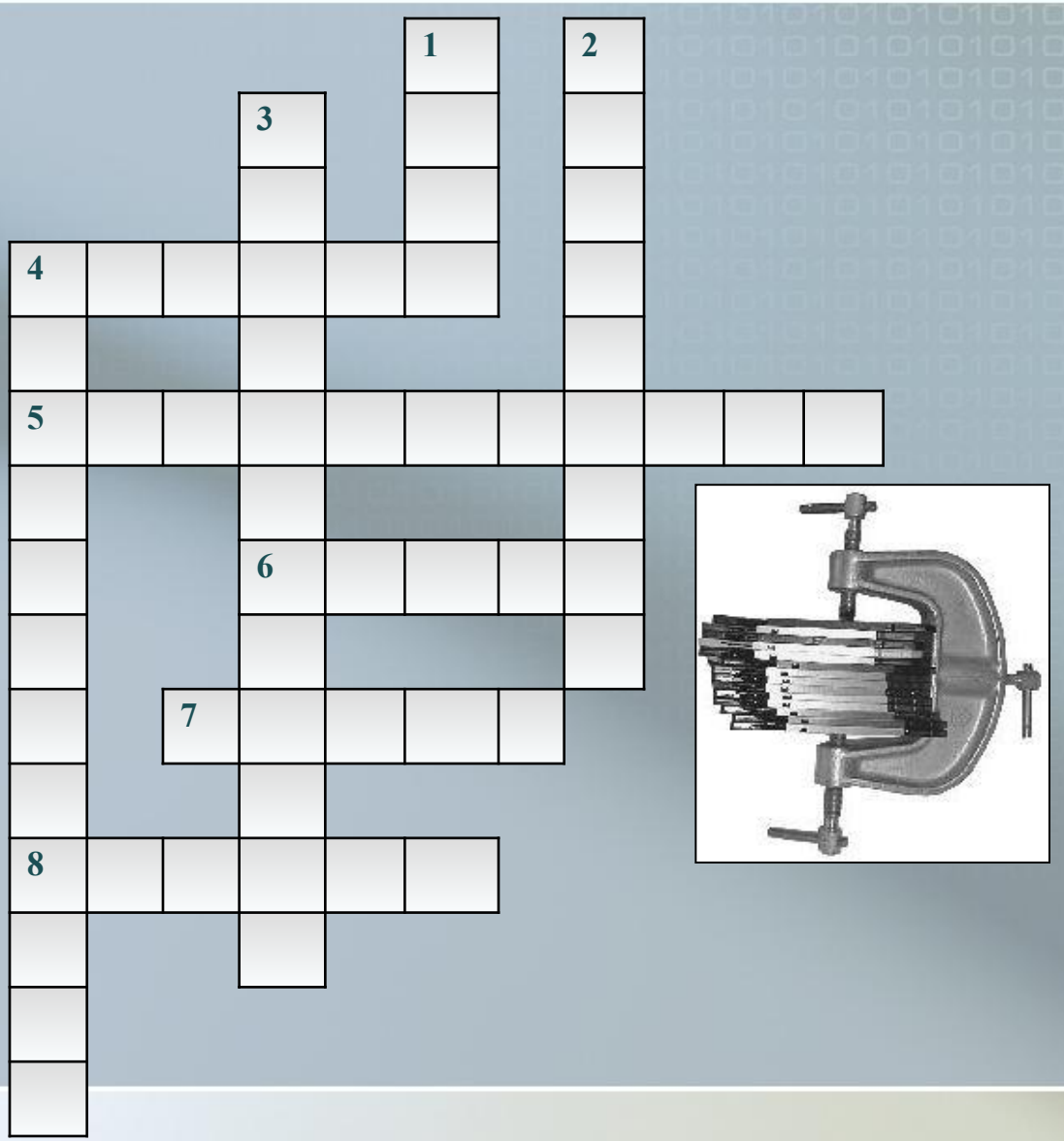
Исходные данные в виде строки, содержащей латинские буквы и пробелы, находятся в текстовом файле input.txt (длина строки не более 255). Результат должен выводиться в текстовый файл output.txt, первая строка которого содержит сжатую строку, вторая – коэффициент сжатия с точностью до сотых.

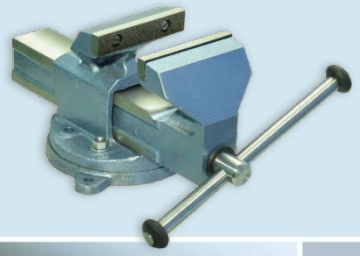
Пример файла:

LLLLLLLLEESSSSSSSSSooooooooooooNN one

Ответ:
8L2E9S12o2N8 1o1n1e
2.32

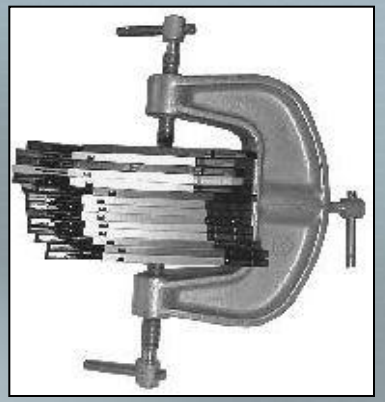
Кроссворд

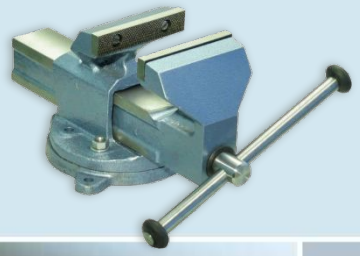




Кроссворд

					ф		а			
			п		а		р			
			р		н		х			
д	е	р	е	в	о		и			
е			ф				в			
к	о	д	и	р	о	в	а	н	и	е
о			к				т			
м			с	л	о	в	о			
п			н				р			
р		м	о	р	з	е				
е			с							
с	ж	а	т	и	е					
с			ь							
и										
я										





Литература

1. *Андреева Е.В.* Математические основы информатики. Элективный курс: учебное пособие / Е.В.Андреева, Л.Л. Босова, И.Н.Фалина – М.:БИНОМ. Лаборатория знаний, 2005.
2. *Гейн А.Г.* Математические основы информатики. / «Информатика» №17 / 2007
3. *Семакин И.Г.* Информатика. Базовый курс. 7-9 классы / И.Г. Семакин, Л.А.Залогова, С.В.Русаков, Л.В.Шестакова. – 2-е изд., испр. И доп. – М.: БИНОМ. Лаборатория знаний, 2004.
4. *Семакин И.Г.* Информатика и ИКТ. Базовый уровень : практикум для 10-11 классов / И.Г.Семакин, Е.К.Хеннер, Т.Ю. Шеина – М. : БИНОМ. Лаборатория знаний, 2007.
5. http://www.compression.ru/ сайт «Все о сжатии»
6. *Устинов П.С.* Архиватор собственными руками! http://vio.fio.ru/vio_09/resource/Print/art_1_9.htm