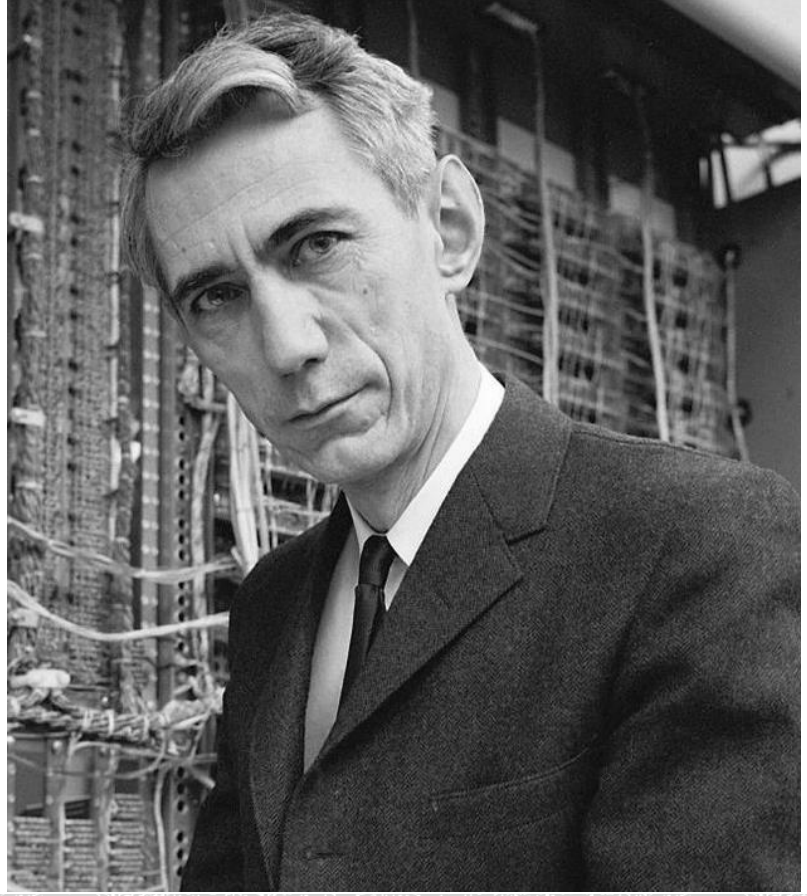
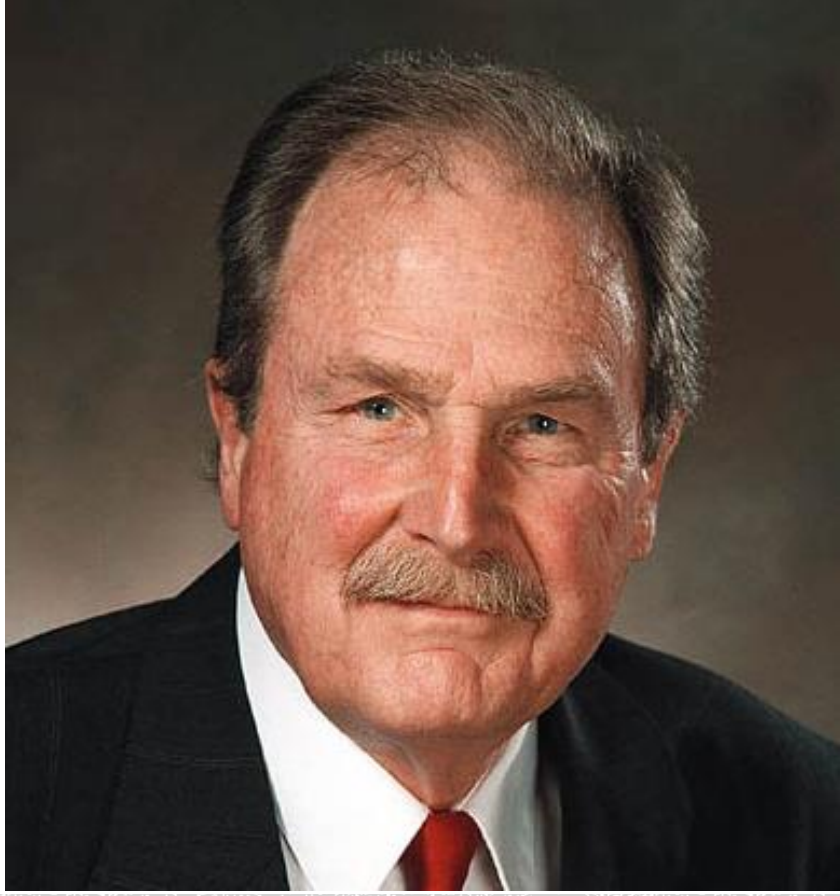



ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ

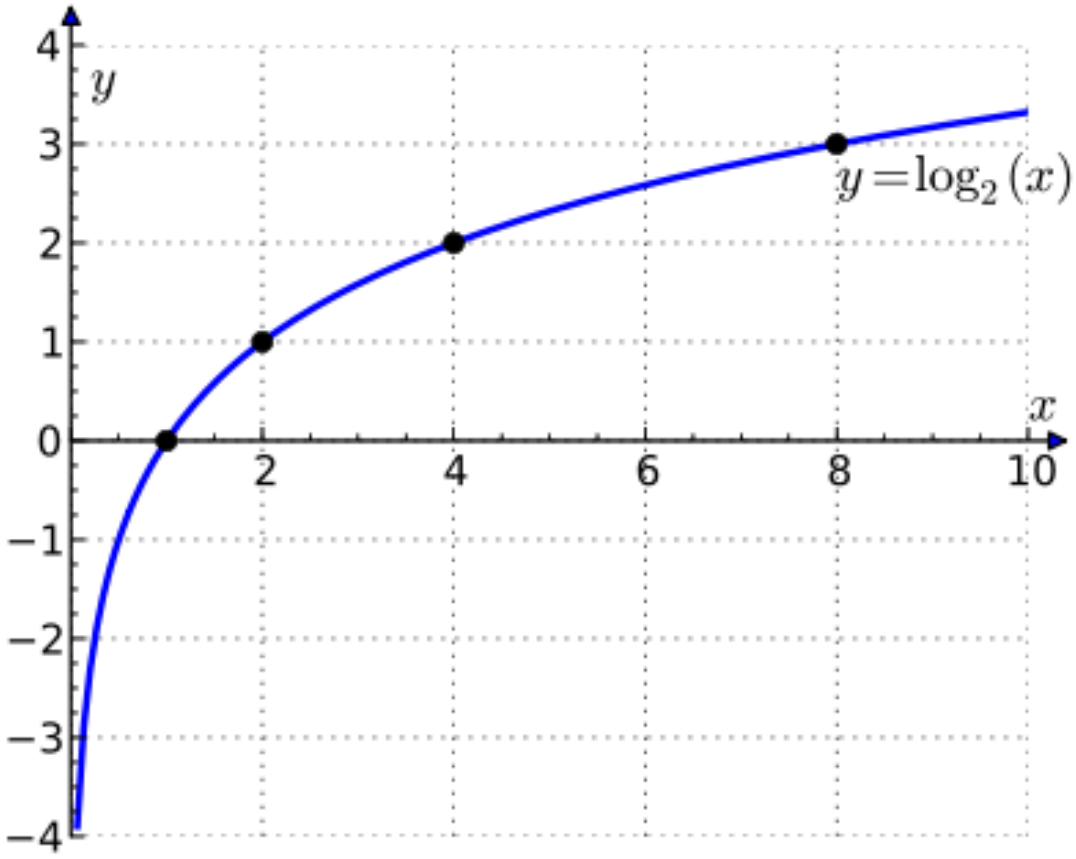
Лекции 3, 4

21 сентября 2018 г.

ПОДХОДЫ К ИЗМЕРЕНИЮ ИНФОРМАЦИИ

Алфавитный

Статистический



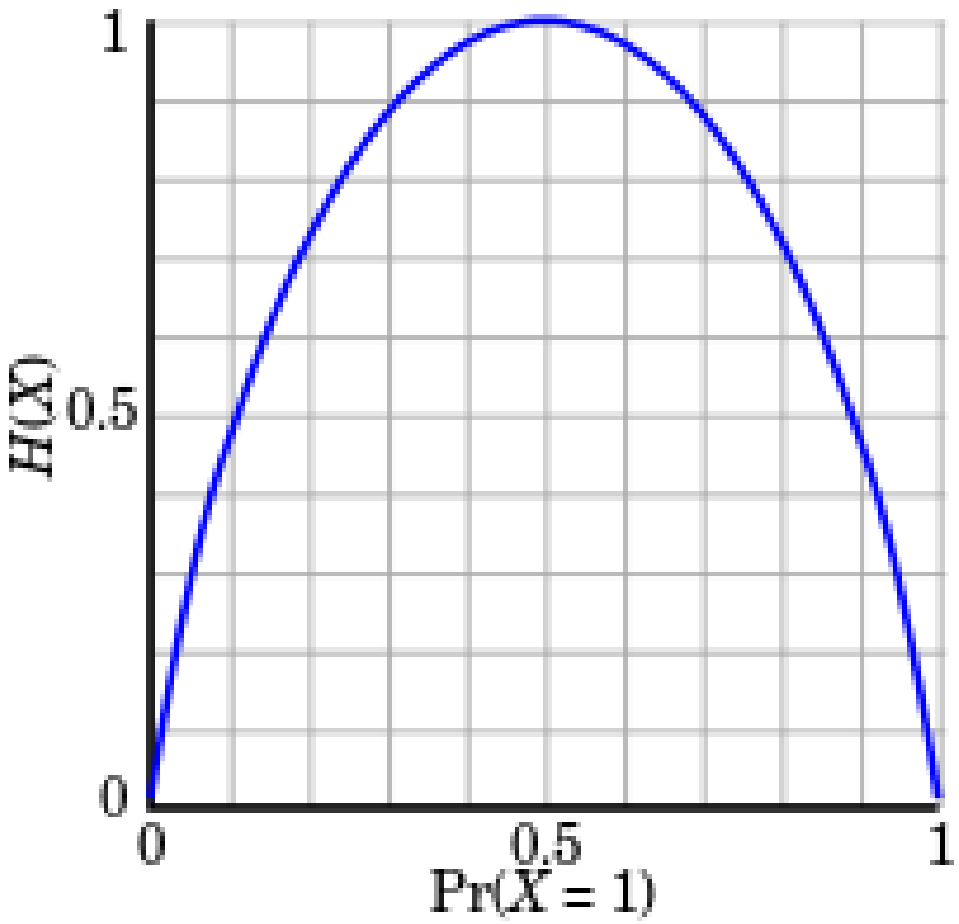
ИЗМЕРЕНИЕ ИНФОРМАЦИИ

Опр. 8: Количество информации по Хартли (Хартлиевская мера информации), содержащееся в последовательности из **n** символов из алфавита **A** мощности **M** равно:

$I = n * \log_2 M$ **бит**

Опр. 12: Количество информации по Шеннону, содержащееся в сообщении из **n** символов, равно:

$\mathrm{n} * \sum_{i=1}^{m} p_i \log_2 \left(\frac{1}{p_i}\right) = n * H(X)$ **бит**



ТЕОРИЯ КОДИРОВАНИЯ

ОПРЕДЕЛЕНИЯ

Опр. 1: Задача кодирования – это задача перевода дискретного сообщения из одного алфавита в другой. Алфавит, с помощью которого представляется информация до преобразования называется **первичным**, а алфавит конечного представления **– вторичным**.

Опр. 2а: Код – это правило или совокупность правил, описывающих соответствие знаков или их сочетаний первичного (исходного) алфавита знакам или их сочетаниям вторичного алфавита.

Опр. 2б: Код – набор знаков вторичного алфавита, используемый для представления знаков или их сочетаний первичного алфавита.

ОПРЕДЕЛЕНИЯ

Опр. 3: Кодирование – это перевод информации, представленной символами первичного алфавита в последовательность кодов.

Опр. 4: Декодирование – операция обратная кодированию – перевод последовательности кодов в соответствующий набор символов первичного алфавита.

Опр. 5: Кодер – устройство, обеспечивающее выполнение операции кодирования.

Опр. 6: Декодер – устройство, производящее декодирование.

ОПРЕДЕЛЕНИЯ

Опр. 7: Общее число символов, составляющих кодовую комбинацию, называется **значностью**, или **длиной кода** n.

Опр. 8: Размер вторичного алфавита называется **основанием кода m**.

ОБРАТИМОСТЬ КОДИРОВАНИЯ

Условие обратимости

Опр. 10: Операции кодирования и декодирования называются **обратимыми,** если их последовательное применение не приводит к потере информации.

Опр. 11: Принципиально необратимое кодирование (хэширование) – такое кодирование, при котором восстановление исходной информации невозможно.

Опр. 12: Обратимое с помощью дополнительной информации кодирование – для восстановления исходного сообщения требуется дополнительная информация (пароль).

СРЕДНЯЯ ДЛИНА КОДА

Опр. 13: Средняя длина кода $K = \sum_{i=1}^{N} n_i p_i$

Опр. 14: Минимальная длина кода $K_{min} = \frac{I_A}{I_B}$

Опр. 15: Относительная избыточность кода $\mathrm{Q} = \frac{K - K_{min}}{K_{min}}$

n_i - длина кода для i-го символа **A**

p_i - вероятность появления i-го символа

I_A и I_B – энтропии **A** и **B**

Пример: Определите среднюю, минимальную длину и избыточность кода для символов первичного алфавита {«А», «Б», «В», «Г»}, если при кодировании получены следующие коды этих символов: «А»- 0, «Б»- 10, «В»- 110, «Г»-111 с известными вероятностями появления символов {0,45; 0,15; 0,05; 0,35}.

Пример: Определите среднюю, минимальную длину и избыточность кода для символов первичного алфавита {«А», «Б», «В», «Г»}, если при кодировании получены следующие коды этих символов: «А»- 0, «Б»- 10, «В»- 110, «Г»-111 с известными вероятностями появления символов {0,45; ***0,35****; 0,05;* ***0,15****}.*

АСИМПТОТИЧЕСКИ ОПТИМАЛЬНЫЙ КОД

Опр. 16: Для некоторого первичного алфавита A и вторичного алфавита B **асимптотически оптимальным кодом** называется такой способ кодирования, при котором избыточность кодирования стремится к нулю, если длина сообщений стремится к бесконечности.

Первая теорема Шеннона: *При отсутствии шумов всегда возможен такой вариант кодирования сообщения, при котором относительная избыточность кода будет сколь угодно близка к нулю.*

КЛАССИФИКАЦИИ

Длины кодов

- равномерные
- неравномерные

Обнаружение и исправление ошибок

- простые
- корректирующие

Число одновременно кодируемых символов первичного алфавита

- алфавитные
- блочные

КОДИРОВАНИЕ ИНФОРМАЦИИ

КОД ФИКСИРОВАННОЙ ДЛИНЫ (FLC)

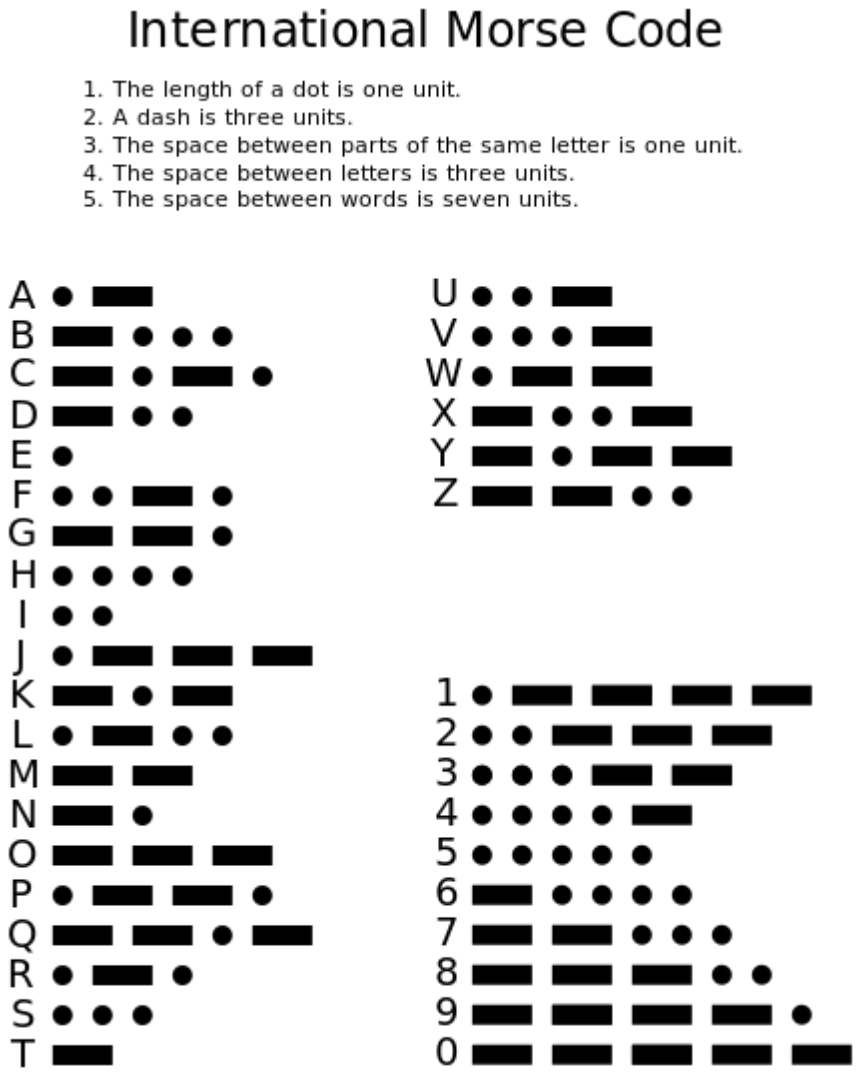
- В современных компьютерах символы кодируются с помощью байтов
- Один байт – 8 бит
- **ASCII** – кодовая таблица, кодирующая все основные печатные символы

32		00100000	56	8	00111000	80	P	01010000	104	h	01101000
33	!	00100001	57	9	00111001	81	Q	01010001	105	i	01101001
34	"	00100010	58	:	00111010	82	R	01010010	106	j	01101010
35	#	00100011	59	;	00111011	83	S	01010011	107	k	01101011
36	$	00100100	60	<	00111100	84	T	01010100	108	l	01101100
37	%	00100101	61	=	00111101	85	U	01010101	109	m	01101101
38	&	00100110	62	>	00111110	86	V	01010110	110	n	01101110
39	'	00100111	63	?	00111111	87	W	01010111	111	o	01101111
40	(	00101000	64	@	01000000	88	X	01011000	112	p	01110000
41	)	00101001	65	A	01000001	89	Y	01011001	113	q	01110001
42	*	00101010	66	B	01000010	90	Z	01011010	114	r	01110010
43	+	00101011	67	C	01000011	91	[	01011011	115	s	01110011
44	,	00101100	68	D	01000100	92	\	01011100	116	t	01110100
45	-	00101101	69	E	01000101	93	]	01011101	117	u	01110101
46	.	00101110	70	F	01000110	94	^	01011110	118	v	01110110
47	/	00101111	71	G	01000111	95	_	01011111	119	w	01110111
48	0	00110000	72	H	01001000	96	`	01100000	120	x	01111000
49	1	00110001	73	I	01001001	97	a	01100001	121	y	01111001
50	2	00110010	74	J	01001010	98	b	01100010	122	z	01111010
51	3	00110011	75	K	01001011	99	c	01100011	123	{	01111011
52	4	00110100	76	L	01001100	100	d	01100100	124	|	01111100
53	5	00110101	77	M	01001101	101	e	01100101	125	}	01111101
54	6	00110110	78	N	01001110	102	f	01100110	126	~	01111110
55	7	00110111	79	O	01001111	103	g	01100111	127	□	01111111

КОД ПЕРЕМЕННОЙ ДЛИНЫ (VLC)

- Пример **VLC**-кода – азбука Морзе
- Принцип энтропийного (статистического) кодирования – построить **VLC** таким образом, чтобы **часто используемые** символы первичного алфавита кодировались самими **короткими кодами**, а редкие – длинными.
- Зная распределение частот, мы можем подсчитать минимальное среднее число необходимых бит на символ – это и есть энтропия.

Опр.: Энтропия – минимальное количество бит на один символ первичного алфавита (в среднем).

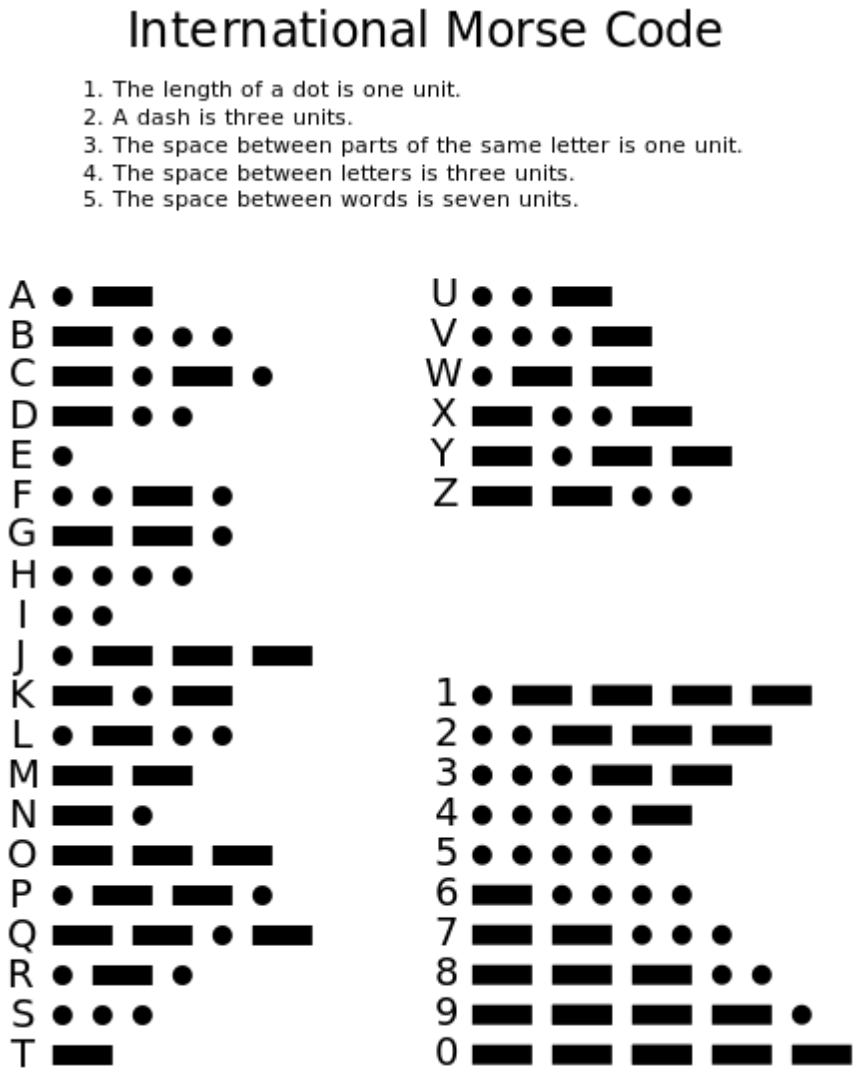


ПРЕФИКСНЫЕ КОДЫ

Опр.: Префиксный код в теории кодирования — код со словом переменной длины, удовлетворяющий условию Роберта Фано.

Опр.: Условие Фано: неравномерный код может быть однозначно декодирован, если никакой из кодов не совпадает с началом (префиксом) какого-либо другого, более длинного кода.

Не префиксный код!



ПРЕФИКСНЫЙ КОД ШЕННОНА-ФАНО

Алгоритм 1: Построение кода Шеннона-Фано

0. Строим таблицу частот символов и упорядочиваем ее

1. Если в таблице одна строка, то ничего не делаем

2. Делим таблицу на две части так, чтобы сумма частот в первой части была как можно ближе к сумме частот из второй части

3. К кодам символов из первой части таблицы дописываем 0, а из второй – 1.

4. Выполняем алгоритм отдельно для первой и второй частей таблицы.

ПРЕФИКСНЫЙ КОД ШЕННОНА-ФАНО

Первичный алфавит	Вероятности появления символов
D	0,3
B	0,2
E	0,2
A	0,15
C	0,1
F	0,05

$$K =$$

$$I^A =$$

$$Q =$$

ПРЕФИКСНЫЙ КОД ХАФФМАНА

Алгоритм 2: Построение кода Хаффмана

0. Для каждого символа из алфавита строим вершину с весом, равным частоте этого символа.

1. Если количество вершин равно 1, то заканчиваем.

2. Упорядочиваем вершины по убыванию веса.

3. Создаем новую вершину и «подцепляем» к ней две вершины с самыми низкими частотами. Частоту новой вершины задаем как сумму частот двух исходных. На ребре, ведущем к левому сыну пишем 0, а к правому – 1.

4. Удаляем **из списка** «подцепленные» вершины и переходим на шаг 1.

Код символа будет записан на ребрах, находящихся на пути из корня (последней оставшейся вершины) к вершине с этим символом.

ПРЕФИКСНЫЙ КОД ХАФФМАНА

D $_{0,3}$ B $_{0,2}$ E $_{0,2}$ A $_{0,15}$ C $_{0,1}$ F $_{0,05}$

$K =$

$I^A =$

$Q =$

УПРАЖНЕНИЯ

1. Определите является ли префиксным следующий двоичный код первых 5 букв латинского алфавита:

a	b	c	d	e
000	110	01	001	10

Декодируйте сообщение 1100000100110, записанное с использованием приведенного кода.

2. Для символов первичного алфавита A={«+», «-», «*», «/»} с известными вероятностями появления символов P={0,4; 0,2; 0,3; 0,1}:
   a) Вычислите среднюю и минимальную длины кода и относительную избыточность, если символы кодируются следующим образом: «+» - 0, «-» - 100, «*» - 1010, «/» - 1100
   b) Постройте префиксный код по алгоритму Шеннона-Фано и подсчитайте K, I_A, Q
   c) Постройте префиксный код по алгоритму Хаффмана и подсчитайте K, I_A, Q